Supplementary Material for "Dataset Knowledge Transfer for Class-Incremental Learning without Memory"

Habib Slim[1*] Eden Belouadah[1,2*] Adrian Popescu[1] Darian Onchis[3]
[1]Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
[2] IMT Atlantique, Lab-STICC, team RAMBO, UMR CNRS 6285, F-29328, Brest, France
[3] West University of Timisoara, Timisoara, Romania
habib.slim@grenoble-inp.org, {eden.belouadah, adrian.popescu}@cea.fr, darian.onchis@e-uvt.ro

Introduction

In this supplementary material, we provide:

- implementation details of *adBiC* and the tested backbone IL methods (Section A).
- classes lists of target datasets used for evaluation (Section B).
- additional figures highlighting the effects of *adBiC* on the tested backbone methods (Section C).
- additional tables for the robustness experiment presented in Section 4.4 of the paper (Section D).
- results on PLACES-100 dataset (Table 2).
- additional accuracy plots for all methods and datasets (Figures 2 and 3).

A. Implementation details

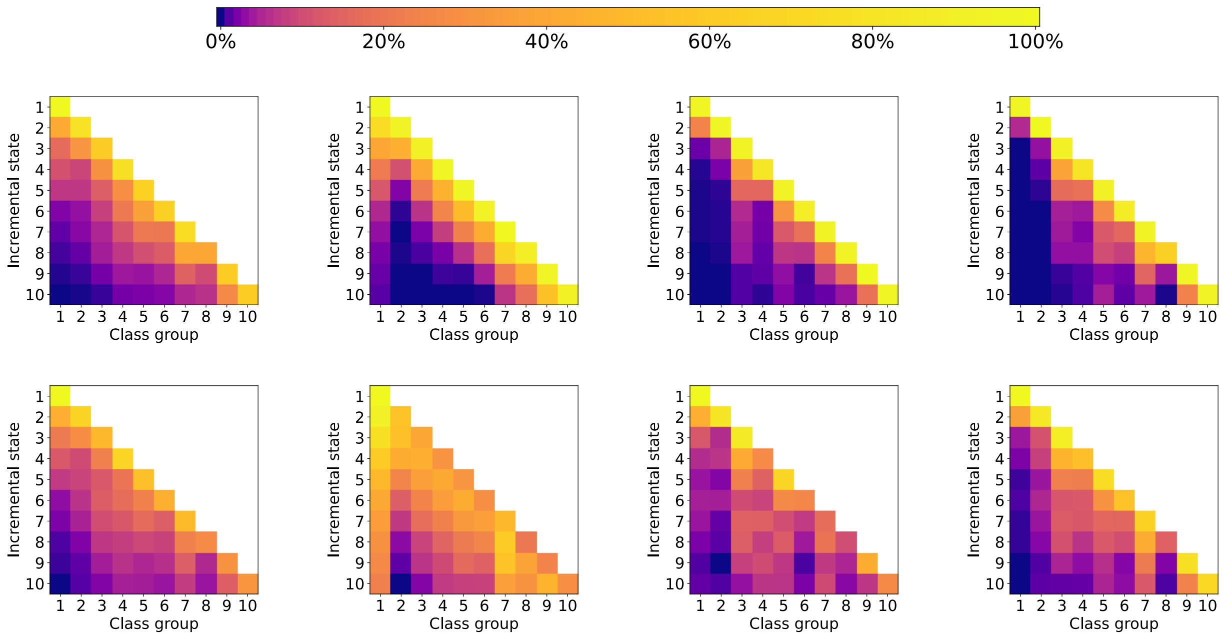
A.1. Backbone IL methods

For *LUCIR* [5] and *SIW* [2], we used the original codes provided by the authors. For *LwF*, we adapted the multi-class Tensorflow [1] implementation from [10] to IL without memory. For *FT+*, we implemented the method by replacing classification weights of each class group by their initial weights learned when classes were encountered for the first time. All methods use a ResNet-18 [4] backbone, with batch size 128. For *LwF*, we use a base learning rate of 1.0 divided by 5 after 20, 30, 40 and 50 epochs. The weight decay is set to 10^{-5} and models are trained for 70 epochs in each state. For *LUCIR*, we mostly use the parameters recommended for CIFAR-100 in the original paper [5]. We set λ_{base} to 5. For each state, we train models for 160 epochs. The base learning rate is set to 0.1 and divided by 10 after 80 and 120 epochs. The weight decay is set to $5 \cdot 10^{-4}$ and the momentum to 0.9. Note that since no memory of past classes is available, the margin ranking loss is unusable and thus removed. *SIW* and *FT+* are both trained with the same set of hyperparameters. Following [2], models are trained from scratch for 300 epochs in the first non-incremental state, using the SGD optimizer with momentum 0.9. The base learning rate is set to 0.1, and is divided by 10 when the loss plateaus for 60 epochs. The weight decay is set to $5 \cdot 10^{-4}$. For incremental states, the same hyperparameters are used, except for the number of epochs which is reduced to 70 and the lr is divided by 10 when the loss plateaus for 15 epochs.

A.2. Adaptive bias correction

The correction of output scores is done in the same way for all methods. After the extraction of scores and labels for each model, batches are fed into a PyTorch [9] module which performs the optimization of *adBiC* parameters, or the transfer of previously learned parameters. Following [8], *BiC* and *adBiC* layers are implemented as pairs of parameters and optimized simply through backpropagation. Parameters α_s^k, β_s^k corresponding to each incremental state s are optimized for 300 epochs, with the Adam [6] optimizer and a starting learning rate of 10^{-3}. An L2-penalty is added to the loss given in Equation 3 of the main paper, with a lambda of $5 \cdot 10^{-3}$ for α parameters and $5 \cdot 10^{-2}$ for β parameters.

*Equal contribution



(a) LwF [7] (b) LUCIR [5] (c) SIW [2] (d) FT+ [8]

Figure 1: Accuracies per incremental state for each class group, for models trained with *LwF* and *LUCIR* on CIFAR-100 for $S = 10$ states, before (*top*) and after (*bottom*) *adBiC* correction. Each row represents an incremental state and each square the accuracy on a group of classes first learned in a specific state. In the first state, represented by the first rows of the matrices, models are only evaluated on the first class group. In the second state, represented by the second rows, models are evaluated on the first two class groups, etc. *Best viewed in color*.

B. Datasets description

We provide in Table 1 the lists of classes contained in each of the target datasets we used for evaluation. Overall, IMN-100, the randomly sampled set of 100 leaf classes from Imagenet [3], is more diversified than CIFAR-100, which mostly contains animal classes. IMN-100 is visually varied between different types of objects, foods, animals, vehicles, clothes and events. CIFAR-100 contains, in addition to animals, some types of objects and vehicles. The FOOD-100 dataset and BIRDS-100 (extracted from ImageNet [3]) are more specialized than IMN-100 and CIFAR-100 and are thus useful to test the robustness of our method on finer-grained datasets. PLACES-100 and FOOD-100 are target datasets which have a larger domain shift with ImageNet classes, and are thus useful to test the robustness of our method against domain variation. Similarly to IMN-100, reference datasets are random subsets of ImageNet leaves. They contain various object types and are useful for knowledge transfer.

C. Effects of adaptive bias correction

In Figure 1, we illustrate the effects of *adBiC* on state-wise accuracies, for all backbone IL methods evaluated in this work. Before adaptive correction (*top*), all methods provide strong performance on the last group of classes learned (represented by the diagonals). Their performance is generally poorer for past classes (under the diagonals). After correction (*bottom*), all methods perform better on past class groups (with a trade-off in accuracy on the last class group) resulting in a higher overall performance.

D. Effect of the number of reference datasets

In Table 3, we provide a full set of results for the accuracies of *adBiC* with *LwF*, *LUCIR*, *SIW* and *FT+* when varying the number R of reference datasets, following Table 3 of the main paper. For all methods considered, a single reference dataset is sufficient to obtain significant gains with *adBiC*.

	Classes names
CIFAR-100	Apple, Aquarium fish, Baby, Bear, Beaver, Bed, Bee, Beetle, Bicycle, Bottle, Bowl, Boy, Bridge, Bus, Butterfly, Camel, Can, Castle, Caterpillar, Cattle, Chair, Chimpanzee, Clock, Cloud, Cockroach, Couch, Crab, Crocodile, Cup, Dinosaur, Dolphin, Elephant, Flatfish, Forest, Fox, Girl, Hamster, House, Kangaroo, Keyboard, Lamp, Lawn mower, Leopard, Lion, Lizard, Lobster, Man, Maple tree, Motorcycle, Mountain, Mouse, Mushroom, Oak tree, Orange, Orchid, Otter, Palm tree, Pear, Pickup truck, Pine tree, Plain, Plate, Poppy, Porcupine, Possum, Rabbit, Raccoon, Ray, Road, Rocket, Rose, Sea, Seal, Shark, Shrew, Skunk, Skyscraper, Snail, Snake, Spider, Squirrel, Streetcar, Sunflower, Sweet pepper, Table, Tank, Telephone, Television, Tiger, Tractor, Train, Trout, Tulip, Turtle, Wardrobe, Whale, Willow tree, Wolf, Woman, Worm
IMN-100	Bletilla striata, Christmas stocking, Cognac, European sandpiper, European turkey oak, Friesian, Japanese deer, Luger, Sitka spruce, Tennessee walker, Torrey pine, Baguet, Bald eagle, Barn owl, Bass guitar, Bathrobe, Batting helmet, Bee eater, Blue gum, Blue whale, Bones, Borage, Brass, Caftan, Candytuft, Carthorse, Cattle egret, Cayenne, Chairlift, Chicory, Cliff dwelling, Cocktail dress, Commuter, Concert grand, Crazy quilt, Delivery truck, Detached house, Dispensary, Drawing room, Dress hat, Drone, Frigate bird, Frozen custard, Gemsbok, Giant kangaroo, Guava, Hamburger bun, Hawfinch, Hill myna, Howler monkey, Huisache, Jennet, Jodhpurs, Ladder truck, Loaner, Micrometer, Mink, Moorhen, Moorhen, Moped, Mortarboard, Mosquito net, Mountain zebra, Muffler, Musk ox, Obelisk, Opera, Ostrich, Ox, Oximeter, Playpen, Post oak, Purple-fringed orchid, Purple willow, Quaking aspen, Ragged robin, Raven, Redpoll, Repository, Roll-on, Scatter rug, Shopping cart, Shower curtain, Slip-on, Spider orchid, Sports car, Steam iron, Stole, Stuffed mushroom, Subcompact, Sundial, Tabby, Tabi, Tank car, Tramway, Unicycle, Wagtail, Walker, Window frame, Wood anemone
BIRDS-100	American bittern, American coot, Atlantic puffin, Baltimore oriole, Barrow's goldeneye, Bewick's swan, Bullock's oriole, California quail, Eurasian kingfisher, European gallinule, European sandpiper, Orpington, Amazon, Barn owl, Black-crowned night heron, Black-necked stilt, Black-winged stilt, Black swan, Black vulture, Black vulture, Blue peafowl, Brambling, Bufflehead, Buzzard, Cassowary, Cockerel, Common spoonbill, Crossbill, Duckling, Eastern kingbird, Emperor penguin, Fairy bluebird, Fishing eagle, Fulmar, Gamecock, Golden pheasant, Goosander, Goshawk, Great crested grebe, Great horned owl, Great white heron, Greater yellowlegs, Greenshank, Gyrfalcon, Hawfinch, Hedge sparrow, Hen, Honey buzzard, Hornbill, Kestrel, Kookaburra, Lapwing, Least sandpiper, Little blue heron, Little egret, Macaw, Mandarin duck, Marsh hawk, Moorhen, Mourning dove, Muscovy duck, Mute swan, Ostrich, Owlet, Oystercatcher, Pochard, Raven, Red-legged partridge, Red-winged blackbird, Robin, Robin, Rock hopper, Roseate spoonbill, Ruby-crowned kinglet, Ruffed grouse, Sanderling, Screech owl, Screech owl, Sedge warbler, Shoveler, Siskin, Snow goose, Snowy egret, Song thrush, Spotted flycatcher, Spotted owl, Sulphur-crested cockatoo, Thrush nightingale, Tropic bird, Tufted puffin, Turkey cock, Weka, Whistling swan, White-breasted nuthatch, White-crowned sparrow, White-throated sparrow, White stork, Whole snipe, Wood ibis, Wood pigeon.
FOOD-100	Apple pie, Baby back ribs, Baklava, Beef carpaccio, Beef tartare, Beet salad, Beignets, Bibimbap, Bread pudding, Breakfast burrito, Bruschetta, Caesar salad, Cannoli, Caprese salad, Carrot cake, Ceviche, Cheese plate, Cheesecake, Chicken curry, Chicken quesadilla, Chicken wings, Chocolate cake, Chocolate mousse, Churros, Clam chowder, Club sandwich, Crab cakes, Creme brulee, Croque madame, Cup cakes, Deviled eggs, Donuts, Dumplings, Edamame, Eggs benedict, Escargots, Falafel, Filet mignon, Fish and chips, Foie gras, French fries, French onion soup, French toast, Fried calamari, Fried rice, Frozen yogurt, Garlic bread, Gnocchi, Greek salad, Grilled cheese sandwich, Grilled salmon, Guacamole, Gyoza, Hamburger, Hot and sour soup, Hot dog, Huevos rancheros, Hummus, Ice cream, Lasagna, Lobster bisque, Lobster roll sandwich, Macaroni and cheese, Macarons, Miso soup, Mussels, Nachos, Omelette, Onion rings, Oysters, Pad thai, Paella, Pancakes, Panna cotta, Peking duck, Pho, Pizza, Pork chop, Poutine, Prime rib, Pulled pork sandwich, Ramen, Ravioli, Red velvet cake, Risotto, Samosa, Sashimi, Scallops, Seaweed salad, Shrimp and grits, Spaghetti bolognese, Spaghetti carbonara, Spring rolls, Steak, Strawberry shortcake, Sushi, Tacos, Takoyaki, Tiramisu, Tuna tartare

	Classes names
PLACES-100	Airplane cabin, Amphitheater, Amusement arcade, Aqueduct, Arcade, Archaelogical excavation, Archive, Arena performance, Attic, Bamboo forest, Bar, Barn, Baseball field, Bazaar outdoor, Beach, Beach house, Beauty salon, Bedroom, Bookstore, Bus interior, Cafeteria, Castle, Chemistry lab, Church outdoor, Cliff, Corridor, Courthouse, Crevasse, Department store, Desert sand, Desert vegetation, Dining room, Dorm room, Drugstore, Elevator lobby, Elevator shaft, Entrance hall, Escalator indoor, Farm, Field cultivated, Field wild, Florist shop indoor, Food court, Fountain, Garage indoor, Gazebo exterior, Golf course, Hangar outdoor, Harbor, Hardware store, Hayfield, Heliport, Highway, Home theater, Hospital room, Hot spring, Hotel outdoor, Hunting lodge outdoor, Ice skating rink indoor, Junkyard, Kasbah, Kitchen, Lagoon, Lake natural, Marsh, Mosque outdoor, Oast house, Office cubicles, Pagoda, Park, Pavilion, Physics laboratory, Pier, Porch, Racecourse, Residential neighborhood, Restaurant, Rice paddy, Rock arch, Ruin, Sauna, Server room, Shed, Shopfront, Storage room, Sushi bar, Television room, Television studio, Throne room, Topiary garden, Tower, Tree house, Trench, Underwater ocean deep, Waiting room, Water park, Waterfall, Wet bar, Windmill, Zen garden

Table 1: Classes names of target datasets listed by alphabetical order

Method	PLACES-100			PLACES-100 (*halved*)		
	$S=5$	$S=10$	$S=20$	$S=5$	$S=10$	$S=20$
LwF [7]	43.3	35.1	25.9	35.4	27.7	21.5
w/ adBiC	**44.2** **+ 0.9**	**36.6** **+ 1.5**	**28.6** **+ 2.7**	35.9 **+ 0.5**	28.5 **+ 0.8**	**23.6** **+ 2.1**
LUCIR [5]	40.5	26.0	16.0	35.5	23.2	14.7
w/ adBiC	42.8 **+ 2.3**	35.4 **+ 9.4**	23.3 **+ 7.3**	**40.5** **+ 5.0**	**33.6** **+ 10.4**	22.3 **+ 7.6**
SIW [2]	27.3	20.6	14.0	27.2	19.6	14.8
w/ adBiC	28.8 **+ 1.5**	21.2 **+ 0.6**	13.1 **- 0.9**	28.5 **+ 1.3**	19.3 **- 0.3**	14.3 **- 0.5**
FT+ [8]	26.9	20.8	12.1	00.0	00.0	00.0
w/ adBiC	27.3 **+ 0.4**	19.7 **- 1.1**	13.2 **+ 1.1**	25.6 **- 0.5**	17.2 **- 2.7**	13.5 **+ 1.1**

Table 2: Average top-1 incremental accuracy using $S = \{5, 10, 20\}$ states, for the PLACES-100 dataset with all and half of the training data. The PLACES-100 dataset is extracted from Places365 [11]. Similarly to the other datasets presented in the main paper, we randomly select a hundred classes from the original dataset to construct a dataset with a hundred classes. Results obtained are comparable to those obtained on the other datasets, despite the domain shift from ImageNet. Gains obtained over the backbone method are given in green, and the best results for each setting in bold.

S = 5	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	53.0	54.3 ± 0.2	54.3 ± 0.2	54.3 ± 0.1	54.4 ± 0.1	54.3 ± 0.1	54.3 ± 0.1	54.3 ± 0.1	54.3 ± 0.1	54.3 ± 0.1	54.3
S = 10	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	44.0	46.2 ± 0.3	46.4 ± 0.2	46.4 ± 0.2	46.4 ± 0.2	46.4 ± 0.1	46.4 ± 0.1	46.5 ± 0.1	46.4 ± 0.1	46.4 ± 0.1	46.4
S = 20	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	29.1	31.8 ± 0.3	32.1 ± 0.1	32.1 ± 0.2	32.1 ± 0.1	32.2 ± 0.1	32.2 ± 0.1	32.3 ± 0.1	32.3 ± 0.1	32.3 ± 0.1	32.3

(a) LwF [7]

S = 5	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	50.1	54.7 ± 0.4	54.8 ± 0.3	54.8 ± 0.1	54.8 ± 0.1	54.8 ± 0.1	54.8 ± 0.1	54.8 ± 0.1	54.8 ± 0.1	54.8 ± 0.1	54.8
S = 10	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	33.7	42.0 ± 0.7	42.1 ± 0.3	42.2 ± 0.4	42.3 ± 0.3	42.2 ± 0.2	42.2 ± 0.2	42.2 ± 0.1	42.2 ± 0.1	42.2 ± 0.1	42.2
S = 20	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	19.5	27.5 ± 1.4	27.8 ± 0.7	27.8 ± 0.9	28.3 ± 0.4	28.5 ± 0.5	28.6 ± 0.6	28.5 ± 0.4	28.4 ± 0.3	28.4 ± 0.2	28.4

(b) LUCIR [5]

S = 5	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	29.9	31.6 ± 0.2	31.6 ± 0.2	31.6 ± 0.1	31.7 ± 0.2	31.7 ± 0.1	31.7 ± 0.1	31.7 ± 0.1	31.7 ± 0.1	31.7 ± 0.1	31.7
S = 10	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	22.7	23.8 ± 0.4	23.8 ± 0.2	23.9 ± 0.2	24.0 ± 0.2	23.9 ± 0.1	24.0 ± 0.1	24.1 ± 0.1	24.0 ± 0.1	24.1 ± 0.1	24.1
S = 20	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	14.8	15.7 ± 0.3	15.7 ± 0.2	15.7 ± 0.2	15.8 ± 0.1	15.8 ± 0.2	15.8 ± 0.1	15.8 ± 0.1	15.8 ± 0.1	15.8 ± 0.1	15.8

(c) SIW [2]

S = 5	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	28.9	31.9 ± 0.2	32.0 ± 0.1	32.0 ± 0.1	32.0 ± 0.1	32.0 ± 0.1	32.0 ± 0.1	31.9 ± 0.1	32.0 ± 0.1	32.0 ± 0.1	31.9
S = 10	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	22.6	23.2 ± 0.4	23.5 ± 0.2	23.5 ± 0.2	23.6 ± 0.1	23.5 ± 0.2	23.6 ± 0.1	23.6 ± 0.1	23.6 ± 0.1	23.6 ± 0.1	23.6
S = 20	Raw	$R = 1$	$R = 2$	$R = 3$	$R = 4$	$R = 5$	$R = 6$	$R = 7$	$R = 8$	$R = 9$	$R = 10$
	14.5	14.8 ± 0.2	15.0 ± 0.1	15.0 ± 0.2	15.1 ± 0.1	15.0 ± 0.1	15.1 ± 0.1	15.1 ± 0.1	15.0 ± 0.1	15.0 ± 0.1	15.0

(d) FT+ [8]

Table 3: Average top-1 incremental accuracy of *adBiC*-corrected models trained incrementally on CIFAR-100 with *LwF*, *LUCIR*, *SIW* and *FT+*, for $S = \{5, 10, 20\}$ states, while varying the number R of reference datasets. For $R \leq 9$, results are averaged across 10 random samplings of the reference datasets. *Raw* is the accuracy of each method without bias correction.

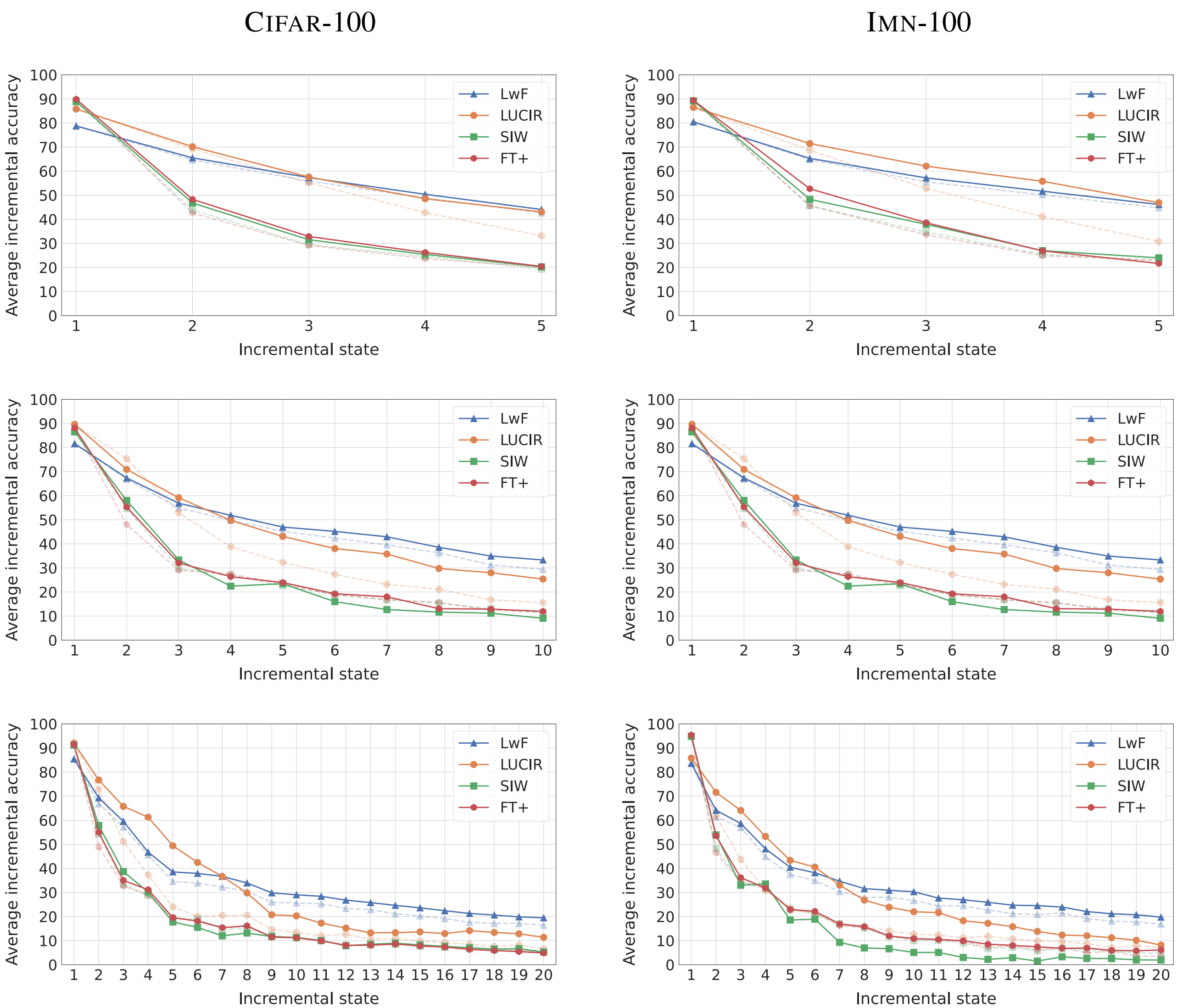


Figure 2: Average accuracies in each state on CIFAR-100 (*left*) and IMN-100 (*right*) datasets with all backbone methods after *adBiC* correction, for $S = 5$ (*top*), $S = 10$ (*middle*) and $S = 20$ (*bottom*) states. The accuracies without correction of the corresponding methods are provided in dotted lines (same colors). *Best viewed in color*.

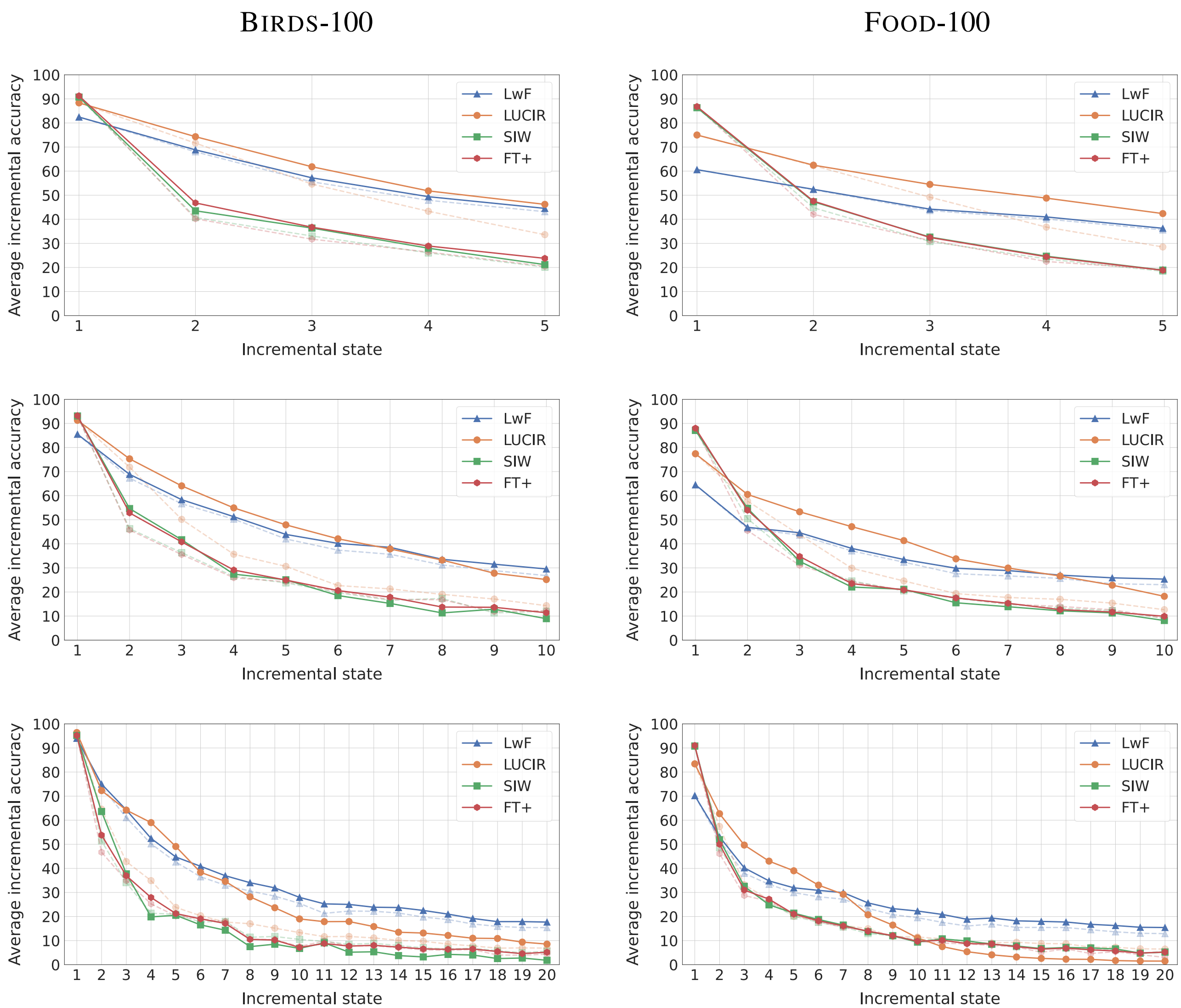


Figure 3: Average accuracies in each state on BIRDS-100 (left) and FOOD-100 (right) datasets with all backbone methods after *adBiC* correction, for $S = 5$ (*top*), $S = 10$ (*middle*) and $S = 20$ (*bottom*) states. The accuracies without correction of the corresponding methods are provided in dotted lines (same colors). *Best viewed in color*.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. Initial classifier weights replay for memoryless class incremental learning. In *British Machine Vision Conference (BMVC)*, 2020.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255, 2009.

[4] 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.

[5] 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.

[6] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

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

[8] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, and Joost van de Weijer. Class-incremental learning: survey and performance evaluation on image classification, 2021.

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[10] 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.

[11] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.