A Additional Experiment Information

We extended the experiments to 5 permutations and tasks in the same manner as the main text. For this experiment, we evaluated the classifier after training had completed (at the end of task 5) and measured the accuracy for examples from each of task 1...5. Table A reports these accuracies for OGD and the baseline training methods. The results suggest that the overall performance of OGD is significantly better than EWC and SGD while being on par with A-GEM.

	Accuracy ± Std. (%)				
	Task 1	**Task 2**	**Task 3**	**Task 4**	**Task 5**
MTL	93.2 ± 1.3	91.5 ± 0.5	91.3 ± 0.7	91.3 ± 0.6	88.4 ± 0.8
OGD	79.5 ± 2.3	$\mathbf{88.9} \pm 0.7$	$\mathbf{89.6} \pm 0.3$	$\mathbf{91.8} \pm 0.9$	92.4 ± 1.1
A-GEM	$\mathbf{85.5} \pm 1.7$	87.0 ± 1.5	$\mathbf{89.6} \pm 1.1$	91.2 ± 0.8	$\mathbf{93.9} \pm 1.0$
EWC	64.5 ± 2.9	77.1 ± 2.3	80.4 ± 2.1	87.9 ± 1.3	93.0 ± 0.5
SGD	60.6 ± 4.3	77.6 ± 1.4	79.9 ± 2.1	87.7 ± 2.9	92.4 ± 1.1

Table 3: *Permuted MNIST*: The accuracy of models for test examples from the indicated class after being trained on all tasks in sequence, except the multi-task setup (MTL). The best continual learning results are highlighted in **bold**.

A.1 Increased Training Epochs

We study the effect that increasing the number of training epochs has on the performance of the different training methods on permuted MNIST. For the MNIST experiments in the Section 4, we train for 5 epochs per task, which is enough to achieve 93% accuracy on vanilla MNIST classification and is in the regime short enough to avoid over-fitting. In order to determine whether increased training time has an effect on the performance in the multi-task setting, we train a classifier on 2-task permuted MNIST running each task training for 20, 40, 80, and 120 epochs and report the classification accuracy on task 1 after task 2 has finished. The results are shown in Figure 5. Note that A-GEM and OGD have maintained competitive performance with increasing number of epochs while in the case of SGD and EWC the performance first increases and then drops.

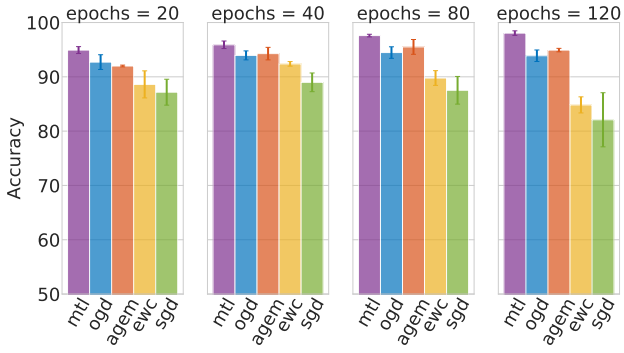

Figure 5: The performance of OGD versus others as a function of the number of training epochs for each task on permuted MNIST.

In the same way as the previous experiments, we extended the rotated MNIST experiment to more tasks by training a classifier on 5 rotated MNIST tasks with increasing angle of rotation. We defined the tasks as classification under angles of $T_1 = Rot(0°)$, $T_2 = Rot(10°)$, ..., $T_5 = Rot(40°)$, and train the models in that order. Table 4 shows the accuracy of the fully-trained model at classifying examples from each tasks. We can observe that OGD outperforms other methods on 10, 20, and 30 degree rotations.

A.2 Split MNIST

We present the results of the split MNIST study described in Section 4.4 on 3 other instances of the split MNIST task. These instances differ by the way the MNIST classes are split into tasks and the order in which the tasks

	Accuracy ± Std. (%)				
	Task 1	**Task 2**	**Task 3**	**Task 4**	**Task 5**
MTL	92.1 ± 0.9	94.3 ± 0.9	95.2 ± 0.9	93.4 ± 1.1	90.5 ± 1.5
OGD	$\mathbf{75.6} \pm 2.1$	$\mathbf{86.6} \pm 1.3$	$\mathbf{91.7} \pm 1.1$	94.3 ± 0.8	93.4 ± 1.1
A-GEM	72.6 ± 1.8	84.4 ± 1.6	91.0 ± 1.1	93.9 ± 0.6	$\mathbf{94.6} \pm 1.1$
EWC	61.9 ± 2.0	78.1 ± 1.8	89.0 ± 1.6	94.4 ± 0.7	93.9 ± 0.6
SGD	62.9 ± 1.0	76.5 ± 1.5	88.6 ± 1.4	$\mathbf{95.1} \pm 0.5$	94.1 ± 1.1

Table 4: *Rotated MNIST*: The accuracy of models for test examples from the indicated class after being trained on all tasks in sequence, except the multi-task setup (MTL). The best continual learning results are highlighted in **bold**.

are presented. Tables 5, 6, and 7 show the results on these tests. We can see that the ordering of the task methods is preserved in all tests: MTL and OGD are very close in performance, with a gap before A-GEM, and finally EWC and SGD.

	Accuracy ± Std. (%)				
	Task 1	Task 2	Task 3	Task 4	Task 5
mtl	99.6 ± 0.2	98.5 ± 0.4	97.7 ± 0.4	96.8 ± 1.0	98.7 ± 0.3
ogd	99.6 ± 0.4	97.7 ± 0.1	97.3 ± 0.5	98.0 ± 0.9	99.3 ± 0.1
agem	99.2 ± 0.6	91.4 ± 3.7	91.4 ± 0.9	87.1 ± 3.9	98.9 ± 0.3
ewc	97.0 ± 3.2	92.7 ± 3.8	91.9 ± 5.7	94.3 ± 2.2	99.2 ± 0.6
sgd	97.4 ± 2.4	92.2 ± 3.5	89.2 ± 8.6	94.5 ± 1.4	99.1 ± 0.3

Table 5: The accuracy of models trained by different methods on split MNIST. The reported values are the accuracy of the model for test examples from the indicated class after the model has been trained on all tasks in sequence. This table contains the same settings as Table 2, but with a different order of MNIST classes assigned to the tasks.

	Accuracy ± Std. (%)				
	Task 1	Task 2	Task 3	Task 4	Task 5
mtl	99.4 ± 0.2	99.2 ± 0.3	98.6 ± 0.4	99.7 ± 0.3	98.6 ± 0.5
ogd	99.0 ± 0.4	98.6 ± 0.1	98.0 ± 0.2	99.6 ± 0.3	99.6 ± 0.2
agem	94.1 ± 2.9	93.8 ± 5.5	90.6 ± 2.2	99.4 ± 0.3	99.4 ± 0.3
ewc	94.8 ± 2.9	95.3 ± 3.1	95.5 ± 0.6	99.3 ± 0.2	99.3 ± 0.2
sgd	94.6 ± 2.1	96.3 ± 1.2	95.0 ± 1.6	99.3 ± 0.4	99.3 ± 0.2

Table 6: The accuracy of models trained by different methods on split MNIST. The reported values are the accuracy of the model for test examples from the indicated class after the model has been trained on all tasks in sequence. This table contains the same settings as Table 2, but with a different order of MNIST classes assigned to the tasks.

	Accuracy ± Std. (%)				
	Task 1	Task 2	Task 3	Task 4	Task 5
mtl	98.4 ± 0.2	100.0 ± 0.0	98.6 ± 0.3	99.5 ± 0.2	98.9 ± 0.5
ogd	98.1 ± 0.8	99.9 ± 0.1	97.8 ± 0.6	99.4 ± 0.3	99.5 ± 0.3
agem	92.1 ± 2.7	93.8 ± 8.2	93.0 ± 3.5	98.6 ± 0.5	99.5 ± 0.3
ewc	92.5 ± 2.2	98.1 ± 3.0	94.0 ± 0.9	99.4 ± 0.2	99.5 ± 0.3
sgd	89.6 ± 4.4	98.9 ± 1.0	89.1 ± 7.9	98.9 ± 0.7	99.5 ± 0.3

Table 7: The accuracy of models trained by different methods on split MNIST. The reported values are the accuracy of the model for test examples from the indicated class after the model has been trained on all tasks in sequence. This table contains the same settings as Table 2, but with a different order of MNIST classes assigned to the tasks.