Supplementary Material for : Initial Classifier Weights Replay for Memoryless Class Incremental Learning

Eden Belouadah[1,2]
eden.belouadah@cea.fr
Adrian Popescu[1]
adrian.popescu@cea.fr
Ioannis Kanellos[2]
ioannis.kanellos@imt-atlantique.fr

[1] CEA, LIST,
F-91191, Gif-sur-Yvette, France
[2] IMT Atlantique,
Computer Science Department,
F-29238, Brest, France

1 Introduction

In this supplementary material we provide:

- details about the evaluation datasets,
- implementation details for the tested methods,
- results with other normalization approaches,
- Error analysis for FT, $inFT^{mc}_{siw}$ and $LUCIR$.

2 Dataset details

2.1 Datasets

Four datasets that were designed for object, face, and landmark recognition are used here. The choice of significantly different tasks is essential to study the adaptability and robustness of the tested methods. The main dataset statistics are provided in Table 1.

- *ILSVRC* [11] is a subset of 1000 *ImageNet* classes used in the *ImagenetLSVRC* challenges. It is constituted of leaves of the *ImageNet* hierarchy which most often depict specific visual concepts.
- *VGGFace2* [2] is designed for face recognition. We selected 1000 classes having the largest number of associated images. Face cropping is done with MTCNN [12] before further processing.
- *Google Landmarks* [7] (*Landmarks* below) is built for landmark recognition, and we selected 1000 classes having the largest number of associated images.
- *CIFAR100* [5] is designed for object recognition and includes 100 basic level classes [10].

Dataset	train	test	μ(train)	σ(train)
ILSVRC	1,231,167	50,000	1231.16	70.18
VGGFace2	491,746	50,000	491.74	49.37
Landmarks	374,367	20,000	374.36	103.82
CIFAR100	50,000	10,000	500.00	0.00

Table 1: Main statistics for the evaluation datasets, μ is the mean number of images per class; and σ is the standard deviation of the distribution of the number of images per class.

3 Implementation details

A ResNet-18 architecture [3] with an SGD optimizer is used as a backbone for all the methods. *LUCIR* [4] is run using the optimal parameters of the public implementation provided in the original paper. *LwF* [6] is run using the code from [2].

FT and its derivatives are based on the same fine-tuning backbone and are implemented in Pytorch [8]. Training images are processed using randomly resized 224×224 crops, horizontal flipping, and are normalized afterward. Given the difference in scale and the number of images between CIFAR100 and the other datasets, we found that a different parametrization was needed for this dataset. Note that the parameters' values presented below are largely inspired by the original ones given in [3].

For CIFAR100, the first non-incremental state and *Full* are run for 300 epochs with $batch\ size = 128$, $momentum = 0.9$ and $weight\ decay = 0.0005$. The lr is set to 0.1 and is divided by 10 when the error plateaus for 60 consecutive epochs. The incremental states of *FT* are trained for 70 epochs with $batch\ size = 128$, $momentum = 0.9$ and $weight\ decay = 0.0005$. The learning rate is set to $lr = 0.1/t$ at the beginning of each incremental state $\mathcal{Z}_t$ and is divided by 10 when the error plateaus for 15 consecutive epochs.

For ILSVRC, VGGFace2 and Landmarks, the first non-incremental state and *Full* are run for 120 epochs with $batch\ size = 256$, $momentum = 0.9$ and $weight\ decay = 0.0001$. The lr is set to 0.1 and is divided by 10 when the error plateaus for 10 consecutive epochs. The incremental states of *FT* are trained for 35 epochs with $batch\ size = 256$, $momentum = 0.9$ and $weight\ decay = 0.0001$. The learning rate is set to $lr = 0.1/t$ at the beginning of each incremental state $\mathcal{Z}_t$ and is divided by 10 when the error plateaus for 5 consecutive epochs.

4 Results with other calibration methods

Table 2 provides results with mean and min-max normalization of weights in addition to *L2* and *siw*. These two supplementary normalization techniques are defined below.

- *min-max normalization* - each dimension of the classifier is calculated using:

$$s_k = \frac{w_k - min(W)}{max(W) - min(W)} \tag{1}$$

- *mean normalization* - each dimension of the classifier is calculated using

$$s_k = \frac{w_k - \mu(W)}{max(W) - min(W)} \tag{2}$$

Dataset	ILSVRC			VGGFace2			Landmarks			CIFAR100			G_{IL}
States	T=10	T=20	T=50	T=10	T=20	T=50	T=10	T=20	T=50	T=10	T=20	T=50	
$inFT_{min-max}$	3.3	10.0	7.1	4.7	20.1	18.5	17.2	12.2	6.3	19.9	18.3	20.7	-55.52
$inFT_{mean}$	54.1	49.4	38.0	69.7	78.4	58.6	72.8	61.1	41.3	52.9	38.1	21.0	-23.76
$inFT_{L2}$	51.6	43.3	34.5	76.8	66.8	55.1	61.4	52.5	39.2	47.5	39.3	22.5	-26.80
$inFT_{siw}$	**61.6**	**51.9**	**39.9**	**84.0**	**80.6**	**61.9**	**75.1**	**62.6**	**43.2**	**56.0**	**41.8**	**22.5**	**-20.97**
$Full$	92.3			99.2			99.1			91.2			-

Table 2: Top-5 average IL accuracy (%) for the min-max and mean normalization tested in addition to L2 and standardization, with T={10, 20, 50} incremental states. Best results are in bold.

Standardization provides the best performance for all tested configurations. Mean calibration is second best and has better performance compared to the L2-normalization already used in [1]. Calibration with min-max is not effective and did not provide any good results.

5 Error analysis

Following [1], in Table 3, we provide top-1 correct and wrong classifications for: (1) FT - the simplest method tested, (2) $LUCIR$ - the best existing method (3) $inFT^{mc}_{siw}$ - the proposed method. The analysis is done for the large dataset ILSVRC, with $T = 20$ states. $c(p)$ and $c(n)$ are the correct classification for past/new classes. $e(p,p)$ and $e(p,n)$ are erroneous classifications for test samples of past classes mistaken for other past classes and new classes respectively. $e(n,p)$ and $e(n,n)$ are erroneous classifications for test samples of new classes mistaken for past classes and other new classes respectively. Note that the percentages on the first three and last three lines of each table sum up to 100%. Since the number of test images varies across IL states, percentages are calculated separately for test images of past and new classes in each $\mathcal{Z}_t$ to get a quick view of the relative importance of each type of errors. $c(p)$, $e(p,p)$, and $e(p,n)$ sum to 100% on each column, as do $c(n)$, $e(n,n)$, and $e(n,p)$. The analysis shows that vanilla FT suffers from a total forgetting of the past classes since all their test images are wrongly classified. The effect of catastrophic forgetting is obvious in the way that 100% of past classes are mistakenly classified as belonging to new classes. Equally important, standardization of the initial weights not only reduces forgetting, but also reduces considerably the confusions among new classes. The comparison of $LUCIR$ and $inFT^{mc}_{siw}$ shows that the first method is better at classifying test samples of new classes but has worse behavior for test samples of past classes. $LUCIR$ $c(p)$ scores are better for the first three iterations but fall behind those of $inFT^{mc}_{siw}$ afterwards. Note that both methods are strongly affected by catastrophic forgetting toward the end of the incremental process, with top-1 accuracy at 6% and 11.8% for $LUCIR$ and $inFT^{mc}_{siw}$ respectively. This finding indicates that, while both distillation in $LUCIR$ and classifier weights replay $inFT^{mc}_{siw}$ have a slight positive effect, memoryless IL remains a very challenging task. It is also interesting that the distribution of errors is different. $LUCIR$ fails to ensure fairness between past and new classes since $e(p,n)$ are much more frequent than $e(p,p)$. $inFT^{mc}_{siw}$ is less biased toward new classes but produces a large number of confusions between past classes ($e(p,p)$.

	Incremental states	$\mathcal{Z}_1$	$\mathcal{Z}_2$	$\mathcal{Z}_3$	$\mathcal{Z}_4$	$\mathcal{Z}_5$	$\mathcal{Z}_6$	$\mathcal{Z}_7$	$\mathcal{Z}_8$	$\mathcal{Z}_9$	$\mathcal{Z}_{10}$	$\mathcal{Z}_{11}$	$\mathcal{Z}_{12}$	$\mathcal{Z}_{13}$	$\mathcal{Z}_{14}$	$\mathcal{Z}_{15}$	$\mathcal{Z}_{16}$	$\mathcal{Z}_{17}$	$\mathcal{Z}_{18}$	$\mathcal{Z}_{19}$
FT	$c(p)$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	$e(p,p)$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	$e(p,n)$	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	$c(n)$	87.8	87.28	90.48	91.4	90.44	87.92	89.64	88.12	87.24	89.68	89.72	90.16	90.6	89.8	87.84	92.4	89.56	89.28	87.52
	$e(n,n)$	12.2	12.72	9.52	8.6	9.56	12.08	10.36	11.88	12.76	10.32	10.28	9.84	9.4	10.2	12.16	7.6	10.44	10.72	12.48
	$e(n,p)$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$inFT^{mc}_{siw}$	$c(p)$	38.4	27.0	33.2	31.3	29.0	22.0	20.1	15.0	17.9	14.7	17.7	16.5	15.3	13.1	13.2	14.0	14.1	12.5	11.8
	$e(p,p)$	22.7	15.0	41.4	41.9	60.7	48.5	51.8	31.9	60.2	40.7	68.1	62.6	66.8	48.2	47.2	66.9	64.9	52.7	50.0
	$e(p,n)$	38.9	58.0	25.4	26.8	10.3	29.5	28.1	53.0	21.9	44.6	14.1	20.9	17.9	38.7	39.6	19.1	21.0	34.8	38.2
	$c(n)$	75.8	82.7	75.7	75.8	67.2	75.1	77.4	83.8	69.8	83.2	68.6	76.1	70.5	82.0	78.4	76.2	72.6	80.4	80.3
	$e(n,n)$	8.5	11.5	4.2	3.1	1.8	6.8	4.6	9.8	4.5	7.9	3.2	3.5	3.1	6.5	8.3	2.7	3.8	5.4	7.7
	$e(n,p)$	15.7	5.8	20.0	21.1	31.0	18.0	18.0	6.4	25.7	8.9	28.2	20.4	26.4	11.5	13.3	21.1	23.6	14.1	12.0
LUCIR	$c(p)$	66.1	46.9	33.5	26.7	23.2	19.0	15.1	13.3	11.8	9.9	9.1	8.3	7.9	8.0	7.5	6.7	6.5	6.0	6.0
	$e(p,p)$	4.2	10.1	14.7	20.4	26.6	25.8	24.1	27.5	27.9	28.1	28.3	28.8	29.8	31.5	29.3	30.8	29.6	29.6	30.6
	$e(p,n)$	29.8	42.9	51.8	52.9	50.2	55.3	60.8	59.2	60.3	62.0	62.6	63.0	62.3	60.5	63.2	62.5	63.9	64.4	63.4
	$c(n)$	78.3	79.7	82.2	82.2	82.4	78.2	82.6	81.5	79.0	84.5	82.7	83.4	84.1	82.9	81.2	86.2	82.8	83.3	81.2
	$e(n,n)$	16.0	15.5	13.5	11.4	12.2	15.2	12.3	13.0	14.5	11.4	11.9	11.9	11.2	11.5	14.2	9.0	11.7	12.1	13.8
	$e(n,p)$	5.6	4.8	4.4	6.4	5.3	6.6	5.2	5.5	6.5	4.1	5.4	4.8	4.6	5.6	4.6	4.8	5.5	4.6	5.0

Table 3: Top-1 correct and wrong classification for FT, $inFT^{mc}_{siw}$ and *LUCIR* for ILSVRC with $T = 20$.

References

[1] Eden Belouadah and Adrian Popescu. Scail: Classifier weights scaling for class incremental learning. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[2] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *13th IEEE International Conference on Automatic Face & Gesture Recognition, FG 2018, Xi'an, China, May 15-19, 2018*, pages 67–74, 2018.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition*, CVPR, 2016.

[4] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 831–839, 2019.

[5] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[6] Zhizhong Li and Derek Hoiem. Learning without forgetting. In *European Conference on Computer Vision*, ECCV, 2016.

[7] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *ICCV*, pages 3476–3485. IEEE Computer Society, 2017.

[8] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *Advances in Neural Information Processing Systems Workshops*, NIPS-W, 2017.

[9] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *Conference on Computer Vision and Pattern Recognition*, CVPR, 2017.

[10] Eleanor Rosch, Carolyn B Mervis, Wayne D Gray, David M Johnson, and Penny Boyes-Braem. Basic objects in natural categories. *Cognitive Psychology*, 8(3):382 – 439, 1976.

[11] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[12] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.*, 23(10):1499–1503, 2016.