Unified Probabilistic Deep Continual Learning through Generative Replay and Open Set Recognition

Martin Mundt, Sagnik Majumder, Iuliia Pliushch, Yong Won Hong, and Visvanathan Ramesh

Abstract—We introduce a probabilistic approach to unify open set recognition with the prevention of catastrophic forgetting in deep continual learning, based on variational Bayesian inference. Our single model combines a joint probabilistic encoder with a generative model and a linear classifier that get shared across sequentially arriving tasks. In order to successfully distinguish unseen unknown data from trained known tasks, we propose to bound the class specific approximate posterior by fitting regions of high density on the basis of correctly classified data points. These bounds are further used to significantly alleviate catastrophic forgetting by avoiding samples from low density areas in generative replay. Our approach requires neither storing of old, nor upfront knowledge of future data, and is empirically validated on visual and audio tasks in class incremental, as well as cross-dataset scenarios across modalities.

Index Terms—Continual Deep Learning, Catastrophic Forgetting, Open Set Recognition, Variational Inference, Deep Generative Models.

1 Introduction

Modern machine learning systems are typically trained in a closed world setting according to an isolated learning paradigm. They take on the assumption that data is available at all times and data inputs encountered during application of the learned model come from the same statistical population as the training data. However, the real world requires dealing with sequentially arriving tasks and data coming from potentially yet unknown sources. A neural network that is trained exclusively on such newly arriving data will overwrite its representations and thus forget knowledge of past tasks, an early identified phenomenon coined catastrophic forgetting [1]. Moreover, when confronting the learned model with unseen concepts, overconfident misclassification is inevitable [2].

Existing continual learning literature predominantly concentrates its efforts on finding mechanisms to alleviate catastrophic forgetting [3] and the term continual learning is not necessarily used in a wider sense. Specifically, the aforementioned crucial system component to distinguish seen from unseen unknown data, both as a guarantee for robust application and to avoid the requirement of explicit task labels for prediction, is generally missing. A naive conditioning on unseen unknown data through inclusion of a "background" class is infeasible as by definition we do not have access to it a priori. Commonly applied thresholding of prediction values is veritably insufficient as resulting large confidences cannot be prevented [2]. Arguably this also includes variational methods [4], [5], [6], [7] to gauge neural network uncertainty, since the closed world assumption also holds true for Bayesian methods [8]. Recently, Bendale et al. [9] have proposed extreme value theory (EVT) based meta-recognition to address open set detection on the basis of softmax predictions in conventional feed-forward deep neural networks. Inspired by this work, we propose a probabilistic approach to unify open set recognition and the prevention of catastrophic forgetting in continual learning of a single deep model. Our specific contributions are:

- We introduce a single model for continual learning that combines a joint probabilistic encoder with a generative model and a linear classifier. This architecture enables a natural formulation to address open set recognition on the basis of EVT bounds to the class conditional approximate posterior in variational Bayesian inference.
- Apart from using EVT for detection of unseen unknown data, we show that generated samples from areas of low probability density under the aggregate posterior can be excluded in generative replay for continual learning. This leads to significantly reduced catastrophic forgetting without storing real data.
- Empirically, we show that our model can incrementally learn the classes of two image and one audio dataset, as well as cross-dataset scenarios across modalities, while being able to successfully distinguish various unseen datasets from data belonging to known tasks.
- Finally, we demonstrate how our proposed framework can be extended and readily profits from recent advances in deep generative modelling, such as autoregression [10], [11], [12] and introspection [13], [14]. This is then empirically validated by scaling to high resolution color images in further experiments.

The remainder of the paper follows the structure of

- *M. Mundt, I. Pliushch and V. Ramesh are with the Department of Computer Science, Geothe University, Frankfurt am Main, Germany. E-mail: {mmundt, vramesh}@em.uni-frankfurt.de*
- *S. Majumder is with the Department of Computer Science, University of Texas at Austin, USA, but was with the Department of Computer Science at Goethe University while working on this project.*
- *Y. W. Hong is with the Department of Computer Science, Yonsei University, Seoul, Republic of Korea.*

these listed contributions. We start section two by formally describing continual learning and open set recognition in the context of deep supervised learning, followed by a respective review of recent literature. Section three provides a step by step introduction of our probabilistic framework to unify open set recognition with the prevention of catastrophic forgetting in continual learning. Section four proceeds with an experimental evaluation and analysis, which is then revisited and extended in section five under the consideration of recent auxiliary generative modelling advances. The sixth and final section concludes the paper.

2 Background and Related Work

2.1 Continual Learning

In isolated supervised machine learning the core assumption is the presence of i.i.d. data at all times and training is conducted using a dataset $\boldsymbol{D} \equiv \left\{\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)\right\}_{n=1}^{N}$, consisting of N pairs of data instances $\boldsymbol{x}^{(n)}$ and their corresponding labels $y^{(n)} \in \{1 \dots C\}$ for C classes. In contrast, in continual learning task data $\boldsymbol{D}_t \equiv \left\{\left(\boldsymbol{x}_t^{(n)}, y_t^{(n)}\right)\right\}_{n=1}^{N_t}$ with $t = 1, \dots, T$ arrives sequentially for T disjoint datasets, each with number of classes C_t. It is assumed that only the data of the current task is available. Different methods in the literature have been identified to prevent a model from forgetting past knowledge, either explicitly, through regularization or freezing of weights, or implicitly, through rehearsal of data by sampling retained subsets or sampling from a generative memory. A recent review of many continual learning methods is provided by Parisi et al. [3]. Here, we present a brief summary of particular related works.

Regularization and weight freezing: Regularization methods such as synaptic intelligence (SI) [15] or elastic weight consolidation (EWC) [16] explicitly constrain the weights during continual learning to avoid drifting too far away from previous tasks' solutions. In a related picture, learning without forgetting [17] uses knowledge distillation [18] to regularize the end-to-end functional. Further methods employ dynamically expandable neural networks [19] or progressive networks [20], that expand the capacity while freezing or regularizing existing representations.

Rehearsal: These methods store and rehearse data from distributions belonging to old tasks or generate samples in pseudo-rehearsal [21]. The central component of the former is thus the selection of significant instances. For methods such as iCarl [22] it is therefore common to resort to auxiliary techniques such as a nearest-mean classifier [23] or coresets [24]. Inspired by complementary learning systems theory [25], dual-model approaches sample data from a separate generative memory. In GeppNet [26] an additional long-short term memory [27] is used for storage, whereas generative replay [28] samples form a separately trained generative adversarial network (GAN) [29].

Bayesian methods: As detailed in Variational Generative Replay (VGR) [6], Bayesian methods provide natural capability for continual learning by making use of the learned distribution. Existing works nevertheless fall into the above two categories: a prior-based approach using the former task's approximate posterior as the new task's prior [30] or estimating the likelihood of former data through generative replay or other forms of rehearsal [6], [7].

Evaluation assumptions and multiple model heads: The success of many of these techniques can be attributed mainly to the considered evaluation scenario. With the exception of VGR [6], all above techniques train a separate classifier per task and thus either require explicit storage of task labels, or assume the presence of a task oracle during evaluation. This multi-head classifier scenario prevents "cross-talk" between classifier units by not sharing them, which would otherwise rapidly decay the accuracy as newly introduced classes directly confuse existing concepts. While the latter is acceptable to assess catastrophic forgetting, it also signifies a major limitation in practical application. Even though VGR [6] uses a single classifier, they train a separate generative model per task to avoid catastrophic forgetting of the generator.

Our approach builds upon these previous works by proposing a single model with single classifier head with a natural mechanism for open set recognition and improved generative replay from a Bayesian perspective.

2.2 Out-of-distribution and open set recognition

The above mentioned literature focuses their continual learning efforts predominantly on addressing catastrophic forgetting. Corresponding evaluation is thus conducted in a closed world setting, where instances that do not belong to the observed data distribution are not encountered. In reality, this is not guaranteed as users could provide arbitrary input or unknowingly present the system with novel inputs that deviate substantially from previously seen instances. Our models thus need the ability to identify unseen examples in an open world and categorize them as either belonging to the already known set of classes or as presently being unknown. We briefly recall the formal definition of open set recognition presented in Scheirer et al. [31] and corresponding follow-up literature [8], [9], [32], [33]: *For any recognition function f over an input space $\mathcal{X}$, the open space $\mathcal{O}$ is defined as $\mathcal{O} \subset= \mathcal{X} - \mathcal{S}_K$, where $\mathcal{S}_K$ is a union of balls of radius r_o including all of the training examples for known classes $x \in K$.* The goal in open set recognition is to learn this function f using the training data of known classes, i.e. minimizing the empirical risk R_ϵ (expected loss $\mathbb{E}[L(\dots)]$), while simultaneously limiting the open space risk $\mathcal{R}_\mathcal{O}(f) = \int_\mathcal{O} f_K(x)dx / \int_{\mathcal{S}_K} f_K(x)dx$. Minimizing the latter requires the ability to detect novelty with respect to the empirically observed distribution.

We provide a small overview of approaches that can be regarded as related to solving open set recognition in deep neural networks. A more comprehensive and general overview of recent methods is provided in the review by Boult et al. [8].

Bayesian uncertainty and deep generative models: Bayesian neural network models [34] could be believed

to intrinsically be able to reject statistical outliers through model uncertainty [6]. In inference with deep neural networks, it has been suggested that the use of stochastic forward passes with Monte-Carlo Dropout [35] provides a suitable approximation. However, repeating the argument of Boult et al. [8], this is generally insufficient as uncertain inputs are not necessarily unknown and unknowns do not necessarily have to appear as uncertain. In the context of deep generative models that are trained with various variational approximations, it is particularly well known that relying solely on deep uncertainty quantification to distinguish unseen data is unsatisfactory [36], [37].

Calibration: The aim of these works is to seperate a known and unknown input through prediction confidence, often by fine-tuning or re-training an already existing model. In ODIN [38] this is addressed through perturbations and temperature scaling, while Lee et al. [39] use a separately trained GAN to generate out-of-distribution samples from low probability densities and explicitly reduce their confidence through inclusion of an additional loss term. Similarly the objectosphere loss [40] defines an objective that explicitly aims to maximize entropy for upfront available unknown inputs.

Extreme value theory: One approach to open set recognition in deep neural networks is through extreme-value theory (EVT) based meta-recognition [9], [32], i.e. without re-training or modifying loss functions by assuming upfront presence of unknown data. The goal here is to bound the open space on the basis of already seen data instances. Scheirer et al. [32] have introduced the notion of a compact abating probability, a probabilistic model where the recognition function's probability decreases monotonically with increasing distance to known training points. They have identified the Weibull distribution as a suitable candidate to satisfy the latter when modelling the extreme prediction values. Bendale et al. [9] have extended this to the use with deep neural networks. They empirically observe that the penultimate layer of a deep neural network can be used as the underlying feature space for open set recognition. On the basis of extreme values to this layer's average activation values, the authors devise a procedure to revise a deep neural network's softmax prediction values. The proposed OpenMax algorithm thus aims to mitigate the issue of predicted scores summing to unity and unseen unknown data instances can in principle be assigned zero probability across all known classes.

Our work extends these approaches by moving away from potentially non-calibrated predictive values or empirically chosen deep neural network feature spaces. We instead propose to use EVT to bound the approximate posterior. In contrast to predictive values such as reconstruction losses, where differences in reconstructed images do not necessarily have to reflect the outcome with respect to our task's target, we thus directly operate on the underlying (lower-bound to the) data distribution, and the generative factors. This additionally allows us to constrain generative replay to distribution inliers, which further alleviates catastrophic forgetting in continual learning. While we can still leverage variational inference to gauge model uncertainty, the need to rely on classifier entropy or confidence, that are known to be overconfident and can never be calibrated for all unknown inputs, is circumvented.

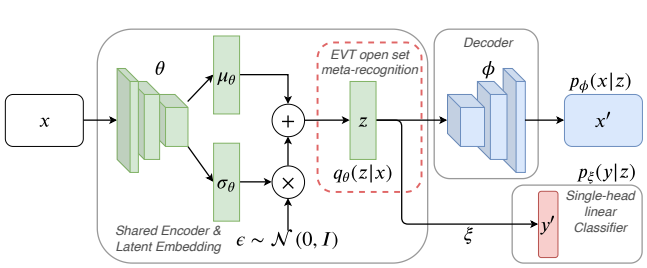


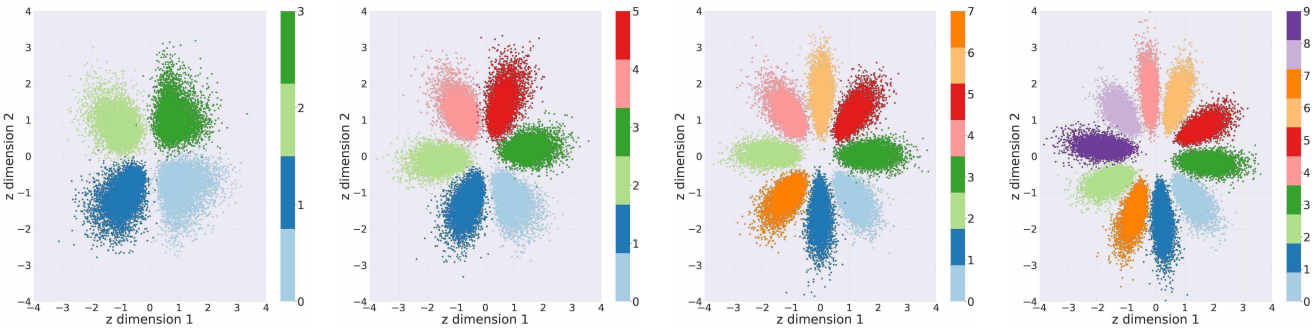
Fig. 1: Joint continual learning model consisting of a shared probabilistic encoder $q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$, probabilistic decoder $p_{\boldsymbol{\phi}}(\boldsymbol{x}, \boldsymbol{z})$ and probabilistic classifier $p_{\boldsymbol{\xi}}(y, \boldsymbol{z})$. For open set recognition and generative replay with outlier rejection, EVT based bounds on the basis of the approximate posterior are established.

3 Unifying Open Set Recognition with the Prevention of Catastrophic Forgetting in Continual Learning

We consider the continual learning scenario with awareness of an open world from a perspective of variational inference in deep generative models [5]. Our model consists of a shared encoder with variational parameters $\boldsymbol{\theta}$, decoder and linear classifier with respective parameters $\boldsymbol{\phi}$ and $\boldsymbol{\xi}$. The joint probabilistic encoder learns an encoding to a latent variable $\boldsymbol{z}$, over which a unit Gaussian prior is placed. Using variational inference, the encoder's purpose is to approximate the true posterior to both $p_{\boldsymbol{\phi}}(\boldsymbol{x}, \boldsymbol{z})$ and $p_{\boldsymbol{\xi}}(y, \boldsymbol{z})$. The probabilistic decoder $p_{\boldsymbol{\phi}}(\boldsymbol{x}|\boldsymbol{z})$ and probabilistic linear classifier $p_{\boldsymbol{\xi}}(y|\boldsymbol{z})$ then return the conditional probability density of the input $\boldsymbol{x}$ and target y under the respective generative model given a sample $\boldsymbol{z}$ from the approximate posterior $q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$. This yields a generative model $p(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, for which we assume a factorization and generative process of the form $p(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})$. For variational inference with this model, the sum over all elements in the dataset n $\in$ D of the following loss thus needs to be optimized:

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}\right) &= -\beta KL(q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}^{(n)}) \,||\, p(\boldsymbol{z})) \\ &+ \mathbb{E}_{q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}^{(n)})}\left[\log p_{\boldsymbol{\phi}}(\boldsymbol{x}^{(n)}|\boldsymbol{z}) + \log p_{\boldsymbol{\xi}}(y^{(n)}|\boldsymbol{z})\right]\end{aligned} \tag{1}$$

This model can be seen as a variant of β-VAE [41], where in addition to approximating the data distribution the model learns to incorporate the class structure into the latent space. It forms the basis for continual learning with open set recognition and respective improvements to generative replay, which will be discussed in subsequent sections. An illustration of the model is shown in figure 1 and the corresponding full derivation of equation 1, the lower-bound to the joint distribution $p(\boldsymbol{x}, \boldsymbol{y})$ is supplied in the appendix.



(a) 4-classes (b) 6-classes (c) 8-classes (d) 10-classes

Fig. 2: (a) 2-D latent space visualization for continually learned MNIST.

3.1 Learning continually through generative replay

Without further constraints, one could continually train above model by sequentially accumulating and optimizing equation 1 over all currently present tasks $t = 1, \ldots, T$:

$$\mathcal{L}_t^{\text{UB}}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}) = \frac{1}{t} \sum_{\tau=1}^{t} \frac{1}{N_\tau} \sum_{n=1}^{N_\tau} \mathcal{L}\left(\boldsymbol{x}_\tau^{(n)}, \boldsymbol{y}_\tau^{(n)}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}\right) \tag{2}$$

Being based on the accumulation of real data, this equation provides an upper-bound to achievable performance in continual learning. However, this form of continued training is generally infeasible if only the most recent task's data is assumed to be available. Making use of the generative nature of our model, we follow previous works [6], [7] and estimate the likelihood of former data through generative replay:

$$\begin{aligned} \mathcal{L}_t(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}) = & \frac{1}{2} \frac{1}{N_t} \sum_{n=1}^{N_t} \mathcal{L}\left(\boldsymbol{x}_t^{(n)}, \boldsymbol{y}_t^{(n)}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}\right) \\ & + \frac{1}{2} \frac{1}{N_t'} \sum_{n=1}^{N_t'} \mathcal{L}\left(\boldsymbol{x}_t'^{(n)}, \boldsymbol{y}_t'^{(n)}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}\right) \end{aligned} \tag{3}$$

where,

$$\boldsymbol{x}_t' \sim p_{\boldsymbol{\phi}, t-1}(\boldsymbol{x}|\boldsymbol{z});\ y_t' \sim p_{\boldsymbol{\xi}, t-1}(y|\boldsymbol{z}) \text{ and } \boldsymbol{z} \sim p(\boldsymbol{z}) \tag{4}$$

Here, $\boldsymbol{x}_t'$ is a sample from the generative model with its corresponding label y_t' obtained from the classifier. N_t' is the number of total data instances of all previously seen tasks or alternatively a hyper-parameter. This way the expectation of the log-likelihood for all previously seen tasks is estimated and the dataset at any point in time $\tilde{\boldsymbol{D}}_t \equiv \left\{(\tilde{\boldsymbol{x}}_t^{(n)}, \tilde{y}_t^{(n)})\right\}_{n=1}^{\tilde{N}_t} = \{(\boldsymbol{x}_t \cup \boldsymbol{x}_t', y_t \cup y_t')\}$ is a combination of generations from seen past data distributions and the current task's real data.

3.2 Linear classifier expansion and the role of β

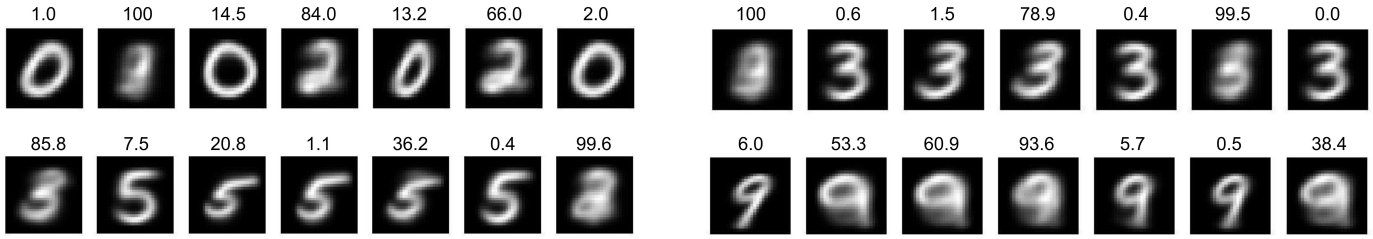
In contrast to prior works based on multiple models, our approach of using equation 3 to continually train a single model has two implications. With every encounter of an additional class: 1. a new classifier unit and corresponding weights need to be added. 2: the latent encoding needs to adjust to accommodate the additional class under the constraint of the classifier requirement of linear separability.

The first implication can be addressed by expanding the existing classifier weight tensor and only initializing the newly added weights. If the distribution from which the newly added weights are drawn is independent of the number of classes and only depends on the input dimensionality, such as the initialization scheme proposed by He et al. [42], the initialization scheme remains constant throughout training. While the addition itself will temporarily confuse existing units, this should make sure that newly added parameters are on the same scale as existing weights and thus trained in practice. Note that in principle, during the optimization of a task the weight distribution could shift significantly from its initial state. However, we do not encounter this potential issue in empirical experiments. Nevertheless, we point out that this currently under-explored topic requires separate future investigation in the context of model expansion.

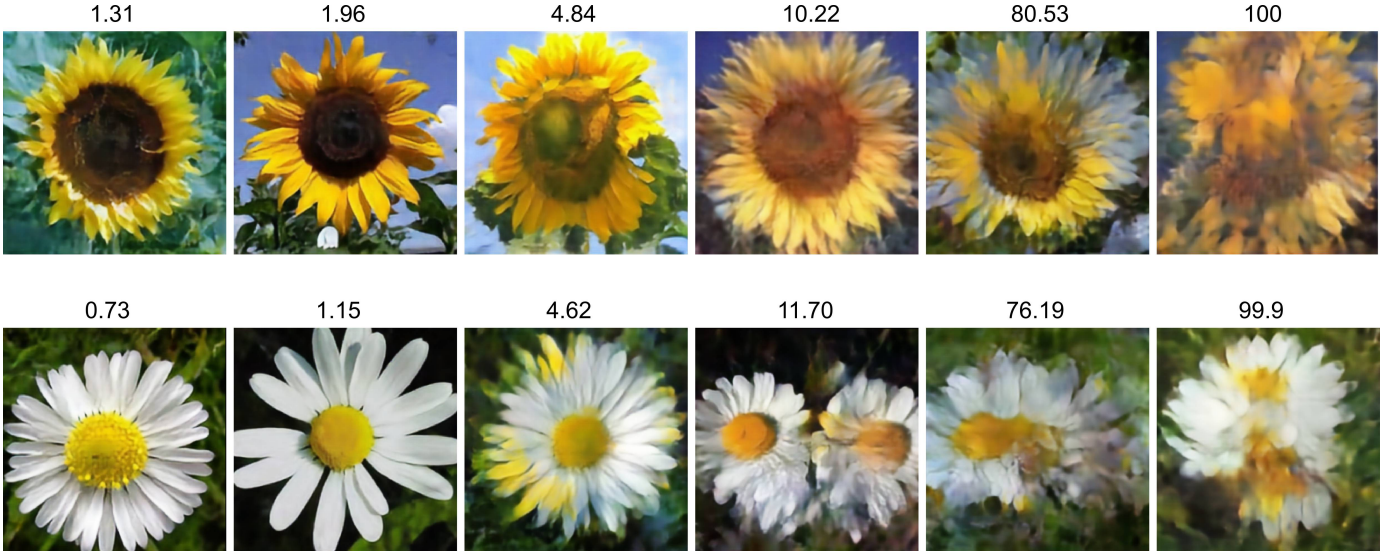
For the second implication, the β term of equation 1 is crucial. Here, the role of beta is to control the capacity of the information bottleneck and regulate the effective latent encoding overlap [43], which can best be summarized with a direct quote from the recent work of Mathieu et al. [44]: *"The overlap factor is perhaps best understood by considering extremes: too little, and the latents effectively become a lookup table; too much, and the data and latents do not convey information about each other. In either case, meaningfulness of the latent encodings is lost." (p. 4)*. This can be seen as under- or over-regularization by the prior of what is typically referred to as the aggregate posterior [45]:

$$q_{\boldsymbol{\theta}, t}(\boldsymbol{z}) = \mathbb{E}_{p_{\tilde{D}_t}(\tilde{\boldsymbol{x}})}\left[q_{\boldsymbol{\theta}, t}(\boldsymbol{z}|\tilde{\boldsymbol{x}})\right] \approx \frac{1}{\tilde{N}_t} \sum_{n=1}^{\tilde{N}_t} q_{\boldsymbol{\theta}, t}(\boldsymbol{z}|\tilde{\boldsymbol{x}}^{(n)}) \tag{5}$$

As an extension of this argument to our model, the necessity of linear class separation given $\boldsymbol{z}$ requires a suitable level of encoding overlap. This forms the basis for our open set recognition and respective improved generative replay for continual learning, which will be discussed in the following paragraphs. Example two-dimensional latent encodings for a continually trained MNIST [46] model with appropriate β are shown in figure 2. Here, we can see that the classes are cleanly separated in latent space, as enforced by the linear classification objective, and new classes can be accommodated continually. Further discussion on the choice of β can be found in the supplementary material.



(a) MNIST: 28×28 resolution classified as $c = 0$ (top left), $c = 3$ (top right), $c = 5$ (bottom left) and $c = 9$ (bottom right). Images were generated from the two-dimensional latent space visualized in figure 2.



(b) Flowers: 256×256 resolution classified as "sunflower" (top row) and "daisy" (bottom row). Images generated from a 60 dimensional latent space of deep wide residual models trained with introspection, as detailed in later experimental sections.

Fig. 3: Generated images $x \sim p_{\boldsymbol{\phi},t}(\boldsymbol{x}|\boldsymbol{z})$ with $\boldsymbol{z} \sim p(\boldsymbol{z})$ and their corresponding class c obtained from the classifier $p_{\boldsymbol{\xi},t}(y|\boldsymbol{z})$ together with their open set outlier percentage. Image quality degradation and class ambiguity can be observed with increasing outlier likelihood. Flower images have been compressed for side-by-side view.

3.3 Open set recognition and generative replay with statistical outlier rejection

Trained naively in above fashion, our model would suffer from accumulated errors with each successive iteration of generative replay, similar to current literature approaches. The main challenge is that high density areas under the prior $p(\boldsymbol{z})$ are not necessarily reflected in the structure of the aggregate posterior $q_{\boldsymbol{\theta},t}(\boldsymbol{z})$ [47]. Thus, generated data from low density regions of the latter does not generally correspond to encountered data instances. Conversely, data instances that fall into high density regions under the prior should not generally be considered as statistical inliers with respect to the observed data distribution.

Ideally, this challenge would be solved by modifying equations 1 and 2 by replacing the Gaussian prior in the KL-divergence with $q_{\boldsymbol{\theta},t}(\boldsymbol{z})$ and respectively sampling $z \sim q_{\boldsymbol{\theta},t-1}(\boldsymbol{z})$ for generative replay in equations 3 and 4. Even though using the aggregate posterior as the prior is the objective in multiple recent works, it can be challenging in high dimensions, lead to over-fitting and often comes at the expense of additional hyper-parameters [47], [48], [49]. To avoid finding an explicit representation for the multi-modal $q_{\boldsymbol{\theta},t}(\boldsymbol{z})$, we leverage our model's class disentanglement and draw inspiration from the EVT based OpenMax approach [9] in deep neural networks. However, instead of using knowledge about extreme distance values in penultimate layer activations to modify a Softmax prediction's confidence, we propose to apply EVT on the basis of the class conditional aggregate posterior. In this view, any sample can be regarded as statistically outlying if its distance to the classes' latent means is extreme with respect to what has been observed for the majority of correctly predicted data instances, i.e. the sample falls into a region of low density under the aggregate posterior and is less likely to belong to $p_{\tilde{D}}(\tilde{\boldsymbol{x}})$.

For convenience, let us introduce the indices of all correctly classified data instances at the end of task t as $m = 1, \ldots, \tilde{M}_t$. To construct a statistical meta-recognition model, we first obtain each class' mean latent vector for all correctly predicted seen data instances:

$$\bar{\boldsymbol{z}}_{c,t} = \frac{1}{|\tilde{M}_{c,t}|} \sum_{m \in \tilde{M}_{c,t}} \mathbb{E}_{q_{\theta,t}(\boldsymbol{z}|\tilde{\boldsymbol{x}}_t^{(m)})} [\boldsymbol{z}] \tag{6}$$

and define the respective set of latent distances as:

$$\Delta_{c,t} \equiv \left\{ f_d \left(\bar{\boldsymbol{z}}_{c,t}, \mathbb{E}_{q_{\theta,t}(\boldsymbol{z}|\tilde{\boldsymbol{x}}_t^{(m)})} [\boldsymbol{z}] \right) \right\}_{m \in \tilde{M}_{c,t}} \tag{7}$$

Here, f_d signifies a choice of distance metric. We proceed to model this set of distances with a per class heavy-tail

Weibull distribution $\boldsymbol{\rho}_{c,t} = (\tau_{c,t}, \kappa_{c,t}, \lambda_{c,t})$ on $\Delta_{c,t}$ for a given tail-size η. As these distances are based on the class conditional approximate posterior, we can thus bound the latent space regions of high density. The tightness of the bounds is characterized through η, that can be seen as a prior belief with respect to the outlier quantity assumed to be inherently present in the data distribution. The choice of f_d determines the nature and dimensionality of the obtained distance distribution. For our experiments, we find that the cosine distance and thus a univariate Weibull distance distribution per class seems to be sufficient.

Using the cumulative distribution function of this Weibull model $\boldsymbol{\rho}_t$ we can now estimate any sample's outlier probability:

$$\omega_{\boldsymbol{\rho},t}(\boldsymbol{z}) = \texttt{min}\left(1 - \exp\left(-\frac{|f_d\left(\bar{\boldsymbol{z}}_t, \boldsymbol{z}\right) - \boldsymbol{\tau}_t|}{\boldsymbol{\lambda}_t}\right)^{\boldsymbol{\kappa}_t}\right) \tag{8}$$

where the minimum returns the smallest outlier probability across all classes. If this outlier probability is larger than a prior rejection probability Ω_t, the instance can be considered as unknown as it is far away from all known classes. For a novel data instance, the outlier probability can be based on computation of the probabilistic encoder $\boldsymbol{z} \sim q_{\boldsymbol{\theta},\boldsymbol{t}}(\boldsymbol{z}|\boldsymbol{x})$ and a false overconfident classifier prediction avoided. Analogously, for the generative model, equation 8 can be used with $\boldsymbol{z} \sim p(\boldsymbol{z})$ and the probabilistic decoder only calculated for samples that are considered to be statistically inlying. This way, we can constrain the naive generative replay of equation 4 to the aggregate posterior, while avoiding the need to sample $\boldsymbol{z} \sim q_{\boldsymbol{\theta},t}(\boldsymbol{z})$ directly. Although this may sound detrimental to our method, it comes with the advantage of scalability to high dimensions. We further argue that the computational overhead for generative replay, both from sampling from the prior $\boldsymbol{z} \sim p(\boldsymbol{z})$ in large parallelized batches and computation of equation 8, is negligible in contrast to the much more computationally heavy deep probabilistic decoder or even the linear classifier, as the latter only need to be calculated for accepted samples. To give a visual illustration, we show examples of generated MNIST [46] and larger resolution flower images [50] together with their outlier percentage in figure 3.

4 Experiments and Analysis

Similar to recent literature [3], [6], [15], [16], [28], we consider the incremental MNIST [46] dataset, where classes arrive in groups of two, and corresponding versions of the FashionMNIST [51] and AudioMNIST dataset [52]. For the latter we follow the authors' procedure of converting the audio recordings into spectrograms. In addition to this class incremental setting, we also evaluate cross-dataset scenarios, where datasets are sequentially added with all of their classes and the model has to learn across modalities.

For a common frame of reference, we base both encoder and decoder architectures on 14-layer wide residual networks with a latent dimensionality of 60 [11], [12], [53], [54]. For the generative replay with statistical outlier rejection, we use an aggressive rejection rate of $\Omega_t = 0.01$ and dynamically set tail-sizes to 5% of seen examples per class. To avoid over-fitting, we add noise sampled from $\mathcal{N}(0, 0.25)$ to each input. This is preferable to weight regularization as it doesn't entail unrecoverable units that are needed to encode later tasks. We thus refer to our proposed model as Open-set Classifying Denoising Variational Auto-Encoder (OCDVAE), for which we have found a value of $\beta = 0.1$ to consistently work well, see discussion in the appendix. An important practical aspect is that we include normalizing terms into our previously introduced loss functions in order to have a set-up that is agnostic to dataset properties such as image resolution or task complexity that manifests in minimum required latent dimensionality. Specifically, we normalize the reconstruction loss by the spatial data dimension, i.e. dividing it by the number of pixels, and the KL divergence by the latent dimensionality. This way, we do not need to find a different value for beta if the latent dimensionality is altered or alternatively scaling the reconstruction loss' magnitude if the input size is increased. We empirically compare the following methods:

Dual Model: separate generative and discriminative variational models in analogy to the deep generative replay of Shin et al. [28].
EWC: elastic weight consolidation [16] for a purely discriminative model.
OCDVAE (ours): our proposed joint model with posterior based open set recognition and resulting statistical outlier rejection in generative replay.
CDVAE: the naive approach of generating from the prior distribution in our joint model. We include these results to highlight the effect of aggregate posterior to prior mismatch.
ISO: isolated learning, where all data is always present.
UB: upper-bound on achievable model performance by sequentially accumulating all data, given by equation 2.
LB: lower-bound on model performance when only the current task's data is available. No additional mechanism is in place and full catastrophic forgetting occurs.

Our evaluation metrics are inspired by previously proposed continual learning measures [55], [56]. In addition to overall accuracy $\alpha_{t,all}$, these metrics monitor forgetting by computing a base accuracy $\alpha_{t,base}$ for the initial task at increment t, while also gauging the amount of new knowledge that can be encoded by monitoring the accuracy for the most recent increment $\alpha_{t,new}$. We evaluate the quality of the generative models through classification accuracy as it depends on generated replay and a direct evaluation of pixel-wise reconstruction losses is not necessarily coupled to classification accuracy or retention thereof. However, we provide a detailed analysis of reconstruction losses for all tasks, as well as KL divergences for all experiments in the supplementary material.

To provide a fair comparison of achievable accuracy, all above approaches are trained to converge on each task using the Adam optimizer [57]. We repeat all experiments five times to assess statistical consistency. The full hyper-parameter specification can be found in the supplementary material. There, we also provide the quantitative continual learning results for all experiments with a 2-hidden layer and 400 unit multi-layer perceptron [56], as the WRN architecture could be argued to be excessively large for simpler datasets such as MNIST, in particular with the parameters of the

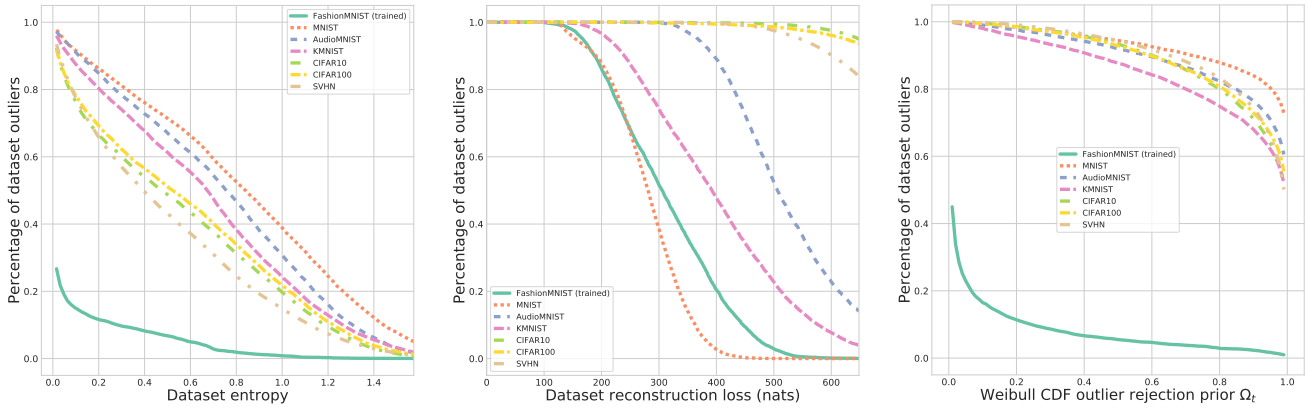


Fig. 4: Trained FashionMNIST OCDVAE evaluated on unknown datasets. All metrics are averaged over 100 approximate posterior samples per data point. (Left) Classifier entropy values are insufficient to separate most of unknown from the known task's test data. (Center) Reconstruction loss allows for a partial distinction. (Right) Our posterior based open set recognition considers the large majority of unknown data as statistical outliers across a wide range of rejection priors Ω_t.

TABLE 1: Results for continual learning across datasets averaged over 5 runs, baselines and the reference isolated learning scenario for FashionMNIST (F) $\rightarrow$ MNIST (M) $\rightarrow$ AudioMNIST (A) and the reverse order. α_T indicates the respective accuracy at the end of the last increment $T = 3$.

Cross-dataset		α_T(%) (T=3)		
		base	new	all
F-M-A	**CDVAE ISO**			94.95
	CDVAE UB	89.10	97.88	95.00
	CDVAE LB	00.00	98.12	22.70
	EWC	22.85 $\pm$ 0.294	93.31 $\pm$ 0.138	43.42 $\pm$ 0.063
	Dual Model	81.89 $\pm$ 0.104	96.78 $\pm$ 0.067	91.75 $\pm$ 0.064
	CDVAE	57.70 $\pm$ 4.480	96.73 $\pm$ 0.235	81.10 $\pm$ 1.769
	OCDVAE	80.11 $\pm$ 2.922	97.63 $\pm$ 0.042	91.13 $\pm$ 1.045
A-M-F	**CDVAE ISO**			94.95
	CDVAE UB	97.17	89.16	94.91
	CDVAE LB	00.00	89.72	34.51
	EWC	3.420 $\pm$ 0.026	87.54 $\pm$ 0.214	45.42 $\pm$ 0.731
	Dual Model	66.82 $\pm$ 0.337	89.15 $\pm$ 0.050	87.70 $\pm$ 0.102
	CDVAE	79.74 $\pm$ 2.431	88.50 $\pm$ 0.126	89.46 $\pm$ 0.600
	OCDVAE	94.53 $\pm$ 0.283	89.53 $\pm$ 0.367	94.06 $\pm$ 0.156

network being on a similar scale as the dataset itself. Note that all drawn conclusions remain the same independent of the architecture used and the main difference is a mild degradation in performance as an expected consequence of the less complex architecture. All models were trained on a single GTX 1080 GPU and our code will be publicly available.

4.1 Learning across datasets in an open world

Achieved accuracies for continual learning across datasets are summarized in table 1. In general the upper-bound values are almost identical to isolated learning. Similarly, the new task's metrics are negligibly close, as the WRN architecture ensures enough capacity to encode new knowledge. In contrast to EWC that is universally unable to maintain knowledge in a single-head classifier, as also previously observed by [3], [56], approaches based on generative replay are able to partially retain information. Yet they accumulate errors due to samples generated from low density regions. This is noticeable for both the dual model approach, with a separate VAE and discriminative model, and more heavily so for the naive CDVAE, where the structure of $q_{\boldsymbol{\theta},t}(\boldsymbol{z})$ is further affected by the discriminator. However, our proposed OCDVAE model overcomes this issue to a considerable degree, rivalling and improving upon the separately trained models.

Apart from these classification accuracies, we also quantitatively analyze the models' ability to distinguish unknown tasks' data from data belonging to known tasks. Here, the challenge is to consider all unseen test data of already trained tasks as inlying, while successfully identifying 100 % of unknown datasets as outliers. For this purpose, we evaluate models after training on one dataset on its respective test set, the remaining tasks' datasets and additionally the KMNIST [58], SVHN [59] and CIFAR [60] datasets.

We compare and contrast three criteria that could be used for open set recognition: classifier predictive entropy, reconstruction loss and our proposed latent based EVT approach. We approximate the expectation with 100 variational samples from the approximate posterior per data point, i.e. marginalising the latent variable $\boldsymbol{z}$ with Monte Carlo samples from $q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$. Figure 4 shows the three criteria and respective percentage of the total dataset being considered as outlying for the OCDVAE model trained on FashionMNIST. In consistence with [36], we can observe that use of reconstruction loss can sometimes distinguish between the known tasks' test data and unknown datasets, but results in failure for others. In the case of classifier predictive entropy, depending on the exact choice of entropy threshold, generally only a partial separation can be achieved. Furthermore, both of these criteria pose the additional challenge of results being highly dependent on the choice of the precise cut-off value. In contrast, the test data from the known tasks is regarded as inlying across a wide range of rejection priors Ω_t and the majority of other datasets is consistently regarded as outlying by our proposed open set mechanism.

We provide quantitative outlier detection accuracies in table 2. Here, a 5% validation split is used to determine the respective value at which 95% of the validation data is considered as inlying before using these priors to determine

TABLE 2: Test accuracies and outlier detection values of the joint OCDVAE and dual model (VAE and separate deep classifier, denoted as "CL + VAE") approaches when considering 95 % of known tasks' validation data is inlying. Percentage of detected outliers is reported based on classifier predictive entropy, reconstruction loss and our posterior based EVT approach, averaged over 100 $\boldsymbol{z} \sim q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ samples per data-point respectively. Note that larger values are better, except for the test data of the trained dataset, where ideally 0% should be considered as outlying.

Outlier detection at 95% validation inliers (%)				MNIST	Fashion	Audio	KMNIST	CIFAR10	CIFAR100	SVHN
Trained	Model	Test acc.	Criterion							
FashionMNIST	Dual, CL + VAE	90.48	Class entropy	74.71	5.461	69.65	77.85	24.91	28.76	36.64
			Reconstruction	5.535	5.340	64.10	31.33	99.50	98.41	97.24
			Latent EVT	96.22	5.138	93.00	91.51	71.82	72.08	73.85
	Joint, OCDVAE	90.92	Class Entropy	66.91	5.145	61.86	56.14	43.98	46.59	37.85
			Reconstruction	0.601	5.483	63.00	28.69	99.67	98.91	98.56
			Latent EVT	96.23	5.216	94.76	96.07	96.15	95.94	96.84
MNIST	Dual, CL + VAE	99.40	Class entropy	4.160	90.43	97.53	95.29	98.54	98.63	95.51
			Reconstruction	5.522	99.98	99.97	99.98	99.99	99.96	99.98
			Latent EVT	4.362	99.41	99.80	99.86	99.95	99.97	99.52
	Joint, OCDVAE	99.53	Class entropy	3.948	95.15	98.55	95.49	99.47	99.34	97.98
			Reconstruction	5.083	99.50	99.98	99.91	99.97	99.99	99.98
			Latent EVT	4.361	99.78	99.67	99.73	99.96	99.93	99.70
AudioMNIST	Dual, CL + VAE	98.53	Class entropy	97.63	57.64	5.066	95.53	66.49	65.25	54.91
			Reconstruction	6.235	46.32	4.433	98.73	98.63	98.63	97.45
			Latent EVT	99.82	78.74	5.038	99.47	93.44	92.76	88.73
	Joint, OCDVAE	98.57	Class entropy	99.23	89.33	5.731	99.15	92.31	91.06	85.77
			Reconstruction	0.614	38.50	3.966	36.05	98.62	98.54	96.99
			Latent EVT	99.91	99.53	5.089	99.81	100.0	99.99	99.98

outlier counts for the known tasks' test set as well as other datasets. We provide this evaluation for both our joint model, as well as separate discriminative and generative models. While MNIST seems to be an easy to identify dataset for all approaches, we can make two major observations:

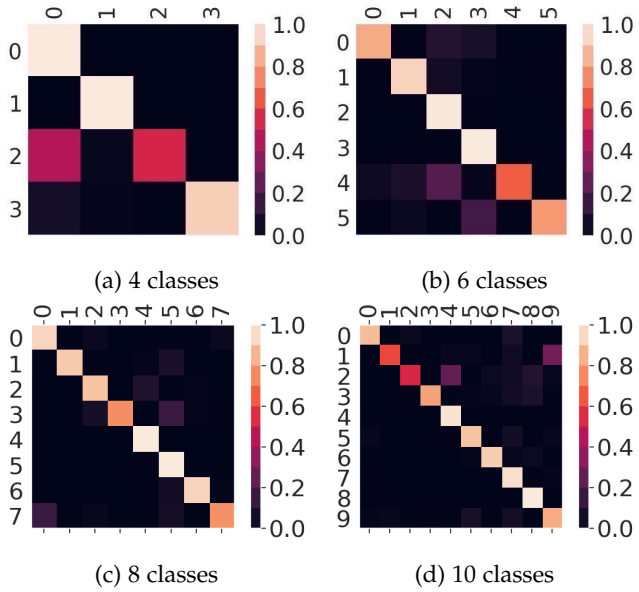
1) The latent based EVT approach outperforms the other criteria, particularly for the OCDVAE where a near perfect open set detection can be achieved.
2) Even though we can apply EVT to purely discriminative models, the joint OCDVAE model consistently exhibits more accurate outlier detection. We hypothesize that this is due to the joint model also optimizing a variational lower bound to the data distribution $p(\boldsymbol{x})$ in addition to taking into account labels.

We provide figures similar to figure 4 for all models reported in table 2 in the supplementary material.

Naively one might at this point be tempted to argue that the trained weights of the individual deep neural network encoder layers are still deterministic and the failure of predictive entropy as a measure for unseen unknown data could thus primarily be attributed to uncertainty not being expressed adequately. Placing a distribution on the weights would then be expected to resolve this issue. Although it has previously been argued that this is not the case [8], we further repeat all of our quantitative open set experiments by treating the model weights as the random variable being marginalised through the use of MC-Dropout [35]. Whereas some improvements upon the presented results of this section are noticeable, they are overall negligible with respect to observed patterns, the two major observations formulated in above list, and drawn conclusions. The corresponding table containing the quantitative Monte-Carlo Dropout results have accordingly been moved to the supplementary material.

TABLE 3: Results for class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for the three datasets. α_T indicates the respective accuracy at the end of the last increment $T = 5$.

Class-incremental		α_T(%) (T=5)		
		base	new	all
FashionMNIST	**CDVAE ISO**			89.54
	CDVAE UB	92.20	97.50	89.24
	CDVAE LB	00.00	99.80	19.97
	EWC	00.17 $\pm$ 0.076	99.60 $\pm$ 0.023	20.06 $\pm$ 0.059
	Dual Model	94.26 $\pm$ 0.192	93.55 $\pm$ 0.708	63.21 $\pm$ 1.957
	CDVAE	39.51 $\pm$ 7.173	96.92 $\pm$ 0.774	58.82 $\pm$ 2.521
	OCDVAE	60.63 $\pm$ 12.16	96.51 $\pm$ 0.707	69.88 $\pm$ 1.712
MNIST	**CDVAE ISO**			99.45
	CDVAE UB	99.57	99.10	99.29
	CDVAE LB	00.00	99.85	20.16
	EWC	00.45 $\pm$ 0.059	99.58 $\pm$ 0.052	20.26 $\pm$ 0.027
	Dual Model	97.31 $\pm$ 0.489	98.59 $\pm$ 0.106	96.64 $\pm$ 0.079
	CDVAE	19.86 $\pm$ 7.396	99.00 $\pm$ 0.100	64.34 $\pm$ 4.903
	OCDVAE	92.35 $\pm$ 4.485	99.06 $\pm$ 0.171	93.24 $\pm$ 3.742
AudioMNIST	**CDVAE ISO**			97.75
	CDVAE UB	98.42	98.67	97.87
	CDVAE LB	00.00	100.0	20.02
	EWC	00.11 $\pm$ 0.007	99.41 $\pm$ 0.207	19.98 $\pm$ 0.032
	Dual Model	61.58 $\pm$ 0.747	89.41 $\pm$ 0.691	47.42 $\pm$ 1.447
	CDVAE	59.36 $\pm$ 7.147	84.93 $\pm$ 6.297	81.49 $\pm$ 1.944
	OCDVAE	79.73 $\pm$ 4.070	89.52 $\pm$ 6.586	87.72 $\pm$ 1.594



(a) 4 classes (b) 6 classes

(c) 8 classes (d) 10 classes

Fig. 5: AudioMNIST confusion matrices for the incrementally learned OCDVAE model. When adding classes two and three the model experiences difficulty in classification, however is able to overcome this challenge by exhibiting backward transfer when later learning classes four and five. Similarly, classes four and five are then retrospectively improved through the addition of classes six and seven. It is also observable how forgetting of the initial classes is limited.

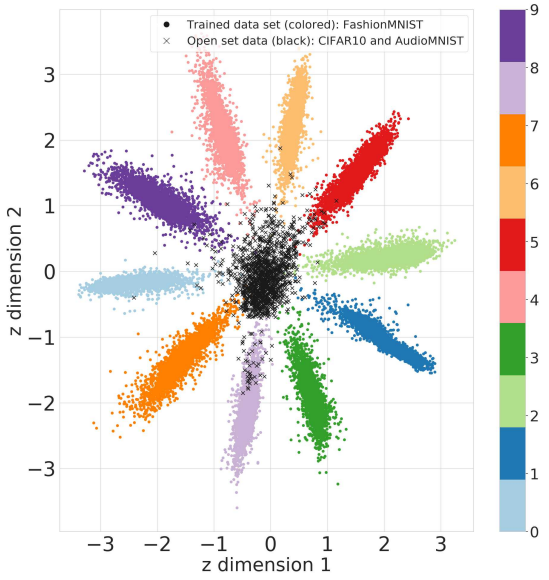


Fig. 6: Latent space visualization for OCDVAE with a two-dimensional latent space trained on FashionMNIST. In addition to the learned classes, embedded data points for unseen unknown classes belonging to AudioMNIST and CIFAR10 are shown. The latter can be observed to be separable by their distance to trained concepts.

4.2 Learning classes incrementally

We show results in analogy to table 1 for the class incremental scenario in table 3. With the exception of MNIST, where the dual model approach fares well, a similar pattern as before can be observed and our proposed OCDVAE approach significantly outperforms all other methods. Interestingly, as a result of using a single model across tasks, we observe backward transfer in some experiments. We dedicate the next subsection to this desirable phenomenon and tie its forthcoming discussion to potential limitations of regularization based continual learning methods.

4.3 Backward transfer and the limits of regularization

The existing tasks' representations are typically exploited in the acquisition of a new task's information in continual learning, transfer learning and all other scenarios that formulate some kind of incremental learning problem. However, the concept of backward transfer is generally less deliberated. It describes the reverse phenomenon where introduction of a new task leads to learning of representations that retrospectively improve former tasks. We observe this behavior in multiple of our experiments, whose detailed numerical account together with examples of all generated images for all increments $t = 1, \ldots, 5$ can be found in the supplementary material. For the purpose of the following discussion, it is sufficient to single out one particularly noteworthy example of backward transfer. Figure 5 highlights the occurrence of retrospective improvement for class-incremental learning with our OCDVAE model on the AudioMNIST dataset, as quantitatively presented in tables 16 and 19. Here, the addition of two new classes (four and five) at the end of the second increment leads to an improvement in the classification performance on class two, as illustrated by the confusion matrices. Analogously, subsequent inclusion of the additional classes six and seven enhances the classes of the second task increment, even if none of the former tasks' real data is present any longer. We point out that this is continual learning desideratum can only emerge from having a single model with a single classification head and alleviated catastrophic forgetting through mechanisms different from heavy regularization. By definition, obtaining retrospective improvement through regularization is unlikely, if not entirely unachievable. This is because continual learning through regularization encourages the model to reproduce solutions for previous tasks by maintaining the parameters or upholding a specific prediction, e.g. through knowledge distillation [17], [63], [64], [65]. The focus therefore lies on avoiding model deterioration, without the possibility of surpassing previously reached performance. Although cases were backward transfer may not always be necessary are conceivable, e.g. if a task's performance requirements are already met from the start, this inability for retroactive correction can be one major drawback of regularization methods.

At this point, we emphasize that the goal of this question is not to altogether question the merit of prior works that have made use of regularization techniques. Instead, we would simply like to raise awareness that there exist contexts in which regularization techniques might be helpful and on the contrary, settings, where use of regularization may be in direct opposition to the desired goals. Apart from task sequences where backward transfer can be of essence, another context in which current continual learning

regularization methods may be antagonistic is the objective of open set recognition. In particular, we posit that commonly employed regularization techniques and the ability to recognize the open set are interdependent. To specify this statement, figure 6 shows another visualization of a trained model's two-dimensional latent embedding for FashionMNIST, similar to the MNIST visualization of figure 2. However, here we have also included the probabilistic encoder's mapping of previously unseen unknown classes from the AudioMNIST and CIFAR10 datasets. On the one hand, it is observable how the corresponding latent values have large distance to the clusters belonging to the learned classes, painting an intuitive two-dimensional picture for the earlier demonstrated success of our framework in open set recognition in high-dimensional latent spaces. On the other hand, the large majority of the unseen unknown data points fall into a central cluster. If we now desire to incorporate these currently unseen unknown classes by including them into the next continual learning task, this single cluster of unseen unknowns will need to be divided in order for the individual classes to be discriminable. In analogy to the visualized rearrangement of the latent space over time in figure 2, the aggregate posterior thus need to be given the flexibility to experience ample change. If despite of this requirement a regularization approach regularizes the current aggregate posterior $q_{\boldsymbol{\theta},t}(\boldsymbol{z})$, e.g. by replacing the Gaussian prior with the former tasks' aggregate posterior $KL(q_{\boldsymbol{\theta},t}(\boldsymbol{z}|\boldsymbol{x}^{(n)}) \mid\mid q_{\boldsymbol{\theta},t-1}(\boldsymbol{z}))$ in equation 1 such as proposed in the variational continual learning (VCL) [30], this may not be possible. A similar argument provides the rationale behind the earlier empirically demonstrated failure of EWC, where restrictions on updates to the probabilistic encoder's parameters hinder the disambiguation of new classes or conversely discount the solution for previous tasks.

Before we continue to showcase ways in which our proposed framework can naturally be scaled to high-resolution color images, we would like to give credit to related works that have purposely not been included in our experimental comparison for an entirely different reason. These approaches are naturally synergistic reporting them separately in a standalone quantitative comparison could mislead the reader. They primarily belong to the category of *exemplar/core set rehearsal*. Prominent examples are iCarl [22], gradient episodic memory [55], FearNet [61], Variational Continual Learning [30] or CLEAR [62]. The retention and rehearsal of real data can always be a valid strategy to address the challenge of continual learning, if memory is of little concern. The problem is then re-framed to the discovery of suitable data subset selection schemes. The latter can naturally be integrated into our proposed framework by devising mechanisms to select data subsets which best approximate the aggregate posterior of the entire dataset.

5 Improving the generative model: scaling through autoregression and introspection

At the time of their initial introduction, it was notorious that variational autoencoders produce blurry examples and were associated with an inability to scale to more complex high-resolution color images. This is in contrast to their prominent generative counterparts, the generative adversarial network [29]. Although this stigma perhaps still holds until today, there has been many successful recent efforts to address this challenge. In what is supposed to constitute a final outlook for our work, we thus empirically investigate the choice of generative model and optionally improve the probabilistic decoding with the help of two promising research directions: autoregression [10], [11], [12] and introspection [13], [14]. The commonality between these approaches is their aim to overcome the limitations of independent pixel-wise reconstructions. We will briefly summarize these generative extensions, empirically show their advantage by revisiting our previous experiments, before continuing to demonstrate our framework's efficacy on high resolution color images.

TABLE 4: PixelVAE based continual learning approaches averaged over 5 runs in analogy to tables 1 and 3.

Class-incremental		α_T(%) (T=5)		
		base	new	all
Fashion	**Dual Pix Model**	60.04 ± 5.151	98.85 ± 0.141	72.41 ± 2.941
	PixCDVAE	47.83 ± 13.41	97.91 ± 0.596	63.05 ± 1.826
	PixOCDVAE	74.45 ± 2.889	98.63 ± 0.176	80.85 ± 0.721
MNIST	**Dual Pix Model**	98.04 ± 1.397	97.31 ± 0.575	96.52 ± 0.658
	PixCDVAE	56.53 ± 4.032	96.77 ± 0.337	83.61 ± 0.927
	PixOCDVAE	97.44 ± 0.785	98.63 ± 0.430	96.84 ± 0.346
Audio	**Dual Pix Model**	64.60 ± 8.739	98.18 ± 0.885	75.50 ± 3.032
	PixCDVAE	29.94 ± 18.47	97.00 ± 0.520	63.44 ± 5.252
	PixOCDVAE	75.25 ± 10.18	99.43 ± 0.495	90.23 ± 1.139
Cross-dataset		α_T(%) (T=3)		
		base	new	all
F-M-A	**Dual Pix Model**	82.88 ± 0.116	97.23 ± 0.212	92.16 ± 0.061
	PixCDVAE	56.44 ± 1.831	97.50 ± 0.184	80.76 ± 0.842
	PixOCDVAE	81.84 ± 0.212	97.75 ± 0.169	91.76 ± 0.212
A-M-F	**Dual Pix Model**	71.58 ± 2.536	88.76 ± 0.255	88.61 ± 0.547
	PixCDVAE	49.38 ± 2.256	88.54 ± 0.042	82.18 ± 0.672
	PixOCDVAE	91.90 ± 0.282	89.91 ± 0.177	93.82 ± 0.354

5.1 Improvements through autoregressive decoding

In essence, autoregressive models improve the probabilistic decoder by introducing a spatial conditioning of each scalar output value on the previous ones, in addition to conditioning on the latent variable:

$$p(\boldsymbol{x}|\boldsymbol{z}) = \prod_i p(x_i|x_1, \ldots, x_{i-1}, \boldsymbol{z}) \tag{9}$$

In an image, generation thus needs to proceed pixel by pixel and is commonly referred to as PixelVAE [11]. This conditioning is generally achieved by providing the input to the decoder during training, i.e. including a skip path that bypasses the probabilistic encoding. A concurrent introduction of autoregressive VAEs has thus coined this model "lossy" [12]. This is because local information can now be modelled without access to the latent variable and only the global information will be encoded in $\boldsymbol{z}$.

We repeat our previously shown continual learning experiments with three additional appended autoregressive decoder layers, each with a kernel size of 7×7 and 60 channels, following the experimental set-up of the original PixelVAE. We also follow the authors' recommendation to

train the decoder using a 256-way Softmax and treating the reconstruction as classification in practice. Results corresponding to tables 1 and 3 for these pixel models are shown in table 4. While we can observe that the introduction of the autoregressive decoder generally further alleviates catastrophic forgetting, it does significantly more so for our proposed approach.

5.2 Introspection and adversarial training

Although the earlier shown accuracies of generative replay with autoregression are assuring, autoregressive sampling comes with a major caveat. When attempting to operate on larger data, the computational complexity of the pixel by pixel data creation procedure grows in direct proportion to the input dimensionality. With increasing input size, the repeated calculation of the autoregressive decoder layers can thus rapidly render the generation required for optimization of equation 3 practically infeasible. A promising alternative perspective towards autoencoding beyond pixel similarities is to leverage the insights obtained from GANs. To this matter, Larsen et al. [66] have proposed a hybrid model called VAEGAN. Here, the crucial idea is to append a GAN style adversarial discriminator to the variational autoencoder. This yields a model that promises to overcome a conventional GAN's mode collapse issues, as the VAE is responsible for the rich encoding, while letting the added discriminator judge the decoder's output based on perceptual criteria rather than individual pixel values. The more recent IntroVAE [13] and adversarial encoder generator networks [14] have subsequently come to the realization that this doesn't necessarily require the auxiliary real-fake discriminator, as the VAE itself already provides strong means for discrimination, namely its probabilistic encoder. We leverage this idea of introspection for our framework, as it doesn't require any architectural or structural changes beyond an additional term in the loss function.

For sake of brevity we denote the probabilistic encoder through their parameters $\boldsymbol{\phi}$ and decoder $\boldsymbol{\theta}$ in the following equations. Training our model with introspection is then equivalent to adding the following two terms to our previously formulated loss function:

$$\begin{aligned} &\mathcal{L}_{IntroCDVAE\_Enc} = \\ &\mathcal{L}_{CDVAE} - \beta \left[m - KL(\boldsymbol{\theta}(\boldsymbol{\phi}(\boldsymbol{z})) \mid\mid p(\boldsymbol{z})) \right]^{+} \end{aligned} \tag{10}$$

and

$$\mathcal{L}_{IntroCDVAE\_Dec} = \mathcal{L}_{Rec} - \beta KL(\boldsymbol{\theta}(\boldsymbol{\phi}(\boldsymbol{z})) \mid\mid p(\boldsymbol{z})) \tag{11}$$

Here, $\mathcal{L}_{CDVAE}$ corresponds to the full loss of equation 1 and $\mathcal{L}_{Rec}$ corresponds to the reconstruction loss portion: $\mathbb{E}_{q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}^{(n)})}[\log p_{\boldsymbol{\phi}}(\boldsymbol{x}^{(n)}|\boldsymbol{z})]$. In above equations, we have followed the original authors proposal to include a positive margin m, with $[\cdot]$ denoting $max(0, \cdot)$. This hinge loss formulation serves the purpose of empirically limiting the encoder's reward to avoid a too massive gap in a min-max game of above competing KL terms. Aside from the regular loss that encourages the encoder to match the approximate posterior to the prior for real data, the encoder is now further driven to maximize the deviation from the posterior to the prior for generated images. Conversely, the decoder is encouraged to "fool" the encoder into producing a posterior distribution that matches the prior for these generated images. The optimization is conducted jointly. In comparison with a traditional VAE, this can thus be seen as training in an adversarial like manner, without necessitating additional discriminative models. As such, introspection fits naturally into our proposed framework and no further changes are required.

Before we proceed with demonstrating the empirical value, we note that the original authors of IntroVAE have introduced additional weighting terms in front of the reconstruction loss, in order to drastically lower its magnitude, and the added KL divergence. We have observed that the former is simply due to lack of normalization with respect to input width and height and hence the reconstruction loss growing proportionally with the spatial input size, whereas the KL divergence typically does not reflect this behavior for a fixed-size latent space. Given that we have included this normalization in our practical experimentation, we have found this additional hyper-parameter to be unnecessary. The other hyper-parameter to weight the added adversarial KL divergence term is essentially equivalent to the already introduced beta, alas without our motivation in earlier sections, but simply as a heuristic to not overpower the reconstruction loss.

5.3 Incrementally learning high resolution flowers

In this section, we empirically demonstrate the efficacy of generative modelling advances for high resolution natural data and respective improvements by using our proposed open set aware approach. For this purpose, we continual learn five types of flowers, in analogy to the experiment conducted in the recent Lifelong GAN [65]. Whereas the latter makes use of lower resolutions, we let the resolution remain at 256×256 pixels to demonstrate application of our approach to high-resolution. Apart from the high resolution, this scenario is interesting for two further reasons: the dataset contains less than 100 images per class and the classes are introduced one by one in continual training. This introduction of a single class makes a multi-head approach unrealisable, and thus a large portion of previously proposed approaches based on task labels and regularization, infeasible. The small-sample scenario also underlines that deep generative models can be trained without massive amounts of data. Due to the larger resolution we employ a deeper variant of our previously used 14-layer WRN architecture. In addition to the three convolution blocks that comprise a total of 12 layers, three further blocks are added, resulting in a 26 layer architecture that down-samples the input by an extra factor of eight across the added stages. This way, the encoded spatial dimensionality that precedes the 60 dimensional latent space is the same for WRN-14 experiments on 32×32 resolution and this section's experiments based on a WRN-26 and 256×256 resolution. We use a batch size of 32 and let all other hyper-parameters remain the same as described for previous experiments. The only exception is the amount of epochs, which we increase to 2000 per task in order to reach a significant amount of update iterations as a result of the small dataset size. In analogy to previous sections we report results as the average over multiple runs, with the exception of the autoregressive models. As the latter

had to be trained on multiple NVIDIA V100 GPUs over the course of three weeks for a single experiment, we report a single run. We did not repeat this experiment as below results are believed to sufficiently demonstrate limitations and are notably surpassed by the computationally favorable introspection model.

5.3.1 Denoising and the choice of perturbation

In the previous continual learning experiments, the introduced denoising acted as one way to avoid over-fitting, akin to the use of data augmentation. However, the choice of noise distribution can have an additional, very different purpose. Recall that in a wider sense, "denoising" refers to the concept of introducing an arbitrarily sampled perturbation that is added to the input, but needs to be discounted in reconstruction with respect to the original unperturbed data instance. This perturbation doesn't necessarily need to take on the earlier introduced pixel-wise noise from a Gaussian or Uniform distribution to alter each pixel independently. If our primarily interest lies in maintaining the discriminative performance of our model and less so on the visual quality of the generated data, we can take advantage of the perturbation distribution as means to encode our prior knowledge of common generative pitfalls. For example, in our specific context, it is well known that a traditional VAE without further advances commonly fails to generate non-blurry, crisp images. However, we can include and work around this belief by letting the denoising assume the form of deblurring, e.g. by stochastically adding a variety of Gaussian blurs to subsets of inputs. Even though the decoder is ultimately still encouraged to remove this blur and reconstruct the original clean image, the encoder is now inherently required to learn how to manage blurry input. It is encouraged to build up a natural invariance to our choice of perturbation. In the context of maintaining a classifier with generative replay, to an extend it should then no longer be a strict requirement to replay locally detailed crisp images, as long as the information required for discrimination is present.

5.3.2 Results and discussion

Figure 7 shows the quantitative accuracy of the evaluated methods and the continual learning upper and lower-bound. As usual, the lower-bound corresponds to predicting just the present class correctly and full catastrophic forgetting occurring for all previously seen concepts. The upper-bound shows an expected gradual decay with increased amount of classes. For all introduced model variants, we can observe significant improvement over baseline versions (dashed lines) with the introduction of our open set method (solid lines). As expected, the plain OCDVAE model is further substantially outperformed by the introspective model. The latter closely mirrors the upper-bound performance and starts to deviate only after multiple repetitions of generative replay. Although the open set aware generation also generally enhances the autoregressive baseline, the autoregressive PixOCDVAE, perhaps to the readers surprise, fares comparably much worse than the OCDVAE or IntroOCDVAE counterparts. We can observe the empirical rationale for this in a qualitative illustration of select generated samples for each continual learning step, provided in figure 8. For the PixOCDVAE

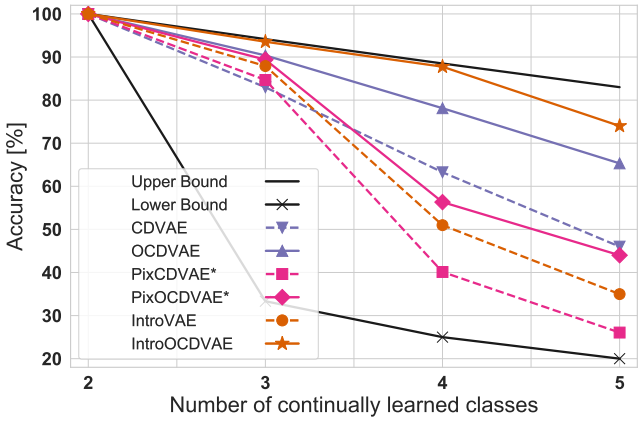


Fig. 7: Continual learning accuracies for flowers at 256×256 resolution to demonstrate how generative modelling advances enable scaling of our framework. Pairs of colored lines show respective improvements of our proposed aggregate posterior constrained generative replay (solid lines) over the open set unaware baselines (dashed lines). Whereas every model surpasses the lower-bound and thus to an extent alleviates catastrophic forgetting, our proposed framework in conjunction with introspection clearly beats the other contestants and approaches the upper-bound achievable accuracy. An accompanying qualitative illustration of generated images is provided in figure 8.

we can see that the initial task's generative replay is locally consistent, which is reflected in the quantitative accuracy values for the first tasks being almost indistinguishable from the other models. However, starting from the second cycle of generative replay, the conditioning of equation 9 seems to lose long-term correlations and an increasing amount of the image is filled with noise with each further step. In a visual comparison between IntroVAE and IntroOCDVAE, we again observe that ambiguous interpolations rapidly take over without constraining generative replay to aggregate posterior inliers, recall figure 4.

For OCDVAE, we observe that all images are blurry from the start. Even though the classes are distinguishable, this blur is amplified over time. The respectively very high accuracy of fig 7 can be attributed to the deblurring objective, where the encoder's hypothesized blur invariance largely compensates the model's inability to generate detailed examples. As a result, the accuracy gap between OCDVAE and IntroOCDVAE is rather small, despite what we as humans would perhaps initially expect from the visually less pleasing images. Correspondingly, when the deblurring is removed, we observe major drops in the final OCDVAE accuracy of up to 15%, with negligible degradation of reported values for the highly detailed introspection images. As a final remark, we note that Lifelong GAN [65] and MeRGAN [64] have conducted similar experiments on the flower dataset. We did not explicitly include results for the latter for two reasons. First, at this stage, it should be clear that an "either or" comparison is deceptive, as VAEs and GANs can go hand in hand to benefit each other. Second, these works have conducted experiments at a lower resolution and we were simply unable to reach accuracies at higher resolution

Fig. 8: Generated 256×256 flower images for various continually trained models. Images have been selected to provide a qualitative intuition behind the quantiative results of figure 7. The unmodified OCDVAE appears to suffer from the limitations of a traditional VAE and generates blurry images, although performs remarkably well in terms of quantitative classification. Its open set unaware counterpart CDVAE deteriorates similarly to earlier experiments due to the generation of ambiguous samples from low density areas outside the aggregate posterior. PixOCDVAE is initially competitive but rapidly loses long-range correlations of the autoregressive conditioning, resulting in increasingly noisy images. Introspection significantly increases the image detail, albeit still degrades considerable due to ambiguous interpolations. This is again resolved by combining introspection with our proposed posterior based EVT approach, where image quality is retained across multiple generative replay steps. Images have been compressed for a side-by-side view.

that would do the method justice, without resorting to substantial hyperparameter and architecture tuning. We have thus decided in favor of showing higher resolution experiments in contrast to a comparison on more heavily down-sampled images. Depending on the precise setup, MeRGAN and Lifelong GAN have been reported to result in final accuracies between 60% and 85% respectively [65], values that are generally similar to the ones reported in 7. However, note that the former achieves this accuracy by keeping a complete model copy at all times, whereas the latter makes use of auxiliary data and augmentation. This is in addition to both of these works requiring a separately trained deep discriminative model to solve the classification task. Neither of them considers the challenge of open set recognition, where the uniqueness of our work lies, and treats the problem in a closed world. With this in mind, we encourage future replay based continual learning to further explore generative modelling advances and their hybrid combinations, while keeping in mind that continual learning goes beyond subjective visual generation quality and measuring catastrophic forgetting.

6 Conclusion

We have proposed a probabilistic approach to unify the prevention of catastrophic forgetting with open set recognition based on variational inference in continual learning. Using a single model that combines a shared probabilistic encoder with a generative model and an expanding linear classifier, we have introduced EVT based bounds to the approximate posterior. The derived open set recognition and corresponding generative replay with statistical outlier rejection have been shown to achieve compelling results in both task incremental as well as cross-dataset continual learning across image and audio modalities, while being able to distinguish seen from unseen data. Our approach readily benefits from recent generative modelling techniques, which has been empirically demonstrated in the context of high resolution flower images. We expect future work to explore more natural synergies with further generative modelling advances and investigate a range of practical applications.

References

[1] M. McCloskey and N. J. Cohen, "Catastrophic Interference in Connectionist Networks : The Sequential Learning Problem," *Psychology of Learning and Motivation - Advances in Research and Theory*, vol. 24, no. C, pp. 109–165, 1989.

[2] O. Matan, R. Kiang, C. E. Stenard, and B. E. Boser, "Handwritten Character Recognition Using Neural Network Architectures," *4th USPS Advanced Technology Conference*, vol. 2, no. 5, pp. 1003–1011, 1990.

[3] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual Lifelong Learning with Neural Networks: A Review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[4] A. Graves, "Practical variational inference for neural networks," *Neural Information Processing Systems (NeurIPS)*, 2011.

[5] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *International Conference on Learning Representations (ICLR)*, 2013.

[6] S. Farquhar and Y. Gal, "A Unifying Bayesian View of Continual Learning," *Neural Information Processing Systems (NeurIPS) Bayesian Deep Learning Workshop*, 2018.

[7] A. Achille, T. Eccles, L. Matthey, C. P. Burgess, N. Watters, A. Lerchner, and I. Higgins, "Life-Long Disentangled Representation Learning with Cross-Domain Latent Homologies," *Neural Information Processing Systems (NeurIPS)*, 2018.

[8] T. E. Boult, S. Cruz, A. Dhamija, M. Gunther, J. Henrydoss, and W. Scheirer, "Learning and the Unknown : Surveying Steps Toward Open World Recognition," *AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

[9] A. Bendale and T. E. Boult, "Towards Open Set Deep Networks," *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel Recurrent Neural Networks," *International Conference on Machine Learning (ICML)*, vol. 48, pp. 1747–1756, 2016.

[11] I. Gulrajani, K. Kumar, A. Faruk, A. A. Taiga, F. Visin, D. Vazquez, and A. Courville, "PixelVAE: a Latent Variable Model for Natural Images," *International Conference on Learning Representations (ICLR)*, 2017.

[12] X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel, "Variational Lossy Autoencoder," *International Conference on Learning Representations (ICLR)*, 2017.

[13] H. Huang, Z. Li, R. He, Z. Sun, and T. Tan, "Introvae: Introspective variational autoencoders for photographic image synthesis," *Neural Information Processing Systems (NeurIPS)*, 2018.

[14] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "It Takes (Only) Two : Adversarial Generator-Encoder Networks," *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[15] F. Zenke, B. Poole, and S. Ganguli, "Continual Learning Through Synaptic Intelligence," *International Conference on Machine Learning (ICML)*, vol. 70, pp. 3987–3995, 2017.

[16] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences (PNAS)*, vol. 114, no. 13, pp. 3521–3526, 2017.

[17] Z. Li and D. Hoiem, "Learning without forgetting," *European Conference on Computer Vision (ECCV)*, 2016.

[18] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," *NeurIPS Deep Learning Workshop*, 2014.

[19] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong Learning with Dynamically Expandable Networks," *International Conference on Learning Representations (ICLR)*, 2018.

[20] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive Neural Networks," *arXiv preprint arXiv:1606.04671*, 2016.

[21] A. Robins, "Catastrophic Forgetting, Rehearsal and Pseudorehearsal," *Connection Science*, vol. 7, no. 2, pp. 123–146, 1995.

[22] S. A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[23] T. Mensink, J. Verbeek, F. Perronnin, G. Csurka, T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka, "Metric Learning for Large Scale Image Classification : Generalizing to New Classes at Near-Zero Cost," *European Conference on Computer Vision (ECCV)*, 2012.

[24] O. Bachem, M. Lucic, and A. Krause, "Coresets for Nonparametric Estimation - the Case of DP-Means," *International Conference on Machine Learning (ICML)*, vol. 37, pp. 209–217, 2015.

[25] R. C. O'Reilly and K. A. Norman, "Hippocampal and neocortical contributions to memory: Advances in the complementary learning systems framework," *Trends in Cognitive Sciences*, vol. 6, no. 12, pp. 505–510, 2003.

[26] A. Gepperth and C. Karaoguz, "A Bio-Inspired Incremental Learning Architecture for Applied Perceptual Problems," *Cognitive Computation*, vol. 8, no. 5, pp. 924–934, 2016.

[27] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] H. Shin, J. K. Lee, and J. J. Kim, "Continual Learning with Deep Generative Replay," *Neural Information Processing Systems (NeurIPS)*, 2017.

[29] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," *Neural Information Processing Systems (NeurIPS)*, 2014.

[30] C. V. Nguyen, Y. Li, T. D. Bui, and R. E. Turner, "Variational Continual Learning," *International Conference on Learning Representations (ICLR)*, 2018.

[31] W. J. Scheirer, A. Rocha, A. Sapkota, and T. E. Boult, "Towards Open Set Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 35, no. 7, pp. 1757–1772, 2013.

[32] W. J. Scheirer, L. P. Jain, and T. E. Boult, "Probability Models For Open Set Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[33] A. Bendale and T. Boult, "Towards Open World Recognition," *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[34] D. J. C. MacKay, "A Practical Bayesian Framework," *Neural Computation*, vol. 472, no. 1, pp. 448–472, 1992.

[35] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian Approximation : Representing Model Uncertainty in Deep Learning," *International Conference on Machine Learning (ICML)*, vol. 48, 2015.

[36] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan, "Do Deep Generative Models Know What They Don't Know?" *International Conference on Learning Representations (ICLR)*, 2019.

[37] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan, and J. Snoek, "Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift," *Neural Information Processing Systems (NeurIPS)*, 2019.

[38] S. Liang, Y. Li, and R. Srikant, "Enhancing the Reliability of Out-of-distribution Image Detection in Neural Networks," *International Conference on Learning Representations (ICLR)*, 2018.

[39] K. Lee, H. Lee, K. Lee, and J. Shin, "Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples," *International Conference on Learning Representations (ICLR)*, 2018.

[40] A. R. Dhamija, M. Günther, and T. E. Boult, "Reducing Network Agnostophobia," *Neural Information Processing Systems (NeurIPS)*, 2018.

[41] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework," *International Conference on Learning Representations (ICLR)*, 2017.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *International Conference on Computer Vision (ICCV)*, 2015.

[43] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner, "Understanding disentangling in beta-VAE," *Neural Information Processing Systems (NeurIPS), Workshop on Learning Disentangled Representations*, 2017.

[44] E. Mathieu, T. Rainforth, N. Siddharth, and Y. W. Teh, "Disentangling disentanglement in variational autoencoders," *International Conference on Machine Learning (ICML)*, pp. 7744–7754, 2019.

[45] M. D. Hoffman and M. J. Johnson, "ELBO surgery: yet another way to carve up the variational evidence lower bound," *Neural Information Processing Systems (NeurIPS), Advances in Approximate Bayesian Inference Workshop*, 2016.

[46] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998.

[47] J. M. Tomczak and M. Welling, "VAE with a vampprior," *International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol. 84, 2018.

[48] M. Bauer and A. Mnih, "Resampled Priors for Variational Autoencoders," *International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol. 89, 2019.

[49] H. Takahashi, T. Iwata, Y. Yamanaka, M. Yamada, and S. Yagi, "Variational Autoencoder with Implicit Optimal Priors," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 5066–5073, 2019.

[50] M. E. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," *Computer Vision and Pattern Recognition (CVPR)*, 2006.

[51] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," *arXiv preprint arXiv: 1708.07747*, 2017.

[52] S. Becker, M. Ackermann, S. Lapuschkin, K.-R. Müller, and W. Samek, "Interpreting and Explaining Deep Neural Networks for Classification of Audio Signals," *arXiv preprint arXiv: 1807.03418*, 2018.

[53] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[54] S. Zagoruyko and N. Komodakis, "Wide Residual Networks," *British Machine Vision Conference (BMVC)*, 2016.

[55] D. Lopez-Paz and M. A. Ranzato, "Gradient Episodic Memory for Continual Learning," *Neural Information Processing Systems (NeurIPS)*, 2017.

[56] R. Kemker, M. McClure, A. Abitino, T. Hayes, and C. Kanan, "Measuring Catastrophic Forgetting in Neural Networks," *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[57] D. P. Kingma and J. L. Ba, "Adam: a Method for Stochastic Optimization," *International Conference on Learning Representations (ICLR)*, 2015.

[58] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha, "Deep Learning for Classical Japanese Literature," *Neural Information Processing Systems (NeurIPS), Workshop on Machine Learning for Creativity and Design*, 2018.

[59] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading Digits in Natural Images with Unsupervised Feature Learning," *Neural Information Processing Systems (NeurIPS), Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[60] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," Toronto, Tech. Rep., 2009.

[61] R. Kemker and C. Kanan, "FearNet: Brain-inspired model for incremental learning," *International Conference on Learning Representations (ICLR)*, 2018.

[62] Y. Wu, Y. Chen, L. Wang, Y. Ye, Z. Liu, Y. Guo, and Y. Fu, "Large Scale Incremental Learning," *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[63] A. Rannen, R. Aljundi, M. B. Blaschko, and T. Tuytelaars, "Encoder Based Lifelong Learning," *International Conference on Computer Vision (ICCV)*, 2017.

[64] C. Wu, L. Herranz, X. Liu, Y. Wang, J. van de Weijer, and B. Raducanu, "Memory Replay GANs: learning to generate images from new categories without forgetting," *Neural Information Processing Systems (NeurIPS)*, 2018.

[65] M. Zhai, L. Chen, F. Tung, J. He, M. Nawhal, and G. Mori, "Lifelong GAN: Continual Learning for Conditional Image Generation," *International Conference on Computer Vision (ICCV)*, 2019.

[66] A. B. L. Larsen, S. K. Sonderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *International Conference On Machine Learning (ICML)*, 2016.

[67] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling, "Semi-Supervised Learning with Deep Generative Models," *Neural Information Processing Systems (NeurIPS)*, 2014.

[68] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *International Conference on Machine Learning (ICML)*, 2015.

APPENDIX

SUPPLEMENTARY MATERIAL STRUCTURE

The supplementary material provides further details for the material presented in the main body. Specifically, the structure is as follows:

- **A.** Derivation of our model and loss in equation 1 of the main body.
- **B.** Extended discussion, qualitative and quantitative examples for the role of β.
- **C.** Full specification of training procedure and hyper-parameters, including exact architecture definitions.
- **D.** Additional visualization of open set detection for all quantitatively evaluated models considered in table 2 of the main body.
- **E.** Continual learning results with a multi-layer perceptron (MLP).
- **F.** Full continual learning results for all task increments of the MNIST, FashionMNIST and AudioMNIST main body experiments, including all reconstruction losses and KL divergences.
- **G.** Visualization of generative replay examples for MNIST, FashionMNIST and AudioMNIST.

A. LOSS DERIVATION

As mentioned in the main body of the paper, in supervised continual learning we are confronted with a dataset $\boldsymbol{D} \equiv \left\{\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)\right\}_{n=1}^{N}$, consisting of N pairs of data instances $\boldsymbol{x}^{(n)}$ and their corresponding labels $y^{(n)} \in \{1 \dots C\}$ for C classes. We consider a problem scenario similar to the one introduced in "Auto-Encoding Variational Bayes" [5], i.e. we assume that there exists a data generation process responsible for the creation of the labelled data given some random latent variable $\boldsymbol{z}$. For simplicity, we follow the authors' derivation for our model with the additional inclusion of data labels, but without the β term that is present in the main body.

Ideally we would like to maximize $p(\boldsymbol{x}, \boldsymbol{y}) = \int p(\boldsymbol{z}) p(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{z}) d\boldsymbol{z}$, where the integral and the true posterior density

$$p(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y}) = \frac{p(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{z})}{p(\boldsymbol{x}, \boldsymbol{y})} \tag{12}$$

are intractable. We thus follow the standard practice of using variational Bayesian inference and introducing an approximation to the posterior $q(\boldsymbol{z})$, for which we will specify the exact form later. Making use of the properties of logarithms and applying above Bayes rule, we can now write:

$$\begin{aligned}\log p(\boldsymbol{x}, \boldsymbol{y}) = \int q(\boldsymbol{z}) [&\log p(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{z}) + \log p(\boldsymbol{z}) \\ &- \log p(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y}) + \log q(\boldsymbol{z}) - \log q(\boldsymbol{z})] d\boldsymbol{z},\end{aligned} \tag{13}$$

as the left-hand side is independent of $\boldsymbol{z}$ and $\int q(\boldsymbol{z}) d\boldsymbol{z} = 1$. Using the definition of the Kullback-Leibler divergence (KLD) $KL(q \,||\, p) = -\int q(x) \log(p(x)/q(x))$ we can rewrite this as:

$$\begin{aligned}&\log p(\boldsymbol{x}, \boldsymbol{y}) - KL(q(\boldsymbol{z}) \,||\, p(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})) = \\ &\mathbb{E}_{q(\boldsymbol{z})} \left[\log p(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{z})\right] - KL(q(\boldsymbol{z}) \,||\, p(\boldsymbol{z}))\end{aligned} \tag{14}$$

Here, the right hand side forms a variational lower-bound to the joint distribution $p(\boldsymbol{x}, \boldsymbol{y})$ as the KLD between approximate and true posterior on the left hand side is strictly positive.

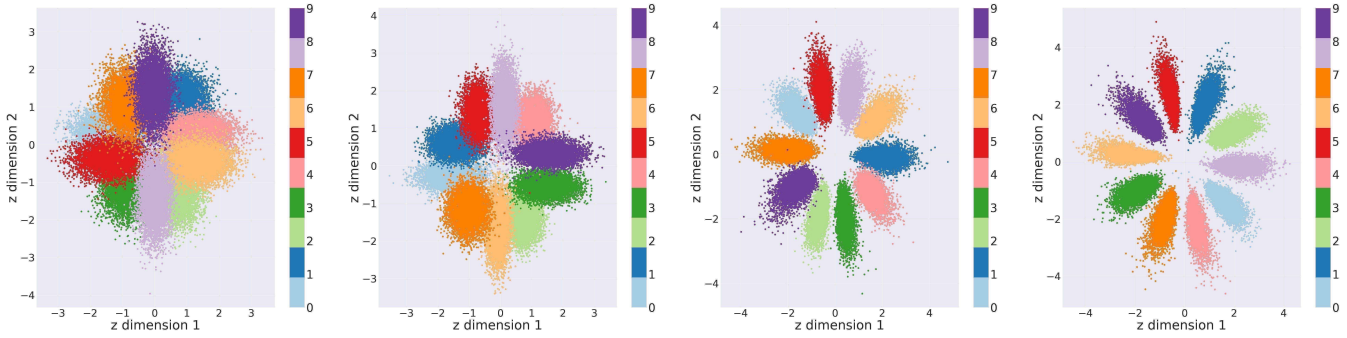
At this point we make two choices that deviate from prior works that made use of labelled data in the context of generative models for semi-supervised learning [67]. We assume a factorization of the generative process of the form $p(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z}) p(\boldsymbol{y}|\boldsymbol{z}) p(\boldsymbol{z})$ and introduce a dependency of $q(\boldsymbol{z})$ on $\boldsymbol{x}$, but not explicitly on $\boldsymbol{y}$, i.e. $q(\boldsymbol{z}|\boldsymbol{x})$. In contrast to class-conditional generation, this dependency essentially assumes that all information about the label can be captured by the latent $\boldsymbol{z}$ and there is thus no additional benefit in explicitly providing the label when estimating the data likelihood $p(\boldsymbol{x}|\boldsymbol{z})$. This is crucial as our probabilistic encoder should be able to predict labels without requiring it as input to our model, i.e. $q(\boldsymbol{z}|\boldsymbol{x})$ instead of the intuitive choice of $q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})$. However, we would like the label to nevertheless be directly inferable from the latent $\boldsymbol{z}$. In order for the latter to be achievable, we require the corresponding classifier that learns to predict $p(\boldsymbol{y}|\boldsymbol{z})$ to be linear in nature. This guarantees linear separability of the classes in latent space, which can in turn then be used to for open set recognition and generation of specific classes as shown in the main body.

B. FURTHER DISCUSSION ON THE ROLE OF β

In the main body the role of the β term [41] in our model's loss function is pointed out. Here, we delve into further detail with qualitative and quantitative examples to support the arguments. To facilitate the discussion, we repeat equation 1 of the main body:

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)}; \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\xi}\right) = &-\beta KL(q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}^{(n)}) \,||\, p(\boldsymbol{z})) \\ &+ \mathbb{E}_{q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}^{(n)})} \left[\log p_{\boldsymbol{\phi}}(\boldsymbol{x}^{(n)}|\boldsymbol{z}) + \log p_{\boldsymbol{\xi}}(y^{(n)}|\boldsymbol{z})\right]\end{aligned} \tag{15}$$

The β term weights the strength of the regularization by the prior through the KL divergence. Selection of this strength is necessary to control the information bottleneck of the latent space and regulate the effective latent encoding overlap. To repeat the main body, and previous arguments by [45] and [43]: too large β values (typically $>> 1$) will result in a collapse of any structure present in the aggregate posterior. Too small β values (typically $<< 1$) lead to the latent space being a lookup table. In either case, there is no meaningful information between the latents. This is particularly relevant to our objective of linear class separability, that requires formation of an aggregate latent encoding that is disentangled with respect to the different classes. To visualize this, we have trained multiple models with different β values on the MNIST dataset, in an isolated fashion with all data present at all times to focus on the effect of β. The corresponding aggregate encodings at the end of training are shown in figure 9. Here, we can empirically observe above points. With a beta of one and larger, the aggregate posterior's structure starts to collapse and the aggregate encoding converges to a Normal distribution. While this minimizes the distributional mismatch with respect to the prior, the separability of classes is also lost and an accurate classification cannot be achieved. On the other hand, if the beta value gets ever smaller there is insufficient regularization present and the aggregate



(a) β 1.0 (b) β 0.5 (c) β 0.1 (d) β 0.05

Fig. 9: 2-D MNIST latent space visualization with different β values for the used WRN architecture.

TABLE 5: Losses obtained for different β values for MNIST using the WRN architecture with 2-D latent space. Training conducted in isolated fashion to quantitatively showcase the role of β. Un-normalized values in nats are reported in brackets for reference purposes.

		In nats per dimension (nats in brackets)			
2-D latent	Beta	KLD	Recon loss	Class Loss	Accuracy [%]
train	1.0	1.039 (2.078)	0.237 (185.8)	0.539 (5.39)	79.87
test		1.030 (2.060)	0.235 (184.3)	0.596 (5.96)	78.30
train	0.5	1.406 (2.812)	0.230 (180.4)	0.221 (2.21)	93.88
test		1.382 (2.764)	0.228 (178.8)	0.305 (3.05)	92.07
train	0.1	2.055 (4.110)	0.214 (167.8)	0.042 (0.42)	99.68
test		2.071 (4.142)	0.212 (166.3)	0.116 (1.16)	98.73
train	0.05	2.395 (4.790)	0.208 (163.1)	0.025 (0.25)	99.83
test		2.382 (4.764)	0.206 (161.6)	0.159 (1.59)	98.79

TABLE 6: Losses obtained for different β values for MNIST using the WRN architecture with 60-D latent space. Training conducted in isolated fashion to quantitatively showcase the role of β. Un-normalized values in nats are reported in brackets for reference purposes.

		In nats per dimension (nats in brackets)			
60-D latent	Beta	KLD	Recon loss	Class Loss	Accuracy [%]
train	1.0	0.108 (6.480)	0.184 (144.3)	0.0110 (0.110)	99.71
test		0.110 (6.600)	0.181 (142.0)	0.0457 (0.457)	99.03
train	0.5	0.151 (9.060)	0.162 (127.1)	0.0052 (0.052)	99.87
test		0.156 (9.360)	0.159 (124.7)	0.0451 (0.451)	99.14
train	0.1	0.346 (20.76)	0.124 (97.22)	0.0022 (0.022)	99.95
test		0.342 (20.52)	0.126 (98.79)	0.0286 (0.286)	99.38
train	0.05	0.476 (28.56)	0.115 (90.16)	0.0018 (0.018)	99.95
test		0.471 (28.26)	0.118 (92.53)	0.0311 (0.311)	99.34

posterior no longer follows a Normal distribution. The latter does not only render sampling for generative replay difficult, it also challenges the assumption of distances to each class' latent mean being Weibull distributed, as the latter can essentially be seen as a skewed Normal.

As pointed out in the main body, it is important to note that the losses are normalized with respect to spatial image and latent dimensionality. The value of β should thus also be seen as a normalized quantity. While the relative effect of increasing or decreasing beta stays the same, the absolute value of β can be subject to any normalization.

We provide corresponding quantitative examples for the models trained with different β with 2-D latent spaces and 60-D latent spaces in tables 5 and 6 respectively. In both cases, we observe that decreasing the value of beta below one is necessary to improve classification accuracy, as well as the overall variational lower bound. Taking the 60 dimensional case as a specific example, we can also observe that reducing the beta value too far and decreasing it from e.g. 0.1 to 0.05 leads to deterioration of the variational lower bound, from 119.596 to 121.101 natural units, while the classification accuracy by itself does not improve further.

C. Training Hyper-Parameters and Architecture Definitions

We provide a full specification of hyper-parameters, model architectures and the training procedure used in the main body. We base our encoder and decoder architecture on 14-layer wide residual networks [53], [54] with a latent dimensionality of 60 to demonstrate scalability to high-dimensions and as used in lossy auto-encoders [11], [12]. These architectures are shown in detail in tables 7 and 8. Hidden layers include batch-normalization [68] with a value of 10^{-5} and use ReLU activations. For a common frame of reference, all methods' share the same underlying WRN architecture, including the separate classifiers and generative models of the dual model approaches. Experiments with a simpler MLP architecture can be found in section E of the supplementary material. For the higher resolution 256×256 flower images, we have used a deeper 26 layer WRN version, in analogy to previous works [11], [12]. Here, the last encoder, and first decoder blocks are repeated an extra three times, resulting in an additional three stages of down- and up-sampling by factor two. The encoder's spatial output dimensionality is thus equivalent to the 14-layer architecture applied to the eight times lower resolution images of the simpler datasets. For the autoregressive addition to our joint model, we set the number of output channels of the decoder to 60 and append three additional pixel decoder layers, each with a kernel size of 7×7 and 60 channels. Whereas we report reconstruction log-likelihoods in nats, these models are practically formulated as a classification problem with a 256-way softmax. The corresponding loss is in bits per dimension. We have converted these values to have a better comparison, but in order to do so we need to sample from the pixel decoder's multinomial distribution to calculate a binary cross-entropy on reconstructed images. We further note that all losses are normalized with respect to spatial and latent dimensions, as mentioned in the main body.

We use hyper-parameters consistent with the literature [11], [12]. Accordingly, all models are optimized using

TABLE 7: 14-layer WRN encoder with a widen factor of 10. Convolutional layers (conv) are parametrized by a quadratic filter size followed by the amount of filters. p and s represent zero padding and stride respectively. If no padding or stride is specified then p = 0 and s = 1. Skip connections are an additional operation at a layer, with the layer to be skipped specified in brackets. Every convolutional layer is followed by batch-normalization and a ReLU activation function. The probabilistic encoder ends on fully-connected layers for μ and σ that depend on the chosen latent space dimensionality and the data's spatial size.

Layer type	WRN encoder	
Layer 1	conv 3 × 3 - 48, p = 1	
Block 1	conv 3 × 3 - 160, p = 1; conv 3 × 3 - 160, p = 1 conv 3 × 3 - 160, p = 1; conv 3 × 3 - 160, p = 1	conv 1 × 1 - 160 (skip next layer) shortcut (skip next layer)
Block 2	conv 3 × 3 - 320, s = 2, p = 1; conv 3 × 3 - 320, p = 1 conv 3 × 3 - 320, p = 1; conv 3 × 3 - 320, p = 1	conv 1 × 1 - 320, s = 2 (skip next layer) shortcut (skip next layer)
Block 3	conv 3 × 3 - 640, s = 2, p = 1; conv 3 × 3 - 640, p = 1 conv 3 × 3 - 640, p = 1; conv 3 × 3 - 640, p = 1	conv 1 × 1 - 640, s = 2 (skip next layer) shortcut (skip next layer)

TABLE 8: 14-layer WRN decoder with a widen factor of 10. P_w and P_h refer to the input's spatial dimension. Convolutional (conv) and transposed convolutional (conv_t) layers are parametrized by a quadratic filter size followed by the amount of filters. p and s represent zero padding and stride respectively. If no padding or stride is specified then p = 0 and s = 1. Skip connections are an additional operation at a layer, with the layer to be skipped specified in brackets. Every convolutional and fully-connected (FC) layer are followed by batch-normalization and a ReLU activation function. The model ends on a Sigmoid function.

Layer type	WRN decoder	
Layer 1	FC 640 × $\lfloor P_w/4 \rfloor$ × $\lfloor P_h/4 \rfloor$	
Block 1	conv_t 3 × 3 - 320, p = 1; conv 3 × 3 - 320, p = 1 conv 3 × 3 - 320, p = 1; conv 3 × 3 - 320, p = 1 upsample × 2	conv_t 1 × 1 - 320 (skip next layer) shortcut (skip next layer)
Block 2	conv_t 3 × 3 - 160, p = 1; conv 3 × 3 - 160, p = 1 conv 3 × 3 - 160, p = 1; conv 3 × 3 - 160, p = 1 upsample × 2	conv_t 1 × 1 - 160 (skip next layer) shortcut (skip next layer)
Block 3	conv_t 3 × 3 - 48, p = 1; conv 3 × 3 - 48, p = 1 conv 3 × 3 - 48, p = 1; conv 3 × 3 - 48, p = 1	conv_t 1 × 1 - 48 (skip next layer) shortcut (skip next layer)
Layer 2	conv 3 × 3 - 3, p = 1	

stochastic gradient descent with a mini-batch size of 128 and Adam [57] with a learning rate of 0.001 and first and second momenta equal to 0.9 and 0.999. For MNIST, FashionMNIST and AudioMNIST no data augmentation or preprocessing is applied. For the flower experiments, images are stochastically flipped horizontally with a 50 % chance and the batch size is reduced to 32. We initialize all weights according to [42].

All class incremental models are trained for 120 epochs per task on MNIST and FashionMNIST and 150 epochs on AudioMNIST. Complementary incremental cross-dataset models are trained for 200 epochs per task on data resized to 32×32. While our proposed model exhibits forward transfer due to weight sharing and need not necessarily be trained for the entire amount of epochs for each subsequent task, this guarantees convergence and a fair comparison of results with respect to achievable accuracy of other methods. Isolated models are trained for 200 and 300 epochs until convergence respectively. Due to the much smaller dataset size, architectures are trained for 2000 epochs on the flower images, in order to obtain a similar amount of update steps. For the generative replay with statistical outlier rejection, we use an aggressive rejection rate of $\Omega_t = 0.01$ (with analogous results with 0.05) and dynamically set tail-sizes to 5% of seen examples per class. As mentioned in the main body, the used open set distance measure is the cosine distance.

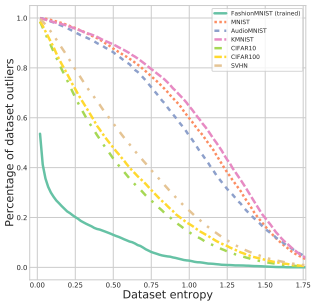
For EWC, the number of Fisher samples is fixed to the total number of data points from all the previously seen tasks. A suitable Fisher multiplier value λ has been determined by conducting a grid search over a set of five values: 50, 100, 500, 1000 and 5000 on held-out validation data for the first two tasks in sequence. We observe exploding gradients if λ is too high. However, a very small λ leads to excessive drift in the weight distribution across subsequent tasks that further results in catastrophic inference. Empirically, $\lambda = 500$ in the class-incremental scenario and $\lambda = 1000$ in the cross-dataset setting seem to provide the best balance.

D. Additional open set recognition visualization

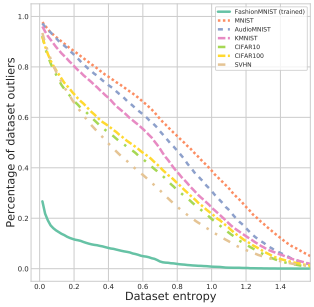
As we point out in section 4 of the main paper, our posterior based open set recognition considers almost all of the unknown datasets as statistical outliers, while at the same time regarding unseen test data from the originally trained tasks as distribution inliers across a wide range of rejection priors. In addition to the outlier rejection curves for FashionMNIST and the quantitative results presented in the main body, we also show the full outlier rejection curves for the remaining datasets, as well as all dual model approaches in figures 10, 11 and 12. These figures visually support the quantitative findings described in the main body and respective conclusions. In summary, the joint OCDVAE performs better at open set recognition in direct comparison to the dual model setting, particularly when using the EVT based criterion. Apart from the MNIST dataset, where reconstruction loss can be a sufficient metric for open set detection, the latent based approach also exhibits less dependency on the outlier rejection prior and consistently improves the ability to discern unknown data.

Monte Carlo Dropout

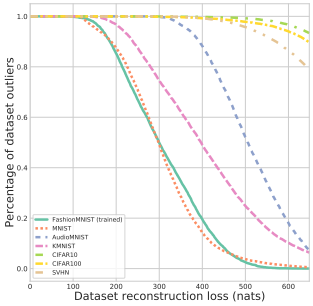
In this subsection we provide additional quantitative results for open set recognition with Monte-Carlo Dropout (MCD) in order to assess the effectiveness of approximating a distribution on the weights to estimate uncertainty, in addition to the experiments of the main body where the latent variable is marginalised. We have therefore re-trained all of the models reported in table 2 with a Dropout probability of 0.2 in each layer. We then conduct 50 stochastic forward passes through the entire model for prediction. The obtained open set recognition results are reported in 9. Although MCD boosts the outlier detection accuracy, particularly



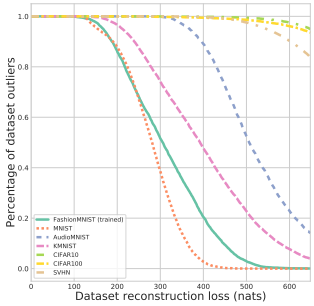
(a) Dual model classifier entropy based OSR



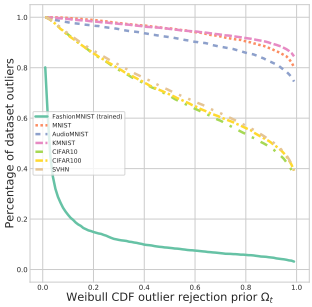
(b) OCDVAE classifier entropy based OSR



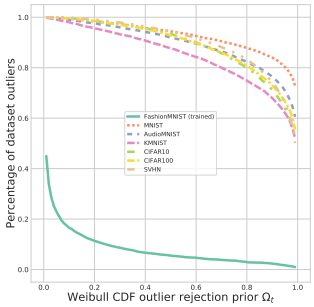
(c) Dual model reconstruction loss based OSR



(d) OCDVAE reconstruction loss based OSR

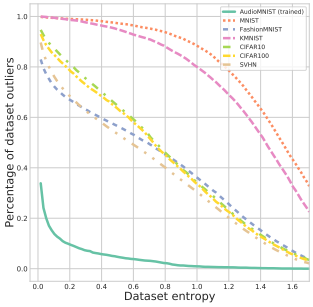


(e) Dual model posterior EVT based OSR

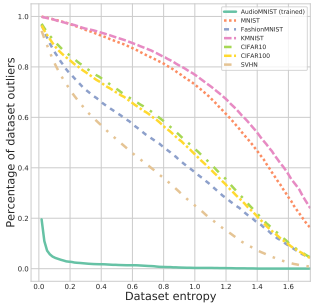


(f) OCDVAE posterior EVT based OSR

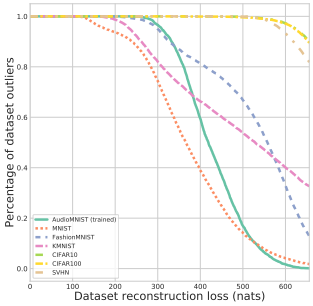
Fig. 10: Dual model and OCDVAE trained on FashionMNIST evaluated on unseen datasets. Pairs of panels show the contrast between the approaches. Left panels correspond to the dual model, right panels show the joint OCDVAE model. (a+b) The classifier entropy values by itself are insufficient to separate most of unknown from the known task's test data. (c+d) Reconstruction loss allows for a partial distinction. (e+f) Our posterior-based open set recognition considers the large majority of unknown data as statistical outliers across a wide range of rejection priors Ω_t, significantly more so in the OCDVAE model. All metrics are reported as the mean over 100 approximate posterior samples per data point.



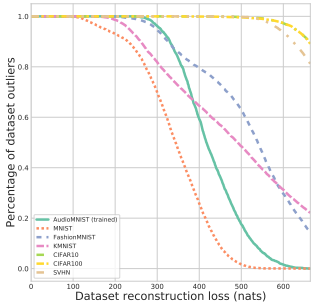
(a) Dual model classifier entropy based OSR



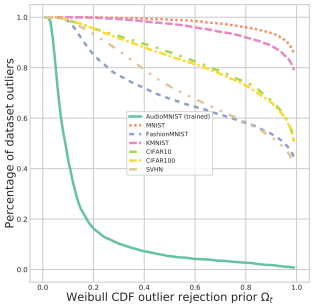
(b) OCDVAE classifier entropy based OSR



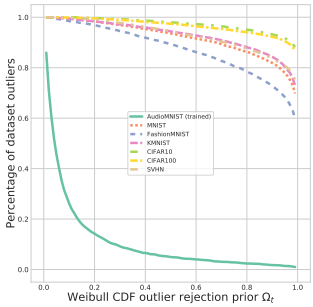
(c) Dual model reconstruction loss based OSR



(d) OCDVAE reconstruction loss based OSR



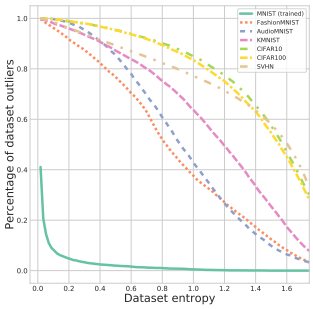
(e) Dual model posterior EVT based OSR



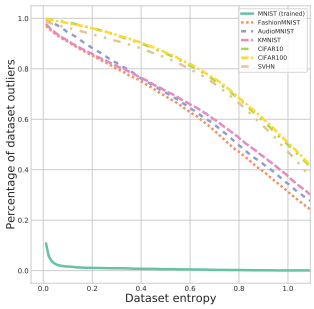
(f) OCDVAE posterior EVT based OSR

Fig. 11: Dual model and OCDVAE trained on AudioMNIST evaluated on unseen datasets. Pairs of panels show the contrast between the approaches. Left panels correspond to the dual model, right panels show the joint OCDVAE model. (a+b) The classifier entropy values by itself are insufficient to separate most of unknown from the known task's test data. (c+d) Reconstruction loss allows for a partial distinction. (e+f) Our posterior-based open set recognition considers the large majority of unknown data as statistical outliers across a wide range of rejection priors Ω_t, significantly more so in the OCDVAE model. All metrics are reported as the mean over 100 approximate posterior samples per data point.

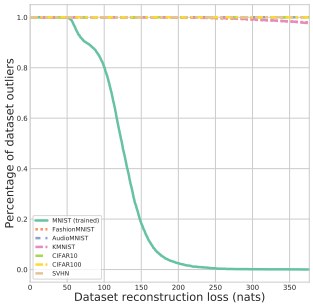
for criteria such as predictive entropy, the insights of the main body still hold. In summary, the joint model generally outperforms a purely discriminative model in terms of open set recognition, independently of the used metric, and our proposed aggregate posterior based EVT approach of the OCDVAE yields an almost perfect separation of known and unseen unknown data. Interestingly, this has already been achieved in the experiments of the main body. Resorting to the repeated model calculation of MCD thus seems to come without enough of an advantage to warrant the added computational complexity in the context of posterior based open set recognition.



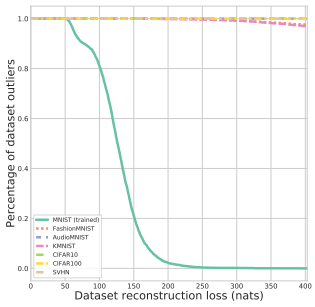
(a) Dual model classifier entropy based OSR



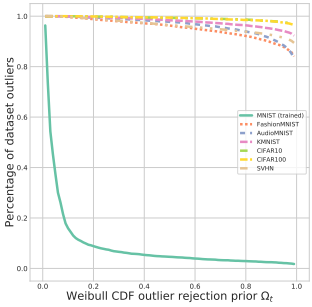
(b) OCDVAE classifier entropy based OSR



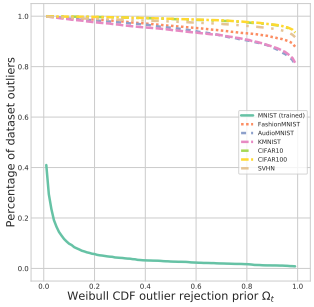
(c) Dual model reconstruction loss based OSR



(d) OCDVAE reconstruction loss based OSR



(e) Dual model posterior EVT based OSR



(f) OCDVAE posterior EVT based OSR

Fig. 12: Dual model and OCDVAE trained on MNIST evaluated on unseen datasets. Pairs of panels show the contrast between the approaches. Left panels correspond to the dual model, right panels show the joint OCDVAE model. (a+b) The classifier entropy values by itself can achieve a partial separation between unknown and the known task's test data. (c+d) Reconstruction loss allows for distinction if the cut-off is chosen correctly. (e+f) Our posterior-based open set recognition considers the large majority of unknown data as statistical outliers across a wide range of rejection priors Ω_t. All metrics are reported as the mean over 100 approximate posterior samples per data point. While the OCDVAE shows improvement upon the dual model approach, particularly if using classifier entropies for OSR, both models trained on MNIST perform well in OSR. In direct contrast with models trained on Fashion- or AudioMNIST and respective figures 10 and 11, this shows that evaluation on MNIST alone is generally insufficient.

TABLE 9: Test accuracies and outlier detection values of the joint OCDVAE and dual model (VAE and separate deep classifier) approaches when considering 95 % of known tasks' validation data is inlying. Percentage of detected outliers is reported based on classifier predictive entropy, reconstruction loss and our posterior based EVT approach, averaged over 50 Monte Carlo dropout samples, with $p_{dropout} = 0.2$ for each layer, per data-point respectively. Note that larger values are better, except for the test data of the trained dataset, where ideally 0% should be considered as outlying.

Outlier detection at 95% validation inliers (%)				MNIST	Fashion	Audio	KMNIST	CIFAR10	CIFAR100	SVHN
Trained	Model	Test acc.	Criterion							
FashionMNIST	Dual, CL + VAE	90.58	Class entropy	75.50	5.366	70.78	74.41	49.42	49.17	38.84
			Reconstruction	55.45	5.048	59.99	99.83	99.35	99.35	99.62
			Latent EVT	77.03	4.920	55.48	70.23	58.73	57.06	44.54
	Joint, OCDVAE	91.50	Class Entropy	85.05	4.740	67.90	78.04	63.89	66.11	59.42
			Reconstruction	1.227	5.422	85.85	39.76	99.94	99.72	99.99
			Latent EVT	95.83	4.516	94.56	96.04	96.81	96.66	96.28
MNIST	Dual, CL + VAE	99.41	Class entropy	4.276	91.88	96.50	96.65	95.84	97.37	98.58
			Reconstruction	4.829	99.99	100.0	99.90	100.0	100.0	100.0
			Latent EVT	4.088	87.84	98.06	95.79	97.34	98.30	95.74
	Joint, OCDVAE	99.54	Class entropy	4.801	97.63	99.38	98.01	99.16	99.39	98.90
			Reconstruction	5.264	99.98	100.0	100.0	100.0	100.0	100.0
			Latent EVT	4.978	99.99	100.0	99.94	99.96	99.95	99.68
AudioMNIST	Dual, CL + VAE	98.76	Class entropy	99.97	61.26	4.996	96.77	63.78	65.76	59.38
			Reconstruction	7.334	52.37	5.100	98.19	99.97	99.90	99.96
			Latent EVT	92.74	67.18	5.073	90.41	90.56	90.97	89.58
	Joint, OCDVAE	98.85	Class entropy	99.39	89.50	5.333	99.16	94.66	95.12	97.13
			Reconstruction	15.81	53.83	4.837	41.89	99.90	99.82	99.95
			Latent EVT	99.50	99.27	5.136	99.75	99.71	99.59	99.91

TABLE 10: Results for continual learning across datasets averaged over 5 runs, baselines and the reference isolated learning scenario for FashionMNIST (F) $\rightarrow$ MNIST (M) $\rightarrow$ AudioMNIST (A) and the reverse order. α_T indicates the respective accuracy at the end of the last increment $T = 3$.

Cross-dataset		α_T(%) (T=3)					
		base		new		all	
		MLP	WRN	MLP	WRN	MLP	WRN
F-M-A	**CDVAE ISO**					93.86	94.95
	CDVAE UB	89.75	89.10	97.28	97.88	93.94	95.00
	CDVAE LB	00.00	00.00	97.38	98.12	22.51	22.70
	EWC	42.10 $\pm$ 1.880	22.85 $\pm$ 0.294	31.33 $\pm$ 2.037	93.31 $\pm$ 0.138	46.04 $\pm$ 1.195	43.42 $\pm$ 0.063
	Dual Model	81.12 $\pm$ 0.341	81.89 $\pm$ 0.104	97.15 $\pm$ 0.320	96.78 $\pm$ 0.067	91.03 $\pm$ 0.096	91.75 $\pm$ 0.064
	CDVAE	74.23 $\pm$ 0.587	57.70 $\pm$ 4.480	97.04 $\pm$ 0.105	96.73 $\pm$ 0.235	85.55 $\pm$ 0.234	81.10 $\pm$ 1.769
	OCDVAE	79.01 $\pm$ 0.591	80.11 $\pm$ 2.922	97.34 $\pm$ 0.152	97.63 $\pm$ 0.042	89.87 $\pm$ 0.262	91.13 $\pm$ 1.045
A-M-F	**CDVAE ISO**					93.67	94.95
	CDVAE UB	96.97	97.17	89.34	89.16	93.75	94.91
	CDVAE LB	00.00	00.00	89.81	89.72	34.55	34.51
	EWC	7.178 $\pm$ 2.432	3.420 $\pm$ 0.026	73.83 $\pm$ 2.873	87.54 $\pm$ 0.214	46.37 $\pm$ 1.908	45.42 $\pm$ 0.731
	Dual Model	51.70 $\pm$ 2.611	66.82 $\pm$ 0.337	89.53 $\pm$ 0.093	89.15 $\pm$ 0.050	83.95 $\pm$ 0.644	87.70 $\pm$ 0.102
	CDVAE	65.38 $\pm$ 2.501	79.74 $\pm$ 2.431	89.30 $\pm$ 0.116	88.50 $\pm$ 0.126	86.19 $\pm$ 0.584	89.46 $\pm$ 0.600
	OCDVAE	81.65 $\pm$ 1.414	94.53 $\pm$ 0.283	89.31 $\pm$ 0.109	89.53 $\pm$ 0.367	90.08 $\pm$ 0.471	94.06 $\pm$ 0.156

TABLE 11: Results for class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for the three datasets. α_T indicates the respective accuracy at the end of the last increment $T = 5$.

Class-incremental		α_T(%) (T=5)					
		base		new		all	
		MLP	WRN	MLP	WRN	MLP	WRN
FashionMNIST	**CDVAE ISO**					87.68	89.54
	CDVAE UB	91.10	92.20	96.75	97.50	87.35	89.24
	CDVAE LB	00.00	00.00	99.75	99.80	19.95	19.97
	EWC	21.79 ± 2.610	00.17 ± 0.076	96.80 ± 0.873	99.60 ± 0.023	24.48 ± 2.862	20.06 ± 0.059
	Dual Model	91.64 ± 1.233	94.26 ± 0.192	97.18 ± 0.171	93.55 ± 0.708	68.49 ± 2.110	63.21 ± 1.957
	CDVAE	49.71 ± 1.363	39.51 ± 7.173	97.84 ± 0.375	96.92 ± 0.774	62.72 ± 1.379	58.82 ± 2.521
	OCDVAE	56.67 ± 2.279	60.63 ± 12.16	97.89 ± 0.332	96.51 ± 0.707	66.14 ± 0.497	69.88 ± 1.712
MNIST	**CDVAE ISO**					98.87	99.45
	CDVAE UB	99.57	99.57	98.04	99.10	98.84	99.29
	CDVAE LB	00.00	00.00	99.75	99.85	19.92	20.16
	EWC	24.08 ± 0.487	00.45 ± 0.059	96.70 ± 2.039	99.58 ± 0.052	26.46 ± 2.351	20.26 ± 0.027
	Dual Model	92.63 ± 1.609	97.31 ± 0.489	98.48 ± 0.145	98.59 ± 0.106	89.74 ± 0.726	96.64 ± 0.079
	CDVAE	34.48 ± 9.512	19.86 ± 7.396	98.84 ± 0.228	99.00 ± 0.100	60.88 ± 3.308	64.34 ± 4.903
	OCDVAE	82.54 ± 2.26	92.35 ± 4.485	98.89 ± 0.151	99.06 ± 0.171	87.31 ± 1.224	93.24 ± 3.742
AudioMNIST	**CDVAE ISO**					96.33	97.75
	CDVAE UB	99.08	98.42	98.25	98.67	96.43	97.87
	CDVAE LB	00.00	00.00	99.92	100.0	20.03	20.02
	EWC	17.51 ± 3.380	00.11 ± 0.007	85.25 ± 4.209	99.41 ± 0.207	20.48 ± 1.727	19.98 ± 0.032
	Dual Model	53.60 ± 0.586	61.58 ± 0.747	97.22 ± 0.559	89.41 ± 0.691	48.42 ± 2.808	47.42 ± 1.447
	CDVAE	20.76 ± 5.521	59.36 ± 7.147	89.21 ± 0.402	84.93 ± 6.297	69.76 ± 1.369	81.49 ± 1.944
	OCDVAE	56.68 ± 5.059	79.73 ± 4.070	89.35 ± 0.244	89.52 ± 6.586	81.84 ± 1.438	87.72 ± 1.594

E. MLP BASED CONTINUAL LEARNING

For comparably simple datasets such as MNIST, it could be argued that optimizing a deep WRN decoder for generative replay is more expensive than simply storing the entire original MNIST dataset for continued classifier training. In the main body we have used this WRN architecture to provide a common frame of reference across all experiments. To nevertheless demonstrate that such a complex network is not essential for continual learning of simple datasets, we repeat all MNIST, FashionMNIST and AudioMNIST with a shallow MLP architecture of limited representational capacity. To allow for a direct comparison with the WRN based results, we use the same latent dimensionality of 60 and similarly let all the other hyper-parameters remain the same. However, we replace the deep encoder and decoder with two fully-connected hidden layers of 400 units [56]. The corresponding quantitative results for cross-dataset and class incremental learning are reported in tables 10 and 11 respectively. The assuring main observation is that the MLP models fare only marginally worse, with the biggest difference to the WRN being perceivable on the audio dataset. However, the relative ranking of individual methods remains the same in almost all cases and the general insight and conclusions of the main body prevail. The only exception is the use of EWC in conjunction with the shallow MLP. With a lambda value of 500, we find EWC in an MLP to work significantly better than in application to the deep counterpart, in particular in initial task increments. Although the approach still faces difficulty with a growing single-head classifier, see the discussion in section 4 of the main body, and is still by far the worst in a global comparison, it no longer directly mirrors the lower bound accuracy. We hypothesize that this is due to a more informative and accurate estimate of important parameters in the presence of only two layers with significantly less units.

F. DETAILED RESULTS FOR THE MNIST, FASHION-MNIST AND AUDIOMNIST EXPERIMENTS

In the main body we have reported three metrics for our continual learning experiments based on classification accuracy: the base task's accuracy over time $\alpha_{t,base}$, the new task's accuracy $\alpha_{t,new}$ and the overall accuracy at any point in time $\alpha_{t,all}$. This is an appropriate measure to evaluate the quality of the generative model over time given that the employed mechanism to avoid catastrophic inference in continual learning is generative replay. On the one hand, if catastrophic inference occurs in the decoder the sampled data will no longer resemble the instances of the observed data distribution. This will in turn degrade the encoder during continued training and thus the classification accuracy. On the other hand, this proxy measure for the generation quality avoids the common pitfalls of pixel-wise reconstruction metrics. The information necessary to maintain respective knowledge of the data distribution through the variational approximation in the probabilistic encoder does not necessarily rely on correctly reconstructing data's local information. To take an example, if a model were to reconstruct all images perfectly but with some degree of spatial translation or rotation, then the negative log likelihood (NLL) would arguable be worse than that of a model which reconstructs local details correctly on a pixel level for a fraction of the image. As this could be details in e.g. the background or other class unspecific areas, training on corresponding generations does not have to prevent loss of encoder knowledge with respect to the classification task.

As such, a similar argument can be conjured for the KL divergence. On the one hand, monitoring the KL divergence as a regularization term by itself over the course of continual learning is meaningless without regarding the data's NLL. On the other hand, for our OCDVAE model the exact value of the KL divergence does not immediately reflect the quality of the generated data. This is because we do not sample merely from the prior, but as explained in the main body employ a rejection mechanism to draw samples that belong to the aggregate posterior.

Nevertheless, for the purpose of completeness and in addition to the results provided in the experimental section of the main body, we provide the reconstruction losses and KL divergences for all applicable models in this supplementary material section. Analogous to the three metrics for classification accuracy of base, new and all tasks, we define the respective reconstruction losses $\gamma_{t,base}, \gamma_{t,new}$ and $\gamma_{t,all}$. The KL divergence KL_t always measures the deviation from the prior $p(z)$ at any point in time, as the prior remains the same throughout continual training. Following the above discussion, we argue that these values should be regarded with caution and should not be interpreted separately.

Full cross dataset results

We show the full cross dataset results in table 12 in extension to table 1 in the main body. An analogous table for the presented autoregressive models can be found in table 13. Similar to the accuracy values, we can observe that the mismatch between aggregate posterior and prior as expressed through the KL divergence is greater in a naive joint model (naive CDVAE) in comparison to a dual model approach with separate generative and discriminative models. Our proposed OCDVAE model, with respective rejection sampling scheme that takes into account the structure of the aggregate posterior, alleviates this to a large degree. The reconstruction losses of both the dual model and the joint OCDVAE approach show only negligible deviation with respect to the achievable upper bound and only limited catastrophic inference of the decoder occurs. However, we can also observe that by itself these quantities are not indicative of maintaining encoder knowledge with respect to representations required for classification. This is particularly visible in the tables' second experiment, where we first train Audio data and then proceed with the two image datasets. Here, the KL divergence and reconstruction loss are both better for the dual model, whereas a much higher accuracy over time is maintained in the OCDVAE model. Naturally, this is because a significant mismatch between aggregate posterior and prior is also present in a purely unsupervised generative model and naively sampling from the prior will result in generated instances that do not resemble those present in the observed data distribution. While weaker in effect, this is similar to the naive CDVAE approach. Without the presence of the linear discriminator on the latents in the purely unsupervised generative model, there is however no

straightforward mechanism to disentangle the latent space according to classes. Our proposed open set approach and the resulting constraint to samples from the aggregate posterior as presented in the OCDVAE is thus not trivially applicable.

Full class incremental results

In addition to reconstruction losses and KL divergences, we also report the detailed full set of intermediate results for the five task steps of the class incremental scenario. We thus extend table 3 in the main body with results for all task increments $t = 1, \ldots, 5$ and a complete list of losses in tables 14, 15 and 16 for the three datasets respectively. The corresponding results for autoregressive models are presented in tables 17, 18 and 19.

Once more, we can observe the increased effect of error accumulation due to unconstrained generative sampling from the prior in comparison to the open set counterpart that limits sampling to the aggregate posterior. The statistical deviations across experiment repetitions in the base and the overall classification accuracies are higher and are generally decreased by the open set models. For example, in table 14 the MNIST base and overall accuracy deviations of a naive CDVAE are higher than the respective values for OCDVAE starting already from the second task increment. Correspondingly, the accuracy values themselves experience larger decline for CDVAE than for OCDVAE with progressive increments. This difference is not as pronounced at the end of the first task increment because the models haven't been trained on any of their own generated data yet. Successful literature approaches such as the variational generative replay proposed by [6] thus avoid repeated learning based on previous generated examples and simply store and retain a separate generative model for each task. The strength of our model is that, instead of storing a trained model for each task increment, we are able to continually keep training our joint model with data generated for all previously seen tasks by filtering out ambiguous samples from low density areas of the posterior. Similar trends can also be observed for the respective pixel models.

We also see that regularization approaches such as EWC already fail at the first increment. In contrast to the success that has been reported in prior literature [16], [56], this is due to the use of a single classification head. This is intuitive because introduction of new units, as described in the main body, directly confuses the existing classification. Regularization approaches by definition are challenged in this scenario because the weights are not allowed to drift too far away from previous values. For emphasis we repeat that however this scenario is much more practical and realistic than a multi-head scenario with a separate classifier per task. While regularization approaches are largely successful in the latter setting, it is not only restricted to the closed world, but further requires an oracle at prediction stage to chose the correct classification head. In contrast, our proposed approach requires no knowledge of task labels for prediction and is robust in an open world.

With respect to KL divergences and reconstruction losses we can make two observations. First, the arguments of the previous section hold and by itself the small relative improvements between models should be interpreted with caution as they do not directly translate to maintaining continual learning accuracy. Second, we can also observe that reconstruction losses at every increment for all $\gamma_{t,all}$ and respective negative log likelihoods for only the new task $\gamma_{t,new}$ are harder to interpret than the accuracy counterpart. While the latter is normalized between zero and unity, the reconstruction loss of different tasks is expected to fluctuate largely according to the task's images' reconstruction complexity. To give a concrete example, it is rather straightforward to come to the conclusion that a model suffers from limited capacity or lack of complexity if a single newly arriving class cannot be classified well. In the case of reconstruction it is common to observe either a large decrease in negative log likelihood for the newly arriving class, or a big increase depending on the specific introduced class. As such, these values are naturally comparable between models, but are challenging to interpret across time steps without also analyzing the underlying nature of the introduced class. The exception is formed by the base task's reconstruction loss $\gamma_{t,base}$. In analogy to base classification accuracy, this quantity still measures the amount of catastrophic forgetting across time. However, in all tables we can observe that catastrophic forgetting of the decoder as measured by the base reconstruction loss is almost imperceivable. As this is not at all reflected in the respective accuracy over time, it further underlines our previous arguments that reconstruction loss is not necessarily the best metric to monitor in the presented continual learning scenario.

G. Generative replay examples with CDVAE and OCDVAE

In this section we provide visualization of data instances that are produced during generative replay at the end of each task increment. In particular, we qualitative illustrate the effect of constraining sampling to the aggregate posterior in contrast to naively sampling from the prior without statistical outlier rejection for low density regions. Figures 13, 14 and 15 illustrate generated images for MNIST, FashionMNIST and AudioMNIST respectively. For both a naive CDVAE as well as the autoregressive PixCDVAE we observe significant confusion with respect to classes. As the generative model needs to learn how to replay old tasks' data based on its own former generations, ambiguity and blurry interpolations accumulate and are rapidly amplified. This is not the case for OCDVAE and PixOCDVAE, where the generative model is capable of maintaining higher visual fidelity throughout continual training and misclassification is scarce.

TABLE 12: Results for incremental cross-dataset continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for FashionMNIST (F) $\rightarrow$ MNIST (M) $\rightarrow$ AudioMNIST (A) and the reverse order. Extension of table 1 in the main body. Here, in addition to the accuracy α_T, γ_T and KL_T also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of the last increment $T = 3$.

Cross-dataset		α_T(%)			γ_T(nats)			KL_T(nats)
		base	new	all	base	new	all	all
F-M-A	**CDVAE ISO**			94.95			269.6	24.97
	CDVAE UB	89.10	97.88	95.00	311.2	434.3	269.7	25.20
	CDVAE LB	00.00	98.12	22.70	689.7	341.0	511.7	98.74
	EWC	22.85 $\pm$ 0.294	93.31 $\pm$ 0.138	43.42 $\pm$ 0.063				
	Dual Model	81.89 $\pm$ 0.104	96.78 $\pm$ 0.067	91.75 $\pm$ 0.064	320.0 $\pm$ 1.275	431.1 $\pm$ 1.474	273.7 $\pm$ 1.174	12.80 $\pm$ 0.060
	CDVAE	57.70 $\pm$ 4.480	96.73 $\pm$ 0.235	81.10 $\pm$ 1.769	360.9 $\pm$ 20.15	432.1 $\pm$ 0.231	296.4 $\pm$ 7.966	44.29 $\pm$ 4.047
	OCDVAE	80.11 $\pm$ 2.922	97.63 $\pm$ 0.042	91.13 $\pm$ 1.045	345.1 $\pm$ 7.446	430.7 $\pm$ 0.600	280.2 $\pm$ 1.069	25.42 $\pm$ 1.876
A-M-F	**CDVAE ISO**			94.95			269.6	24.97
	CDVAE UB	97.17	89.16	94.91	428.8	311.9	268.2	23.91
	CDVAE LB	00.00	89.72	34.51	506.6	311.0	351.1	34.13
	EWC	3.420 $\pm$ 0.026	87.54 $\pm$ 0.214	45.42 $\pm$ 0.731				
	Dual Model	66.82 $\pm$ 0.337	89.15 $\pm$ 0.050	87.70 $\pm$ 0.102	447.3 $\pm$ 6.700	308.5 $\pm$ 0.599	270.9 $\pm$ 1.299	12.89 $\pm$ 0.109
	CDVAE	79.74 $\pm$ 2.431	88.50 $\pm$ 0.126	89.46 $\pm$ 0.600	448.6 $\pm$ 5.187	315.1 $\pm$ 1.305	281.6 $\pm$ 3.205	33.38 $\pm$ 0.898
	OCDVAE	94.53 $\pm$ 0.283	89.53 $\pm$ 0.367	94.06 $\pm$ 0.156	433.4 $\pm$ 0.424	311.6 $\pm$ 0.353	271.2 $\pm$ 0.424	23.16 $\pm$ 0.121

TABLE 13: Results for PixelVAE based cross-dataset continual learning approaches averaged over 5 runs in analogy to table 12. Extension of table 4 in the main body. Here, in addition to the accuracy α_T, γ_T and KL_T also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of the last increment $T = 3$.

Cross-dataset		α_T(%)			γ_T(nats)			KL_T(nats)
		base	new	all	base	new	all	all
F-M-A	**Dual Pix Model**	82.88 $\pm$ 0.116	97.23 $\pm$ 0.212	92.16 $\pm$ 0.061	288.5 $\pm$ 0.723	437.7 $\pm$ 0.404	251.6 $\pm$ 0.231	9.025 $\pm$ 1.378
	PixCDVAE	56.44 $\pm$ 1.831	97.50 $\pm$ 0.184	80.76 $\pm$ 0.842	289.8 $\pm$ 1.283	438.1 $\pm$ 0.990	252.6 $\pm$ 1.424	29.99 $\pm$ 0.629
	PixOCDVAE	81.84 $\pm$ 0.212	97.75 $\pm$ 0.169	91.76 $\pm$ 0.212	288.8 $\pm$ 0.141	437.1 $\pm$ 0.725	251.8 $\pm$ 0.636	21.07 $\pm$ 0.248
A-M-F	**Dual Pix Model**	71.58 $\pm$ 2.536	88.76 $\pm$ 0.255	88.61 $\pm$ 0.547	445.8 $\pm$ 1.601	290.4 $\pm$ 0.603	255.0 $\pm$ 0.533	9.164 $\pm$ 1.312
	PixCDVAE	49.38 $\pm$ 2.256	88.54 $\pm$ 0.042	82.18 $\pm$ 0.672	441.4 $\pm$ 0.495	287.0 $\pm$ 0.212	252.5 $\pm$ 0.201	30.60 $\pm$ 1.556
	PixOCDVAE	91.90 $\pm$ 0.282	89.91 $\pm$ 0.177	93.82 $\pm$ 0.354	438.5 $\pm$ 1.626	289.4 $\pm$ 0.356	251.3 $\pm$ 0.354	20.35 $\pm$ 0.424

TABLE 14: Results for class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for MNIST at the end of every task increment. Extension of table 3 in the main body. Here, in addition to the accuracy α_t, γ_t and KL_t also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of every task increment t.

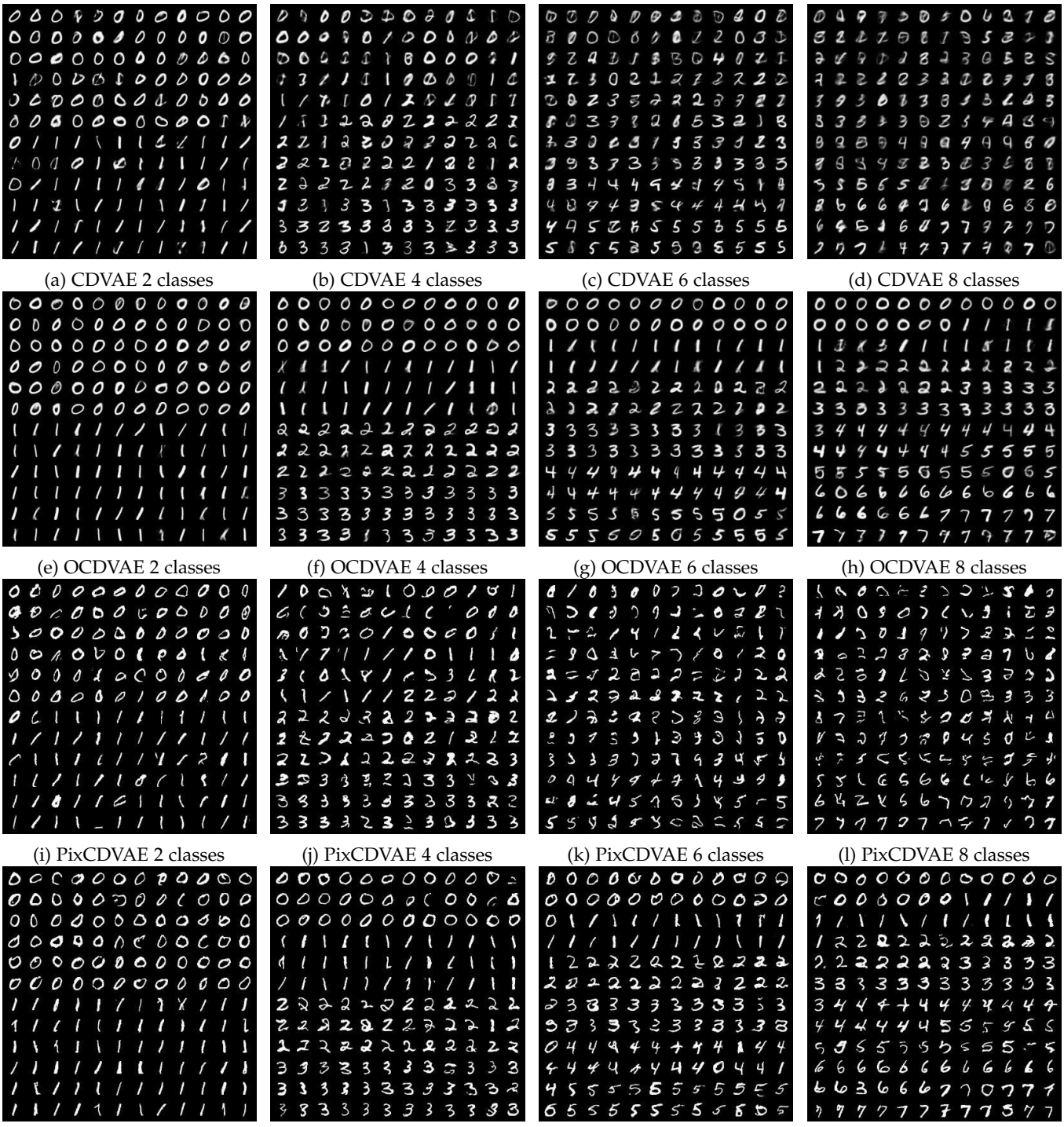
MNIST	**t**	**CDVAE ISO**	**CDVAE UB**	**CDVAE LB**	**EWC**	**Dual Model**	**CDVAE**	**OCDVAE**
$\boldsymbol{\alpha_{base,t}}$ (%)	1		100.0	100.0	99.88 ± 0.010	99.98 ± 0.023	99.97 ± 0.029	99.98 ± 0.018
	2		99.82	00.00	00.61 ± 0.057	99.77 ± 0.032	97.28 ± 3.184	99.30 ± 0.100
	3		99.80	00.00	00.17 ± 0.045	99.51 ± 0.094	87.66 ± 8.765	96.69 ± 2.173
	4		99.85	00.00	00.49 ± 0.017	98.90 ± 0.207	54.70 ± 22.84	94.71 ± 1.792
	5		99.57	00.00	00.45 ± 0.059	97.31 ± 0.489	19.86 ± 7.396	92.53 ± 4.485
$\boldsymbol{\alpha_{new,t}}$ (%)	1		100.0	100.0	99.88 ± 0.010	99.98 ± 0.023	99.97 ± 0.029	99.98 ± 0.018
	2		99.80	99.85	99.70 ± 0.013	99.81 ± 0.062	99.75 ± 0.127	99.80 ± 0.126
	3		99.67	99.94	99.94 ± 0.002	99.48 ± 0.294	99.63 ± 0.172	99.61 ± 0.055
	4		99.49	100.0	99.87 ± 0.015	99.46 ± 0.315	99.05 ± 0.470	99.15 ± 0.032
	5		99.10	99.86	99.58 ± 0.052	98.59 ± 0.106	99.00 ± 0.100	99.06 ± 0.171
$\boldsymbol{\alpha_{all,t}}$ (%)	1		100.0	100.0	99.88 ± 0.010	99.98 ± 0.023	99.97 ± 0.029	99.98 ± 0.018
	2		99.81	49.92	50.16 ± 0.029	99.79 ± 0.049	98.54 ± 1.638	99.55 ± 0.036
	3		99.72	31.35	33.42 ± 0.027	99.32 ± 0.057	95.01 ± 3.162	98.46 ± 0.903
	4		99.50	24.82	25.36 ± 0.025	98.56 ± 0.021	81.50 ± 9.369	97.06 ± 1.069
	5	99.45	99.29	20.16	20.26 ± 0.027	96.64 ± 0.079	64.34 ± 4.903	93.24 ± 3.742
$\boldsymbol{\gamma_{base,t}}$ (nats)	1		63.18	62.08		62.17 ± 0.979	64.34 ± 2.054	62.53 ± 1.166
	2		62.85	126.8		63.69 ± 0.576	74.41 ± 10.89	65.68 ± 1.166
	3		63.36	160.4		67.34 ± 0.445	81.89 ± 10.09	69.29 ± 1.541
	4		64.25	126.9		70.41 ± 0.436	90.62 ± 10.08	71.69 ± 1.379
	5		64.99	123.2		75.08 ± 0.623	101.6 ± 8.347	77.16 ± 1.104
$\boldsymbol{\gamma_{new,t}}$ (nats)	1		63.18	62.08		62.17 ± 0.979	64.34 ± 2.054	62.53 ± 1.166
	2		88.75	87.93		88.03 ± 0.664	89.91 ± 0.107	89.64 ± 3.709
	3		82.53	87.22		83.46 ± 0.992	87.65 ± 0.530	85.37 ± 1.725
	4		72.68	74.61		73.23 ± 0.280	79.49 ± 0.489	74.75 ± 0.777
	5		85.88	92.00		89.32 ± 0.626	93.55 ± 0.391	89.68 ± 0.618
$\boldsymbol{\gamma_{all,t}}$ (nats)	1		63.18	62.08		62.17 ± 0.979	64.34 ± 2.054	62.53 ± 1.166
	2		75.97	107.3		75.64 ± 0.600	82.02 ± 5.488	76.62 ± 1.695
	3		79.58	172.3		81.24 ± 0.262	89.88 ± 3.172	82.95 ± 1.878
	4		79.72	203.1		82.92 ± 0.489	95.83 ± 2.747	85.30 ± 1.524
	5	78.12	81.97	163.7		88.29 ± 0.363	107.6 ± 1.724	92.92 ± 2.283
$\boldsymbol{KL_{all,t}}$ (nats)	1		12.55	13.08		11.81 ± 0.123	13.00 ± 0.897	13.68 ± 0.785
	2		18.50	25.84		16.15 ± 0.149	20.20 ± 1.188	18.01 ± 0.154
	3		20.16	24.28		16.46 ± 0.122	24.24 ± 1.974	20.02 ± 0.161
	4		20.48	26.32		16.09 ± 0.177	27.01 ± 1.851	20.26 ± 0.186
	5	22.12	21.02	24.87		16.13 ± 0.225	30.61 ± 1.240	21.02 ± 0.717

TABLE 15: Results for class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for FashionMNIST at the end of every task increment. Extension of table 3 in the main body. Here, in addition to the accuracy α_t, γ_t and KL_t also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of every task increment t.

Fashion	**t**	**CDVAE ISO**	**CDVAE UB**	**CDVAE LB**	**EWC**	**Dual Model**	**CDVAE**	**OCDVAE**
	1		99.65	99.60	99.17 ± 0.037	99.58 ± 0.062	99.55 ± 0.035	99.59 ± 0.082
	2		96.70	00.00	02.40 ± 0.122	94.50 ± 0.389	92.02 ± 1.175	92.36 ± 2.092
$\boldsymbol{\alpha_{base,t}}$	3		95.95	00.00	01.63 ± 0.032	94.88 ± 0.432	79.26 ± 4.170	83.90 ± 2.310
(%)	4		91.35	00.00	00.33 ± 0.097	82.25 ± 4.782	50.16 ± 6.658	64.70 ± 2.580
	5		92.20	00.00	00.17 ± 0.076	94.26 ± 0.192	39.51 ± 7.173	60.63 ± 12.16
	1		99.65	99.60	99.17 ± 0.037	99.58 ± 0.062	99.55 ± 0.035	99.59 ± 0.082
	2		95.55	97.95	96.09 ± 0.260	89.31 ± 0.311	90.98 ± 0.626	92.64 ± 2.302
$\boldsymbol{\alpha_{new,t}}$	3		93.35	99.95	99.92 ± 0.012	86.06 ± 2.801	90.26 ± 1.435	83.40 ± 3.089
(%)	4		84.75	99.90	99.95 ± 0.060	73.63 ± 3.861	85.65 ± 2.127	84.18 ± 2.715
	5		97.50	99.80	99.60 ± 0.023	93.55 ± 0.708	96.92 ± 0.774	96.51 ± 0.707
	1		99.65	99.60	99.17 ± 0.037	99.58 ± 0.062	99.55 ± 0.035	99.59 ± 0.082
	2		95.75	48.97	49.28 ± 0.242	91.91 ± 0.043	91.83 ± 0.730	92.31 ± 1.163
$\boldsymbol{\alpha_{all,t}}$	3		93.02	33.33	34.34 ± 0.009	79.98 ± 0.634	83.35 ± 1.597	86.93 ± 0.870
(%)	4		87.51	25.00	25.21 ± 0.100	64.37 ± 0.707	64.66 ± 3.204	76.05 ± 1.391
	5	89.54	89.24	19.97	20.06 ± 0.059	63.21 ± 1.957	58.82 ± 2.521	69.88 ± 1.712
	1		209.7	209.8		207.7 ± 1.558	208.9 ± 1.213	209.7 ± 3.655
	2		207.4	240.7		209.0 ± 0.731	212.7 ± 0.579	212.1 ± 0.937
$\boldsymbol{\gamma_{base,t}}$	3		207.6	258.7		213.0 ± 1.854	219.5 ± 1.376	216.9 ± 1.208
(nats)	4		207.7	243.6		213.6 ± 0.509	223.8 ± 0.837	217.1 ± 0.979
	5		208.4	306.5		217.7 ± 1.510	232.8 ± 5.048	222.8 ± 1.632
	1		209.7	209.8		207.7 ± 1.558	208.9 ± 1.213	209.7 ± 3.655
	2		241.1	240.2		238.7 ± 0.081	241.8 ± 0.502	241.9 ± 0.960
$\boldsymbol{\gamma_{new,t}}$	3		213.6	211.8		211.6 ± 0.543	215.4 ± 0.501	213.0 ± 0.635
(nats)	4		220.5	219.7		219.5 ± 0.216	223.6 ± 0.381	220.9 ± 0.522
	5		246.2	242.0		242.8 ± 0.898	248.8 ± 0.398	244.0 ± 0.646
	1		209.7	209.8		207.7 ± 1.558	208.9 ± 1.213	209.7 ± 3.655
	2		224.2	240.4		223.8 ± 0.402	226.6 ± 2.31	226.9 ± 0.918
$\boldsymbol{\gamma_{all,t}}$	3		220.7	246.1		221.9 ± 0.648	227.2 ± 0.606	224.9 ± 0.642
(nats)	4		220.4	238.7		225.1 ± 3.629	230.4 ± 0.524	226.1 ± 0.560
	5	224.8	226.2	275.1		230.5 ± 1.543	242.2 ± 0.754	234.6 ± 0.823
	1		12.17	12.20		9.710 ± 0.345	13.21 ± 0.635	13.28 ± 0.644
	2		16.54	17.47		10.65 ± 0.101	17.60 ± 0.755	15.56 ± 0.696
$\boldsymbol{KL_{all,t}}$	3		18.84	19.34		11.34 ± 0.057	21.25 ± 0.872	17.35 ± 0.307
(nats)	4		20.06	17.31		10.96 ± 0.106	25.21 ± 0.929	19.81 ± 0.462
	5	23.27	20.27	21.61		11.45 ± 0.228	26.68 ± 0.859	20.47 ± 0.742

TABLE 16: Results for class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for AudioMNIST at the end of every task increment. Extension of table 3 in the main body. Here, in addition to the accuracy α_t, γ_t and KL_t also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of every task increment t.

Audio	t	CDVAE ISO	CDVAE UB	CDVAE LB	EWC	Dual Model	CDVAE	OCDVAE
$\boldsymbol{\alpha_{base,t}}$ (%)	1		99.99	100.0	100.0 ± 0.000	100.0 ± 0.000	99.21 ± 0.568	99.95 ± 0.035
	2		99.92	00.00	00.16 ± 0.040	93.08 ± 5.854	98.98 ± 0.766	98.61 ± 0.490
	3		100.0	00.00	00.29 ± 0.029	83.25 ± 6.844	92.44 ± 1.306	95.12 ± 2.248
	4		99.92	00.00	00.31 ± 0.015	72.02 ± 0.677	76.43 ± 4.715	86.37 ± 5.63
	5		98.42	00.00	00.11 ± 0.007	61.57 ± 0.747	59.36 ± 7.147	79.73 ± 4.070
$\boldsymbol{\alpha_{new,t}}$ (%)	1		99.99	100.0	100.0 ± 0.000	100.0 ± 0.000	99.21 ± 0.568	99.95 ± 0.035
	2		99.75	100.0	99.78 ± 0.019	86.25 ± 8.956	91.82 ± 4.577	89.23 ± 7.384
	3		98.92	99.58	99.25 ± 0.054	95.16 ± 1.490	95.20 ± 1.495	94.43 ± 3.030
	4		97.33	98.67	97.03 ± 0.019	62.52 ± 4.022	53.02 ± 6.132	72.22 ± 8.493
	5		98.67	100.0	99.41 ± 0.207	89.41 ± 0.691	84.93 ± 6.297	89.52 ± 6.586
$\boldsymbol{\alpha_{all,t}}$ (%)	1		99.99	100.0	100.0 ± 0.000	100.0 ± 0.000	99.21 ± 0.568	99.95 ± 0.035
	2		99.83	50.00	50.16 ± 0.119	89.67 ± 1.763	93.84 ± 2.558	93.93 ± 3.756
	3		99.56	33.19	33.28 ± 0.022	78.24 ± 3.315	94.26 ± 1.669	95.70 ± 1.524
	4		98.60	24.58	24.50 ± 0.017	60.43 ± 4.209	77.90 ± 4.210	85.59 ± 3.930
	5	97.75	97.87	20.02	19.98 ± 0.032	47.42 ± 1.447	81.49 ± 1.944	87.72 ± 1.594
$\boldsymbol{\gamma_{base,t}}$ (nats)	1		433.7	423.2		422.3 ± 0.573	435.2 ± 15.69	424.2 ± 2.511
	2		422.5	439.4		426.6 ± 2.840	423.9 ± 0.517	425.2 ± 1.402
	3		420.7	429.2		425.0 ± 0.339	422.7 ± 0.690	423.8 ± 1.148
	4		419.9	428.5		425.4 ± 0.081	422.8 ± 0.367	423.5 ± 0.937
	5		418.4	432.9		425.2 ± 0.244	422.7 ± 0.182	423.5 ± 0.586
$\boldsymbol{\gamma_{new,t}}$ (nats)	1		433.7	423.2		422.3 ± 0.573	435.2 ± 15.69	424.2 ± 2.511
	2		381.2	384.1		381.3 ± 2.039	382.5 ± 1.355	385.3 ± 12.56
	3		435.9	436.7		436.8 ± 0.188	436.3 ± 0.639	436.9 ± 0.688
	4		485.9	487.1		486.5 ± 0.432	486.7 ± 0.385	486.5 ± 0.701
	5		421.3	425.2		422.4 ± 0.784	423.9 ± 0.681	422.9 ± 0.537
$\boldsymbol{\gamma_{all,t}}$ (nats)	1		433.7	423.2		422.3 ± 0.573	435.2 ± 15.69	424.2 ± 2.511
	2		401.9	411.8		404.0 ± 2.407	403.2 ± 0.831	403.5 ± 1.274
	3		412.1	418.9		414.4 ± 0.385	413.6 ± 0.410	413.8 ± 0.573
	4		430.3	438.4		433.9 ± 0.374	432.4 ± 0.436	432.6 ± 0.862
	5	429.7	427.2	440.4		432.7 ± 0.385	431.4 ± 0.255	430.9 ± 0.541
$\boldsymbol{KL_{all,t}}$ (nats)	1		11.65	11.20		4.639 ± 0.107	11.78 ± 1.478	11.16 ± 0.713
	2		11.78	13.61		5.135 ± 0.127	15.13 ± 1.128	14.06 ± 1.140
	3		13.40	17.09		5.427 ± 0.105	18.18 ± 1.140	13.61 ± 0.901
	4		13.61	14.41		5.243 ± 0.135	22.93 ± 1.134	17.58 ± 1.102
	5	17.89	15.15	14.52		5.470 ± 0.055	22.96 ± 0.912	18.52 ± 1.131



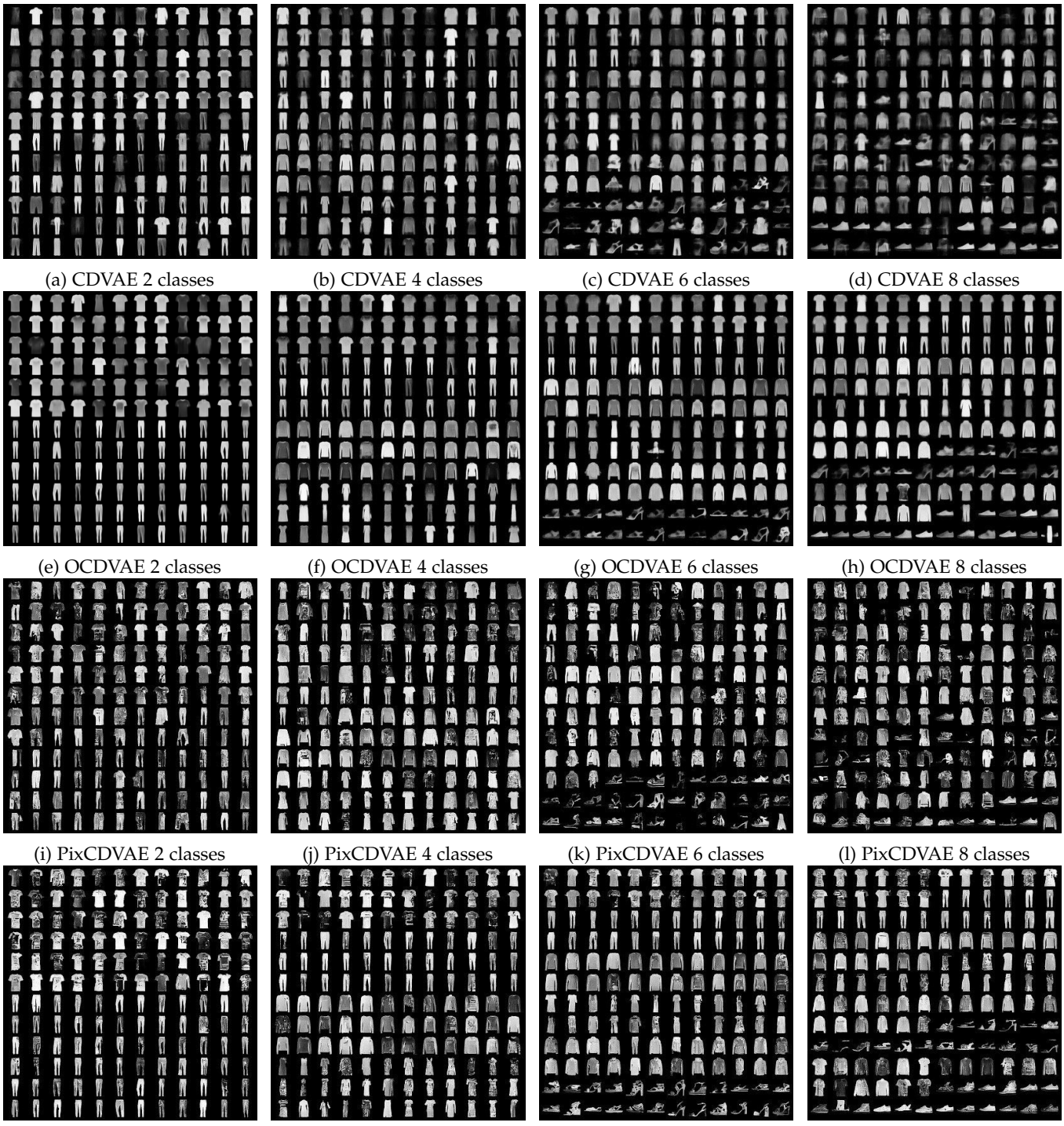
(a) CDVAE 2 classes (b) CDVAE 4 classes (c) CDVAE 6 classes (d) CDVAE 8 classes

(e) OCDVAE 2 classes (f) OCDVAE 4 classes (g) OCDVAE 6 classes (h) OCDVAE 8 classes

(i) PixCDVAE 2 classes (j) PixCDVAE 4 classes (k) PixCDVAE 6 classes (l) PixCDVAE 8 classes

(m) PixOCDVAE 2 classes (n) PixOCDVAE 4 classes (o) PixOCDVAE 6 classes (p) PixOCDVAE 8 classes

Fig. 13: Generated images for continually learned incremental MNIST at the end of task increments for CDVAE (a-d), OCDVAE (e-h), PixCDVAE (i-l) and PixOCDVAE (m-p). Each individual grid is sorted according to the class label that is predicted by the classifier.



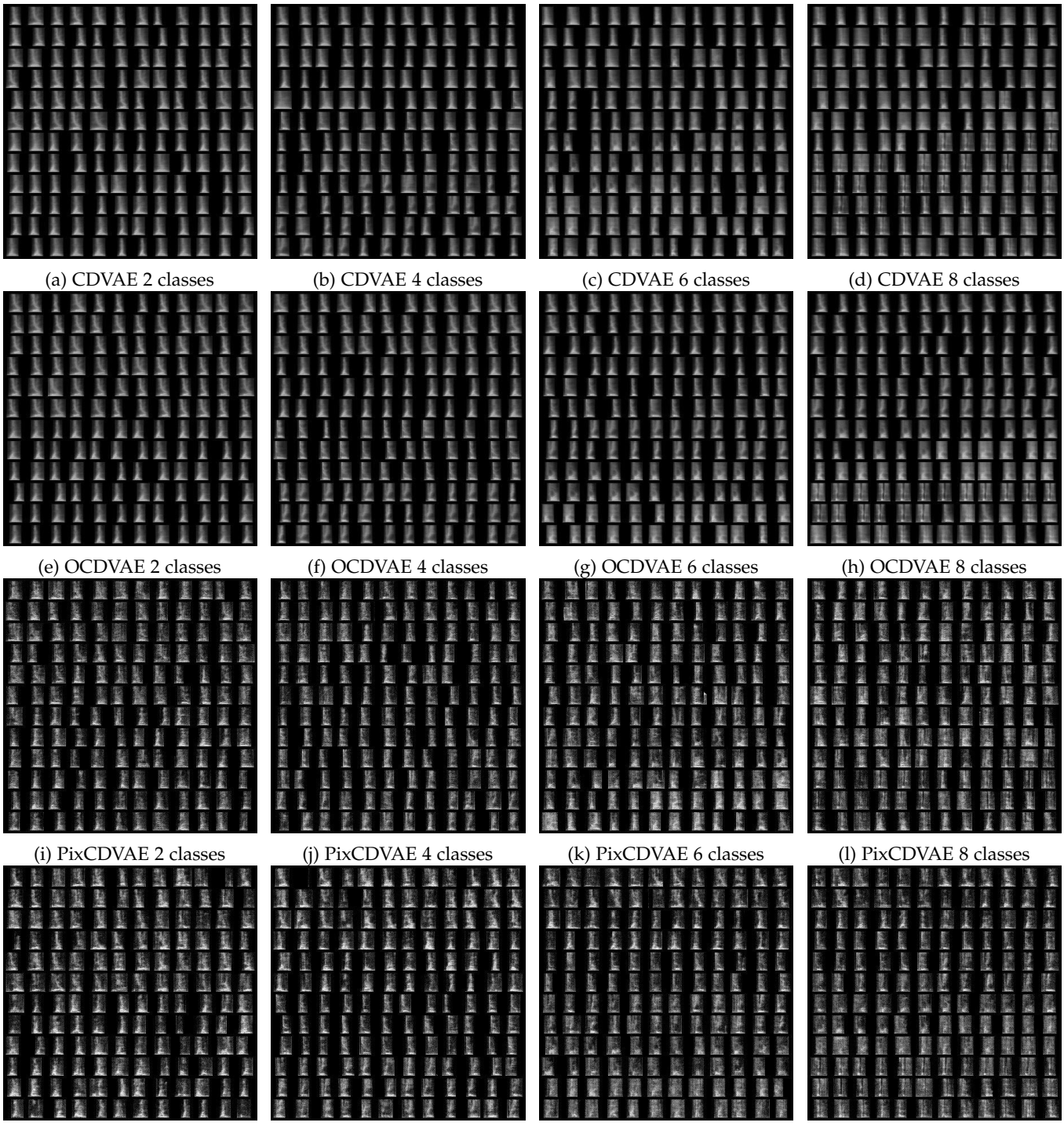
(a) CDVAE 2 classes (b) CDVAE 4 classes (c) CDVAE 6 classes (d) CDVAE 8 classes

(e) OCDVAE 2 classes (f) OCDVAE 4 classes (g) OCDVAE 6 classes (h) OCDVAE 8 classes

(i) PixCDVAE 2 classes (j) PixCDVAE 4 classes (k) PixCDVAE 6 classes (l) PixCDVAE 8 classes

(m) PixOCDVAE 2 classes (n) PixOCDVAE 4 classes (o) PixOCDVAE 6 classes (p) PixOCDVAE 8 classes

Fig. 14: Generated images for continually learned incremental FashionMNIST at the end of task increments for CDVAE (a-d), OCDVAE (e-h), PixCDVAE (i-l) and PixOCDVAE (m-p). Each individual grid is sorted according to the class label that is predicted by the classifier.



(a) CDVAE 2 classes (b) CDVAE 4 classes (c) CDVAE 6 classes (d) CDVAE 8 classes

(e) OCDVAE 2 classes (f) OCDVAE 4 classes (g) OCDVAE 6 classes (h) OCDVAE 8 classes

(i) PixCDVAE 2 classes (j) PixCDVAE 4 classes (k) PixCDVAE 6 classes (l) PixCDVAE 8 classes

(m) PixOCDVAE 2 classes (n) PixOCDVAE 4 classes (o) PixOCDVAE 6 classes (p) PixOCDVAE 8 classes

Fig. 15: Generated images for continually learned incremental AudioMNIST at the end of task increments for CDVAE (a-d), OCDVAE (e-h), PixCDVAE (i-l) and PixOCDVAE (m-p). Each individual grid is sorted according to the class label that is predicted by the classifier.

TABLE 17: Results for PixelVAE based class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for MNIST at the end of every task increment in analogy to table 14. Extension of table 4 in the main body. Here, in addition to the accuracy α_t, γ_t and KL_t also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of every task increment t.

MNIST	t	Dual Pix Model	PixCDVAE	PixOCDVAE
$\boldsymbol{\alpha_{base,t}}$ (%)	1	99.97 ± 0.002	99.97 ± 0.026	99.86 ± 0.084
	2	99.54 ± 0.285	96.90 ± 2.907	99.64 ± 0.095
	3	99.16 ± 0.611	90.12 ± 5.846	98.88 ± 0.491
	4	98.33 ± 1.119	76.84 ± 9.095	98.11 ± 0.797
	5	98.04 ± 1.397	56.53 ± 4.032	97.44 ± 0.785
$\boldsymbol{\alpha_{new,t}}$ (%)	1	99.97 ± 0.002	99.97 ± 0.026	99.86 ± 0.084
	2	99.71 ± 0.122	99.74 ± 0.052	99.82 ± 0.027
	3	99.41 ± 0.084	99.22 ± 0.082	99.56 ± 0.092
	4	98.61 ± 0.312	97.84 ± 0.180	98.80 ± 0.292
	5	97.31 ± 0.575	96.77 ± 0.337	98.63 ± 0.430
$\boldsymbol{\alpha_{all,t}}$ (%)	1	99.97 ± 0.002	99.97 ± 0.026	99.86 ± 0.084
	2	99.60 ± 0.142	98.37 ± 1.448	99.69 ± 0.051
	3	98.93 ± 0.291	96.14 ± 1.836	99.20 ± 0.057
	4	98.22 ± 0.560	91.25 ± 0.992	98.13 ± 0.281
	5	96.52 ± 0.658	83.61 ± 0.927	96.84 ± 0.346
$\boldsymbol{\gamma_{base,t}}$ (nats)	1	90.52 ± 0.263	100.0 ± 1.572	99.77 ± 2.768
	2	91.27 ± 0.789	100.4 ± 1.964	101.2 ± 3.601
	3	91.92 ± 0.991	100.3 ± 4.562	101.1 ± 4.014
	4	91.75 ± 1.136	102.7 ± 7.134	101.0 ± 4.573
	5	92.05 ± 1.212	102.4 ± 6.195	100.5 ± 4.942
$\boldsymbol{\gamma_{new,t}}$ (nats)	1	90.52 ± 0.263	100.0 ± 1.572	99.77 ± 2.768
	2	115.8 ± 0.805	125.7 ± 2.413	124.6 ± 3.822
	3	107.7 ± 0.600	118.3 ± 3.523	116.5 ± 2.219
	4	100.9 ± 0.659	107.1 ± 5.316	102.3 ± 1.844
	5	113.4 ± 0.820	118.2 ± 1.572	113.3 ± 0.755
$\boldsymbol{\gamma_{all,t}}$ (nats)	1	90.52 ± 0.263	100.0 ± 1.572	99.77 ± 2.768
	2	102.9 ± 0.408	111.9 ± 2.627	112.7 ± 3.300
	3	104.8 ± 1.114	114.9 ± 4.590	114.6 ± 4.788
	4	103.9 ± 0.759	114.3 ± 3.963	112.1 ± 2.150
	5	106.1 ± 0.868	118.7 ± 5.320	111.9 ± 2.663
$\boldsymbol{KL_{all,t}}$ (nats)	1	1.410 ± 0.181	5.629 ± 3.749	5.635 ± 3.739
	2	3.177 ± 0.702	9.238 ± 0.674	7.495 ± 0.738
	3	4.923 ± 1.085	12.13 ± 0.977	10.17 ± 1.528
	4	5.603 ± 1.250	14.32 ± 1.040	11.66 ± 1.004
	5	9.296 ± 1.346	16.37 ± 0.970	12.49 ± 0.551

TABLE 18: Results for PixelVAE based class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for FashionMNIST at the end of every task increment in analogy to table 15. Extension of table 4 in the main body. Here, in addition to the accuracy α_t, γ_t and KL_t also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of every task increment t.

Fashion	t	Dual Pix Model	PixCDVAE	PixOCDVAE
$\boldsymbol{\alpha_{base,t}}$ (%)	1	99.57 ± 0.091	99.58 ± 0.076	99.54 ± 0.079
	2	82.40 ± 6.688	90.06 ± 1.782	88.60 ± 1.998
	3	78.55 ± 3.964	83.70 ± 3.571	87.66 ± 0.375
	4	54.69 ± 3.853	50.23 ± 7.004	68.31 ± 3.308
	5	60.04 ± 5.151	47.83 ± 13.41	74.45 ± 2.889
$\boldsymbol{\alpha_{new,t}}$ (%)	1	99.57 ± 0.091	99.58 ± 0.076	99.54 ± 0.079
	2	97.73 ± 1.113	96.47 ± 0.596	97.31 ± 0.475
	3	99.09 ± 0.367	97.33 ± 0.725	96.88 ± 1.156
	4	97.55 ± 0.588	96.12 ± 0.675	95.47 ± 1.332
	5	98.85 ± 0.141	97.91 ± 0.596	98.63 ± 0.176
$\boldsymbol{\alpha_{all,t}}$ (%)	1	99.57 ± 0.091	99.58 ± 0.076	99.54 ± 0.079
	2	86.22 ± 3.704	92.93 ± 0.160	92.17 ± 1.425
	3	76.77 ± 4.378	84.07 ± 1.069	87.30 ± 0.322
	4	62.93 ± 3.738	64.42 ± 1.837	76.36 ± 1.267
	5	72.41 ± 2.941	63.05 ± 1.826	80.85 ± 0.721
$\boldsymbol{\gamma_{base,t}}$ (nats)	1	267.8 ± 1.246	230.8 ± 3.024	232.0 ± 2.159
	2	273.6 ± 0.631	232.5 ± 1.582	231.8 ± 0.416
	3	274.0 ± 0.552	235.6 ± 2.784	231.6 ± 0.832
	4	273.7 ± 0.504	236.4 ± 3.157	231.4 ± 2.550
	5	274.1 ± 0.349	241.1 ± 1.747	234.1 ± 1.498
$\boldsymbol{\gamma_{new,t}}$ (nats)	1	267.8 ± 1.246	230.8 ± 3.024	232.0 ± 2.159
	2	313.4 ± 1.006	275.8 ± 1.888	275.3 ± 1.473
	3	269.1 ± 0.616	268.3 ± 3.852	262.9 ± 1.893
	4	282.4 ± 0.321	259.1 ± 1.305	259.6 ± 2.050
	5	305.8 ± 0.286	283.2 ± 2.150	283.5 ± 2.458
$\boldsymbol{\gamma_{all,t}}$ (nats)	1	267.8 ± 1.246	230.8 ± 3.024	232.0 ± 2.159
	2	293.8 ± 0.349	254.3 ± 1.513	255.8 ± 0.436
	3	285.7 ± 0.510	261.5 ± 2.970	259.1 ± 0.929
	4	284.9 ± 0.703	263.2 ± 2.259	259.5 ± 3.218
	5	289.5 ± 0.396	271.7 ± 2.117	267.2 ± 0.586
$\boldsymbol{KL_{all,t}}$ (nats)	1	3.610 ± 0.856	7.164 ± 0.759	7.809 ± 1.255
	2	6.247 ± 0.710	13.79 ± 0.282	12.23 ± 0.287
	3	7.811 ± 0.799	18.26 ± 0.818	15.36 ± 0.530
	4	8.982 ± 0.812	21.75 ± 0.561	18.31 ± 0.333
	5	9.781 ± 1.068	22.14 ± 0.377	17.93 ± 0.360

TABLE 19: Results for PixelVAE based class incremental continual learning approaches averaged over 5 runs, baselines and the reference isolated learning scenario for AudioMNIST at the end of every task increment in analogy to table 16. Extension of table 4 in the main body. Here, in addition to the accuracy α_t, γ_t and KL_t also indicate the respective NLL reconstruction metrics and corresponding KL divergences at the end of every task increment t.

Audio	**t**	**Dual Pix Model**	**PixCDVAE**	**PixOCDVAE**
$\boldsymbol{\alpha_{base,t}}$ (%)	1	100.0 ± 0.000	99.71 ± 0.218	99.27 ± 0.410
	2	99.52 ± 0.273	97.86 ± 0.799	97.88 ± 2.478
	3	93.15 ± 3.062	81.38 ± 5.433	95.82 ± 3.602
	4	81.55 ± 8.468	50.58 ± 14.60	91.56 ± 5.640
	5	64.60 ± 8.739	29.94 ± 18.47	75.25 ± 10.18
$\boldsymbol{\alpha_{new,t}}$ (%)	1	100.0 ± 0.000	99.71 ± 0.218	99.27 ± 0.410
	2	99.71 ± 0.043	99.78 ± 0.128	99.81 ± 0.189
	3	98.23 ± 1.092	98.41 ± 0.507	99.30 ± 0.550
	4	95.31 ± 0.868	94.30 ± 0.914	97.87 ± 0.293
	5	98.18 ± 0.885	97.00 ± 0.520	99.43 ± 0.495
$\boldsymbol{\alpha_{all,t}}$ (%)	1	100.0 ± 0.000	99.71 ± 0.218	99.27 ± 0.410
	2	99.50 ± 0.157	98.64 ± 0.875	99.67 ± 0.033
	3	95.37 ± 1.750	90.10 ± 1.431	97.77 ± 1.017
	4	86.97 ± 2.797	75.55 ± 3.891	95.41 ± 1.345
	5	75.50 ± 3.032	63.44 ± 5.252	90.23 ± 1.139
$\boldsymbol{\gamma_{base,t}}$ (nats)	1	434.2 ± 1.068	432.6 ± 0.321	433.8 ± 0.370
	2	434.4 ± 1.082	432.5 ± 0.551	433.5 ± 1.464
	3	434.6 ± 0.785	432.9 ± 0.723	433.1 ± 1.269
	4	434.2 ± 1.209	433.0 ± 0.781	433.0 ± 1.283
	5	435.1 ± 1.915	431.4 ± 0.666	432.3 ± 0.189
$\boldsymbol{\gamma_{new,t}}$ (nats)	1	434.2 ± 1.068	432.6 ± 0.321	433.8 ± 0.370
	2	390.4 ± 0.694	389.4 ± 0.208	389.4 ± 1.304
	3	444.7 ± 0.545	442.7 ± 0.513	442.4 ± 0.275
	4	497.4 ± 0.740	494.4 ± 0.700	494.8 ± 0.386
	5	431.9 ± 1.032	428.0 ± 0.851	429.7 ± 1.223
$\boldsymbol{\gamma_{all,t}}$ (nats)	1	435.2 ± 15.69	432.6 ± 0.321	433.8 ± 0.370
	2	412.4 ± 0.871	410.9 ± 0.351	411.5 ± 1.406
	3	423.3 ± 0.618	421.0 ± 1.026	421.9 ± 0.661
	4	441.6 ± 0.420	439.8 ± 0.833	439.8 ± 0.718
	5	440.3 ± 1.297	436.9 ± 0.751	437.7 ± 0.432
$\boldsymbol{KL_{all,t}}$ (nats)	1	4.361 ± 0.671	9.293 ± 0.943	11.87 ± 1.504
	2	5.130 ± 0.636	14.00 ± 0.748	12.40 ± 0.719
	3	5.399 ± 0.724	20.28 ± 0.774	14.41 ± 0.461
	4	5.817 ± 1.038	24.91 ± 0.845	16.00 ± 0.505
	5	6.031 ± 0.832	27.14 ± 1.139	17.45 ± 0.835