The attempt to recreate human neural capacity for life-long learning is a central challenge in Artificial Intelligence. After all, humans master countless tasks in succession without incurring catastrophic loss for past tasks. Nonetheless, state of the art Deep Neural Networks relying on a naïve version of the backpropagation algorithm are entirely unable to learn cumulatively when the data for previous tasks is no longer available. To guarantee optimal performance on sequential tasks, the conventional solution has been to store all learned (old) data and continuously interleave old and new as the network is further trained. However, this method is extremely memory expensive, requiring storage of all samples ever encountered. Generative models have been employed to model old data and bypass the memory constraint of real data storage. Still, these methods so far either employ separate generators for each new task, or distill knowledge to successive new generator networks [1], thus encountering serious scalability issues as well as biological implausibility insofar as the human brain cannot make an "intermediate copy" of itself to distill knowledge.

In contrast, we propose a method that approximates a cumulative generative model and continuous embedded classifier. Moreover, we evaluate incremental learning using a notoriously hard paradigm, "single headed learning". Here, each task is a disjoint subset of classes in the overall dataset and the performance is evaluated as to classification of all previous classes. To account for the growing number of classes, we create extra output nodes which are incrementally used. Previous methods of weight regularizations often perform poorly at this task since the mapping of the output layer does not have a fixed size and this creates a weight rearrangement that is difficult to predict and regularize [2].

In the model, we make use of an Auxiliary Conditional Generative Adversarial Network (AC-GAN) [3] as the generative network because of its ability to generate high quality images as well as its robust classification capacity after careful hyper-parameter tuning. As a new task is learned, the old data is approximated by continuously sampling from the generator. Of course, since a new task also modifies the parametrization of the generator, this procedure cannot be applied without some guarantee that the generated images are reasonable approximations of the old distribution that has been lost. Our method tackles this issue by first, using the embedded classifier to categorize the sampled images and only allow correctly classified images through. Second, we fill a small buffer with samples drawn from the generator before training the new task, thus representing stable samples of the past classes. This buffer can also be constructed from a small fraction of the original past data, with the same memory requirement. The sampling for the old data is then always a freshly sampled generator output of correct label as well as a small fraction of stable buffer samples that enforce "smoothness" in the representation of the old classes.

As a baseline, in the scenario where there is no replay whatsoever, catastrophic forgetting occurs and the accuracy lowers to less than 20% after 5 disjoint tasks of 2 classes on both MNIST and Fashion-MNIST, meaning the DNN only retains the last 2 newly learned classes. On the other hand, our model achieves 84% accuracy using Fashion-MNIST and 93% using MNIST with buffers under 1% of the original dataset size according to preliminary results. Experiments indicate that sampling from the stochastic generator confers higher variability for training than using only a fixed buffer. Moreover, we show that our method outperforms multi-task training when using only buffer data. This difference is further augmented for very small buffers since the stochasticity in the model conveys larger effects when the accuracy due to the buffer alone is lowered.

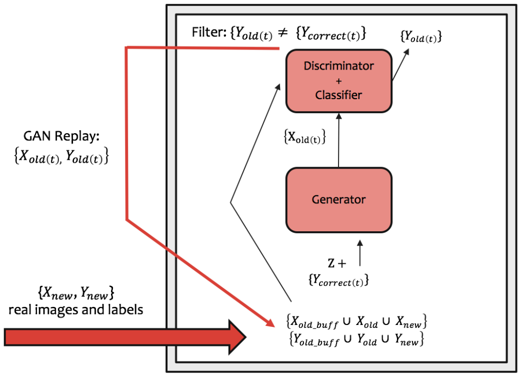
Filter: $\{Y_{old(t)} \neq \{Y_{correct(t)}\}$
$\{Y_{old(t)}\}$
Discriminator
+
Classifier
GAN Replay:
$\{X_{old(t)}, Y_{old(t)}\}$
$\{X_{old(t)}\}$
Generator
Z +
$\{Y_{correct(t)}\}$
$\{X_{new}, Y_{new}\}$
real images and labels
$\{X_{old\_buff} \cup X_{old} \cup X_{new}\}$
$\{Y_{old\_buff} \cup Y_{old} \cup Y_{new}\}$

Figure: Model used for cumulative and continual learning. Past data is sampled from the generator and filtered by the embedded classifier. Old data is a combination of the fresh generator output and a small buffer used to smoothen the old data distribution to guarantee the maintenance of quality output.

[1]. Shin, Hanul, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. "Continual Learning with Deep Generative Replay." ArXiv:1705.08690 [Cs], May 24, 2017. http://arxiv.org/abs/1705.08690.

[2]. Farquhar, Sebastian, and Yarin Gal. "Towards Robust Evaluations of Continual Learning." ArXiv:1805.09733 [Cs, Stat], May 24, 2018. http://arxiv.org/abs/1805.09733.

[3]. Odena, Augustus, Christopher Olah, and Jonathon Shlens. "Conditional Image Synthesis With Auxiliary Classifier GANs." ArXiv:1610.09585 [Cs, Stat], October 29, 2016. http://arxiv.org/abs/1610.09585.