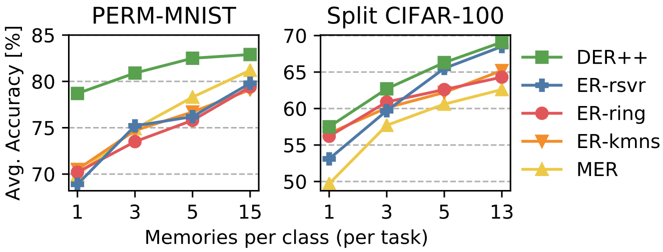

Figure A: Comparison with Fig. 1 of *Chaudhry et al.*

Table A: Single-epoch evaluation setting (Class-IL).

	Buffer	ER	FDR	DER++	JOINT	JOINT
#epochs		1	1	1	1	50/100
Seq. CIFAR 10	200	37.64	21.22	**41.93**	56.74	92.20
	500	45.22	21.06	**48.04**		
	5120	50.28	20.57	**53.31**		
Seq. Tiny ImgNet	200	5.98	4.87	**6.35**	19.37	59.99
	500	8.39	4.76	**8.65**		
	5120	16.04	4.96	**16.41**		

R1 ***[Misleading comparisons]*** We were probably not clear about the Class-IL protocol (we will clarify Sec. 4.1): if a method needs task boundaries (*e.g.* oEWC or LwF), we always provide them during training. Hence, we consider our comparison fair and in line with other works [39, 32, 41, 11]. Moreover, FDR regularizes from the second task onwards, which makes it indistinguishable from SGD during the first one — like EWC, SI, LwF, GEM and others.

R2 ***[On Tiny Episodic Memories in Continual Learning]*** We already draw a thorough comparison with ER-Reservoir, which, according to Chaudry et al., "*works the best across the board* [among ER baselines] *except when the memory size is very small*". However, we here provide a comparison with the experiments of Chaudhry et al. (Fig. A), showing that DER++, despite relying on reservoir, outperforms all ER-based methods even for the smallest memory sizes provided.

R2, R3 ***[Number of epochs]*** We conceive a task as a mere sequence of batches drawn from the dataset: in the case of multiple epochs, this sequence simply results longer to the model. In doing so, our implementation decouples the length of the task from the notion of task boundary: for the latter, an oracle notifies the model about the end of the task through a *callback*. As an example, EWC exploits this callback to compute the Fisher Information Matrix; instead, our method ignores this call as it does not need to take any action at boundaries. Under this perspective, the chance of doing multiple epochs results orthogonal to the use of task boundaries.

Although the hint about the investigation of a single-epoch setup is compelling, we believe that it could be problematic for difficult datasets such as CIFAR-10 and TinyImageNet. If we limit ourselves to a single pass (few gradient steps), we struggle to disentangle the effects of catastrophic forgetting (the focus of our work) from those of underfitting. The single-epoch experiment asked by the reviewers (Tab. A) reveals indeed that even the joint training (upper bound) yields a dramatically low accuracy w.r.t. multi-epoch (see last two columns). Among CL methods, DER++ confirms its reliability approaching the single-epoch joint training. Nevertheless, we feel that future GCL works should be conscious of the above-mentioned points when dealing with single-epoch split setups (we will add these considerations in the appendix). Our MNIST-360 (the first to our knowledge to match the requirements of GCL [10]) points in this direction.

R2, R4 ***[Implementation]*** Whenever available, we referred to our competitors' official repositories; additionally, we validated our implementations by matching the results of the original papers in their specific experimental settings. While R2 is satisfied with our codebase after having reviewed it, the doubts of R4 are due to a mismatch between the performance of our iCaRL implementation and the results presented in the original paper [32]. We ascribe this to some differences in the experimental protocol: i) Rebuffi et al. tested on CIFAR-100, whereas we used CIFAR-10; ii) they relied on ResNet-32, we used ResNet-18; iii) R4 considers the average of the accuracies shown in Fig. 2 (a-top left), but only the last point ($\approx 40\%$) should be considered as our results are expressed as final average accuracy.

R3 ***[Novelty]*** Although we acknowledge that DER and FDR are similar in their objective, our work highlights how an apparently subtle difference – storing logits throughout the optimization trajectory – substantially changes the CL training regime. As appreciated by R1 and R2, our experiments show that this strategy dramatically improves over FDR (with a peak of $+64.11$ and $+7.76$ of accuracy on S-CIFAR10 and S-TinyImg, see Tab. 2) and delivers more remarkable properties (Sec. 5). We consider this finding novel and interesting for the community; we believe this will foster new research and advances regarding rehearsal methods and the use of trajectory information for distilling knowledge.

R3 ***[The need for task boundaries]*** While we agree about A-GEM (we will fix Tab. 1), FDR specifies that responses are stored at the end of the task, so it needs boundaries to store them. Our position on GEM comes from its need of examples labeled with task ids, as it demands one QP-constraint per task. Task ids assume the presence of boundaries (and *viceversa*), even though we agree that GEM does not take specific actions at boundaries. If R3 thinks it would be more correct, we will mark GEM as free from task boundaries and be more specific about its dependence on task ids.

R3 ***[Eq. 5]*** Hinton et al. provide a full derivation in Sec. 2.1 of [16] stating that, under mild assumptions, the derivative of D_{KL} between post-softmax outputs approaches the one of MSE between logits. We will make this passage clearer.

R3 ***[Sampling in DER++]*** It is not about the sampling strategy; DER++ handles sharp distribution changes by storing labels y, whose training signal is more reliable than the one provided by logits z stored at the beginning of a new task.

R3 ***[Training time]*** A-GEM's training time is comparable with ER's as we apply data augmentation on buffer examples for ER (and not for A-GEM, for which is detrimental, see Footnote 2). Otherwise, ER trains 1.7x faster than A-GEM.