# Maintaining Discrimination and Fairness in Class Incremental Learning

**Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, Shu-Tao Xia**

CVPR 2020 **link**

20.10.20
발표: 정채연

# Contents

1. Overview
2. Related Works
3. Motivation
4. Methodology
5. Experiments & Results

# Overview

### Incremental learning

- Learning new classes gradually

Dataset D = $\{D^1, \cdots, D^B\}$ , where $D^b = \{(\mathbf{x}_1^b, y_1^b), \cdots, (\mathbf{x}_{n_b}^b, y_{n_b}^b)\}$

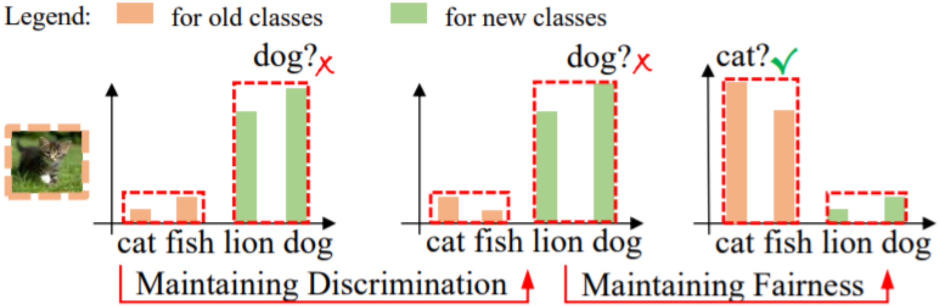
Learn new $D^b$ while maintaining information of old $\{D^1, \cdots, D^{b-1}\}$

### Problem

- Catastrophic forgetting in incremental learning

### Goal

- Maintaining discrimination and fairness in class incremental learning



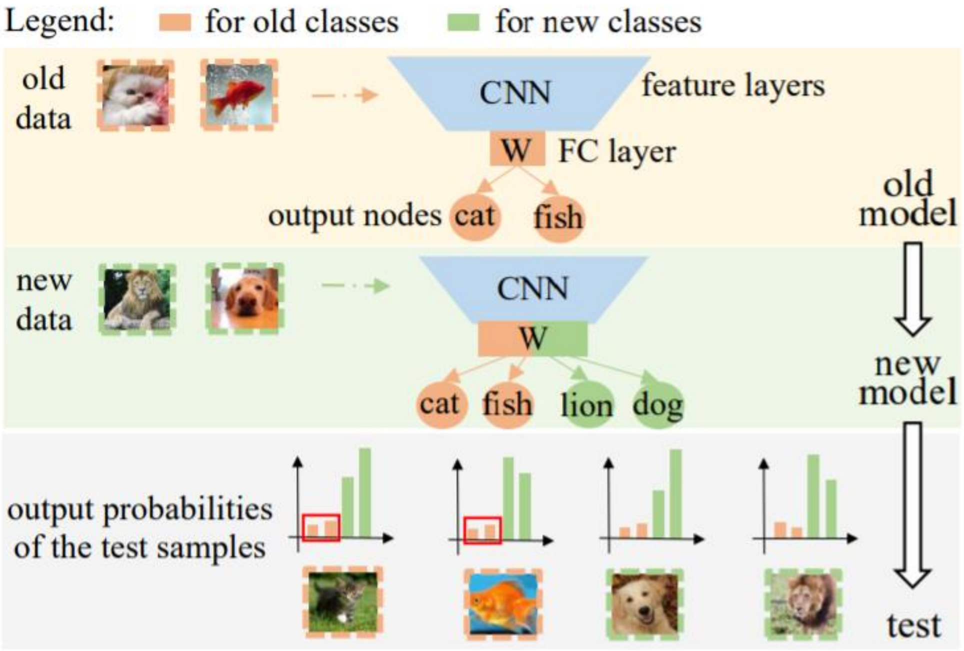


Figure 1: A vanilla method for class incremental learning.

# Related Works

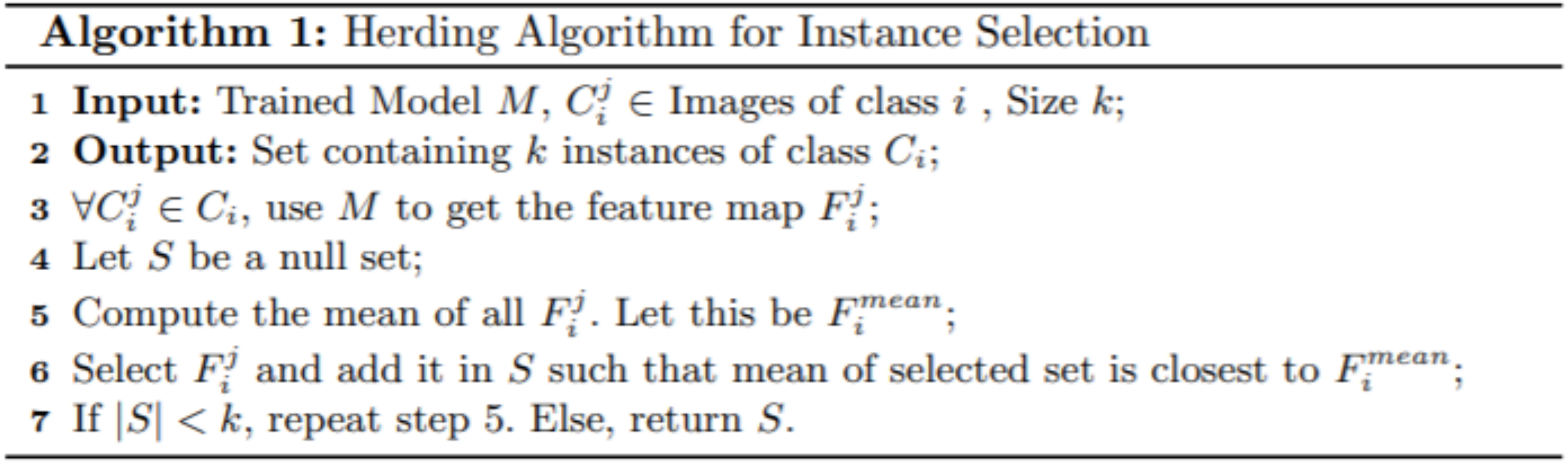
Previous works to alleviate catastrophic forgetting

1) Parameter control: constrain the important weights
    - EWC uses Fisher information matrix (link)
    - SI uses the path integral over the optimization trajectory (link)
    - MAS uses the gradients of the network output (link)
2) Knowledge distillation: transfer key knowledge from teacher to student model
    - Ensemble model to a single model (link)
    - LwF (link) LwF.MC (link) $M^2$KD (link)
3) Rehearsal: use some old data to make up training data
    - Using old data (link1, link2), Using GAN (link1, link2, link3)
4) Improving class imbalance between old data and new data
    - BiC (link), IL2M rectifies scores of old classes (link)

5) **Combination: Knowledge distillation + rehearsal**
    - iCaRL (link), EEIL (link)

# Motivation

1. **Knowledge distillation (KD) + Rehearsal**
   a. Initialize the model with the previous parameters
   b. learn new classes with the new data + a few old data (herding selection)
      → herding selection



**Algorithm 1:** Herding Algorithm for Instance Selection

```
1 Input: Trained Model $M$, $C_i^j \in$ Images of class $i$ , Size $k$;
2 Output: Set containing $k$ instances of class $C_i$;
3 $\forall C_i^j \in C_i$, use $M$ to get the feature map $F_i^j$;
4 Let $S$ be a null set;
5 Compute the mean of all $F_i^j$. Let this be $F_i^{mean}$;
6 Select $F_i^j$ and add it in $S$ such that mean of selected set is closest to $F_i^{mean}$;
7 If $|S| < k$, repeat step 5. Else, return $S$.
```

# Motivation

1. **Knowledge distillation (KD) + Rehearsal**
   a. Initialize the model with the previous parameters
   b. learn new classes with the new data + a few old data (herding selection)
   c. Loss

$$\mathcal{L}(\mathbf{x}, y) = (1-\lambda)\mathcal{L}_{CE}(\mathbf{x}, y) + \lambda\mathcal{L}_{KD}(\mathbf{x}) \qquad \text{( set } \lambda \text{ to } \frac{C^b_{old}}{C^b + C^b_{old}} \text{ )}$$

- CE loss

$$\mathcal{L}_{CE}(\mathbf{x}, y) = \sum_{c=1}^{C^b + C^b_{old}} -\delta_{c=y} \log\left(p_c(\mathbf{x})\right)$$

- KD loss

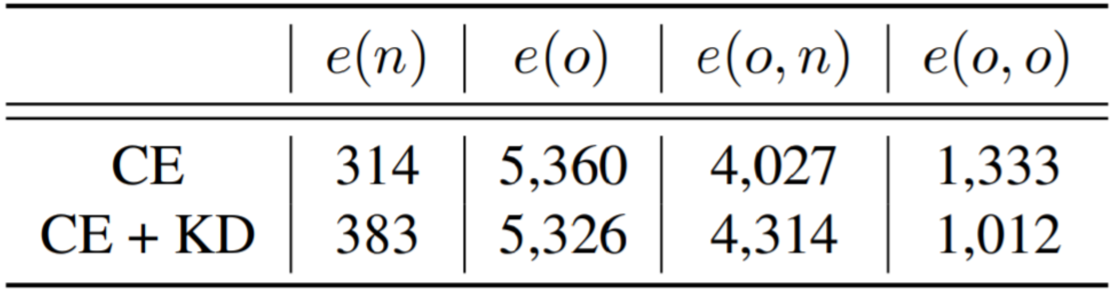
$$\mathcal{L}_{KD}(\mathbf{x}) = \sum_{c=1}^{C^b_{old}} -\hat{q}_c(\mathbf{x}) \log\left(q_c(\mathbf{x})\right),$$

$$\text{where } \hat{q}_c(\mathbf{x}) = \frac{e^{\hat{o}_c(\mathbf{x})/T}}{\sum_{j=1}^{C^b_{old}} e^{\hat{o}_j(\mathbf{x})/T}}, q_c(\mathbf{x}) = \frac{e^{o_c(\mathbf{x})/T}}{\sum_{j=1}^{C^b_{old}} e^{o_j(\mathbf{x})/T}};$$

→ Soft label

# Motivation

## 2. Effect of KD



|  | $e(n)$ | $e(o)$ | $e(o,n)$ | $e(o,o)$ |
|---|---|---|---|---|
| CE | 314 | 5,360 | 4,027 | 1,333 |
| CE + KD | 383 | 5,326 | 4,314 | 1,012 |

(CIFAR-100 with 5 incremental steps and 20 classes per step)

- kept the knowledge of old model
- but prediction bias towards new classes **not** alleviated

## 3. Limitation

- the cost of misclassifying old samples to new classes < that to other old class → bias towards new classes
  - old sample → correct old sample
  - old sample → wrong old sample
  - old sample → wrong new sample

$$\mathcal{L}_{KD}(\mathbf{x}) = \sum_{c=1}^{C_{old}^{b}} -\hat{q}_c(\mathbf{x}) \log\left(q_c(\mathbf{x})\right), \qquad (3)$$

$$\text{where } \hat{q}_c(\mathbf{x}) = \frac{e^{\hat{o}_c(\mathbf{x})/T}}{\sum_{j=1}^{C_{old}^{b}} e^{\hat{o}_j(\mathbf{x})/T}}, q_c(\mathbf{x}) = \frac{e^{o_c(\mathbf{x})/T}}{\sum_{j=1}^{C_{old}^{b}} e^{o_j(\mathbf{x})/T}};$$

# Motivation

## 3. Limitation

- the cost of misclassifying old samples to new classes < that to other old class → bias towards new classes
  - old sample → correct old sample
  - old sample → wrong old sample
  - old sample → wrong new sample

$$\mathcal{L}_{KD}(\mathbf{x}) = \sum_{c=1}^{C_{old}^{b}} -\hat{q}_c(\mathbf{x}) \log\left(q_c(\mathbf{x})\right), \qquad (3)$$

$$\text{where } \hat{q}_c(\mathbf{x}) = \frac{e^{\hat{o}_c(\mathbf{x})/T}}{\sum_{j=1}^{C_{old}^{b}} e^{\hat{o}_j(\mathbf{x})/T}}, q_c(\mathbf{x}) = \frac{e^{o_c(\mathbf{x})/T}}{\sum_{j=1}^{C_{old}^{b}} e^{o_j(\mathbf{x})/T}};$$

# Methodology

## Overall Framework

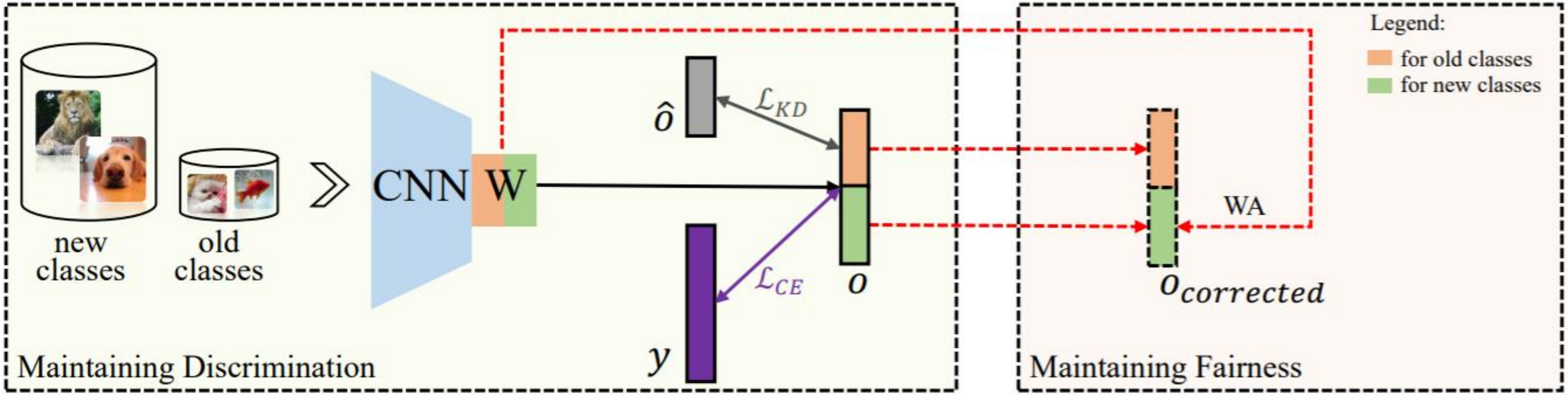


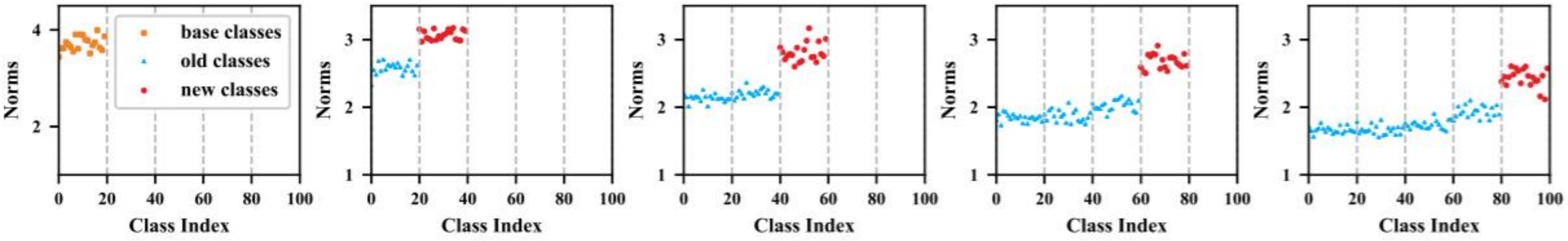
Figure 3: Overview of our solution for class incremental learning. In the first phase, we train the model with the cross-entropy loss ($\mathcal{L}_{CE}$) and the distillation loss ($\mathcal{L}_{KD}$). In the second phase, we correct the biased weights in the trained model via Weight Aligning (WA). $\mathbf{o}$ and $\hat{\mathbf{o}}$ represent the output logits of the current model and the old model respectively, $y$ stands for the true label, $\mathbf{o}_{corrected}$ represents the corrected output logits by using WA.

# Methodology

1) Maintaining discrimination

$$\mathcal{L}(\mathbf{x}, y) = (1 - \lambda)\mathcal{L}_{CE}(\mathbf{x}, y) + \lambda\mathcal{L}_{KD}(\mathbf{x})$$

2) Maintaining fairness: “Weight Aligning”



(a) $C^1 = 20, C^1_{old} = 0$ (b) $C^2 = 20, C^2_{old} = 20$ (c) $C^3 = 20, C^3_{old} = 40$ (d) $C^4 = 20, C^4_{old} = 60$ (e) $C^5 = 20, C^5_{old} = 80$

Figure 4: Norms of the weight vectors $\{\mathbf{w}_c\}$. (a) Results of the $1^{st}$ step (20 base classes), which does not correspond to class incremental learning; (b), (c), (d) and (e) are the results of the $2^{nd}$, $3^{rd}$, $4^{th}$, $5^{th}$ incremental step respectively, which show the norms of the weight vectors of new classes are much larger than those of old classes. (Best viewed in color)

* no bias term (convenience)

# Methodology

1) Maintaining discrimination

$$\mathcal{L}(\mathbf{x}, y) = (1-\lambda)\mathcal{L}_{CE}(\mathbf{x}, y) + \lambda\mathcal{L}_{KD}(\mathbf{x})$$

2) Maintaining fairness: “Weight Aligning”

$$\mathbf{o}(\mathbf{x}) = \begin{pmatrix} \mathbf{o}_{old}(\mathbf{x}) \\ \mathbf{o}_{new}(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \mathbf{W}_{old}^{T}\,\phi(\mathbf{x}) \\ \mathbf{W}_{new}^{T}\,\phi(\mathbf{x}) \end{pmatrix}$$

$$\widehat{\mathbf{W}}_{new} = \gamma \cdot \mathbf{W}_{new},$$

$$\mathbf{o}_{corrected}(\mathbf{x}) = \begin{pmatrix} \mathbf{W}_{old}^{T}\,\phi(\mathbf{x}) \\ \widehat{\mathbf{W}}_{new}^{T}\,\phi(\mathbf{x}) \end{pmatrix}$$

$$= \begin{pmatrix} \mathbf{W}_{old}^{T}\,\phi(\mathbf{x}) \\ \gamma \cdot \mathbf{W}_{new}^{T}\,\phi(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \mathbf{o}_{old}(\mathbf{x}) \\ \gamma \cdot \mathbf{o}_{new}(\mathbf{x}) \end{pmatrix}$$

$$\gamma = \frac{Mean(\boldsymbol{Norm}_{old})}{Mean(\boldsymbol{Norm}_{new})}$$

$$\boldsymbol{Norm}_{old} = (||\mathbf{w}_1||, \cdots, ||\mathbf{w}_{C_{old}^b}||),$$
$$\boldsymbol{Norm}_{new} = (||\mathbf{w}_{C_{old}^b+1}||, \cdots, ||\mathbf{w}_{C_{old}^b+C^b}||).$$

* Restriction to the weights (weight clipping)

# Experiment & Results

## Dataset: ImageNet ILSVRC 2012 & CIFAR-100

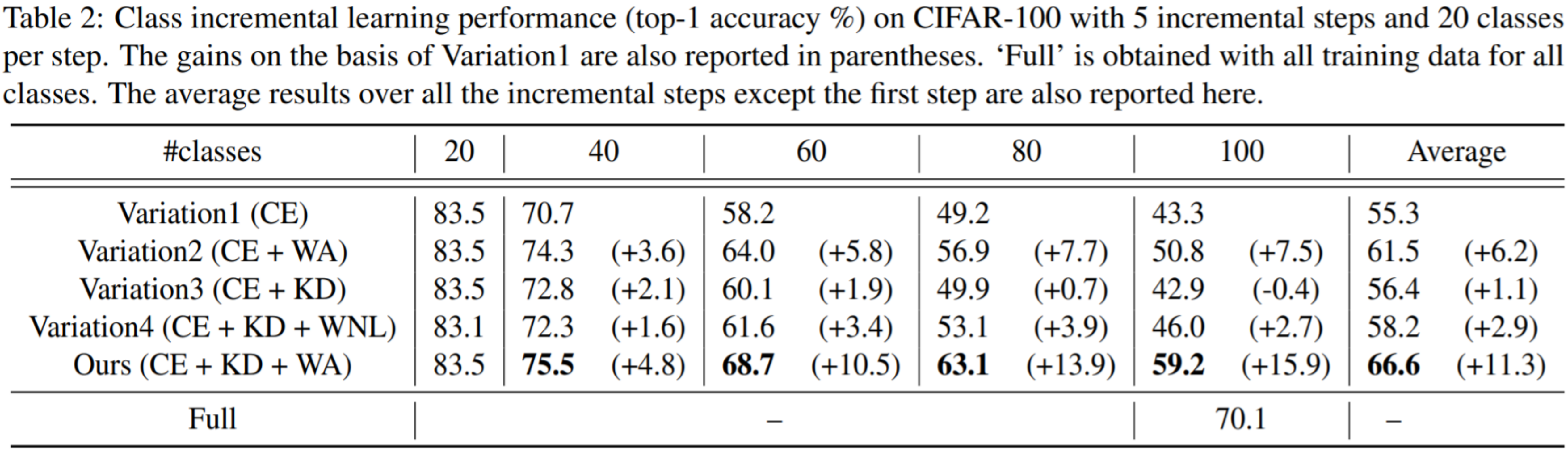


Table 2: Class incremental learning performance (top-1 accuracy %) on CIFAR-100 with 5 incremental steps and 20 classes per step. The gains on the basis of Variation1 are also reported in parentheses. 'Full' is obtained with all training data for all classes. The average results over all the incremental steps except the first step are also reported here.

| #classes | 20 | 40 | | 60 | | 80 | | 100 | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Variation1 (CE) | 83.5 | 70.7 | | 58.2 | | 49.2 | | 43.3 | | 55.3 | |
| Variation2 (CE + WA) | 83.5 | 74.3 | (+3.6) | 64.0 | (+5.8) | 56.9 | (+7.7) | 50.8 | (+7.5) | 61.5 | (+6.2) |
| Variation3 (CE + KD) | 83.5 | 72.8 | (+2.1) | 60.1 | (+1.9) | 49.9 | (+0.7) | 42.9 | (-0.4) | 56.4 | (+1.1) |
| Variation4 (CE + KD + WNL) | 83.1 | 72.3 | (+1.6) | 61.6 | (+3.4) | 53.1 | (+3.9) | 46.0 | (+2.7) | 58.2 | (+2.9) |
| Ours (CE + KD + WA) | 83.5 | **75.5** | (+4.8) | **68.7** | (+10.5) | **63.1** | (+13.9) | **59.2** | (+15.9) | **66.6** | (+11.3) |
| Full | – | | | | | | | 70.1 | | – | |

Weight normalization layer (WNL): the weights of all classes in FC layer have a unit norm during the training

# Experiment & Results

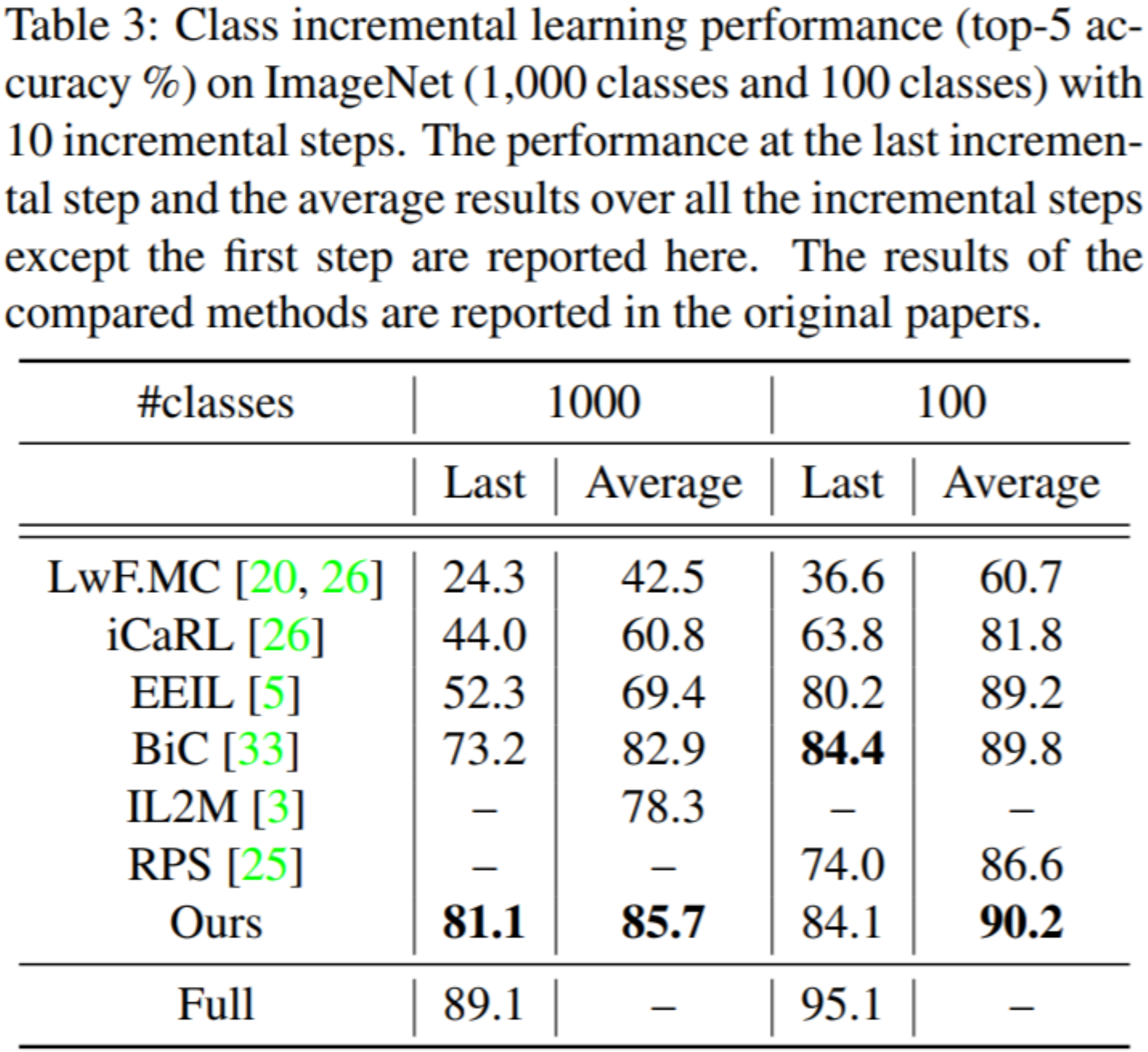


Table 3: Class incremental learning performance (top-5 accuracy %) on ImageNet (1,000 classes and 100 classes) with 10 incremental steps. The performance at the last incremental step and the average results over all the incremental steps except the first step are reported here. The results of the compared methods are reported in the original papers.

| #classes | 1000 | | 100 | |
|---|---|---|---|---|
| | Last | Average | Last | Average |
| LwF.MC [20, 26] | 24.3 | 42.5 | 36.6 | 60.7 |
| iCaRL [26] | 44.0 | 60.8 | 63.8 | 81.8 |
| EEIL [5] | 52.3 | 69.4 | 80.2 | 89.2 |
| BiC [33] | 73.2 | 82.9 | **84.4** | 89.8 |
| IL2M [3] | – | 78.3 | – | – |
| RPS [25] | – | – | 74.0 | 86.6 |
| Ours | **81.1** | **85.7** | 84.1 | **90.2** |
| Full | 89.1 | – | 95.1 | – |

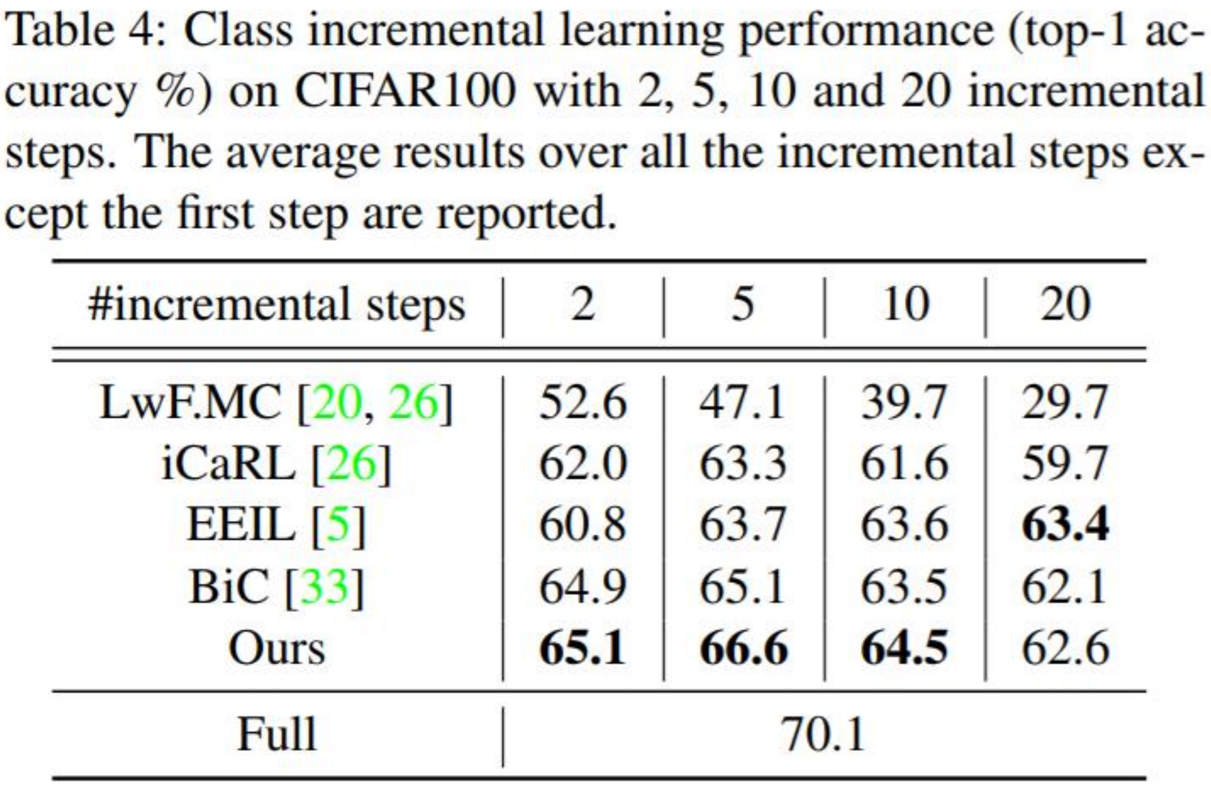
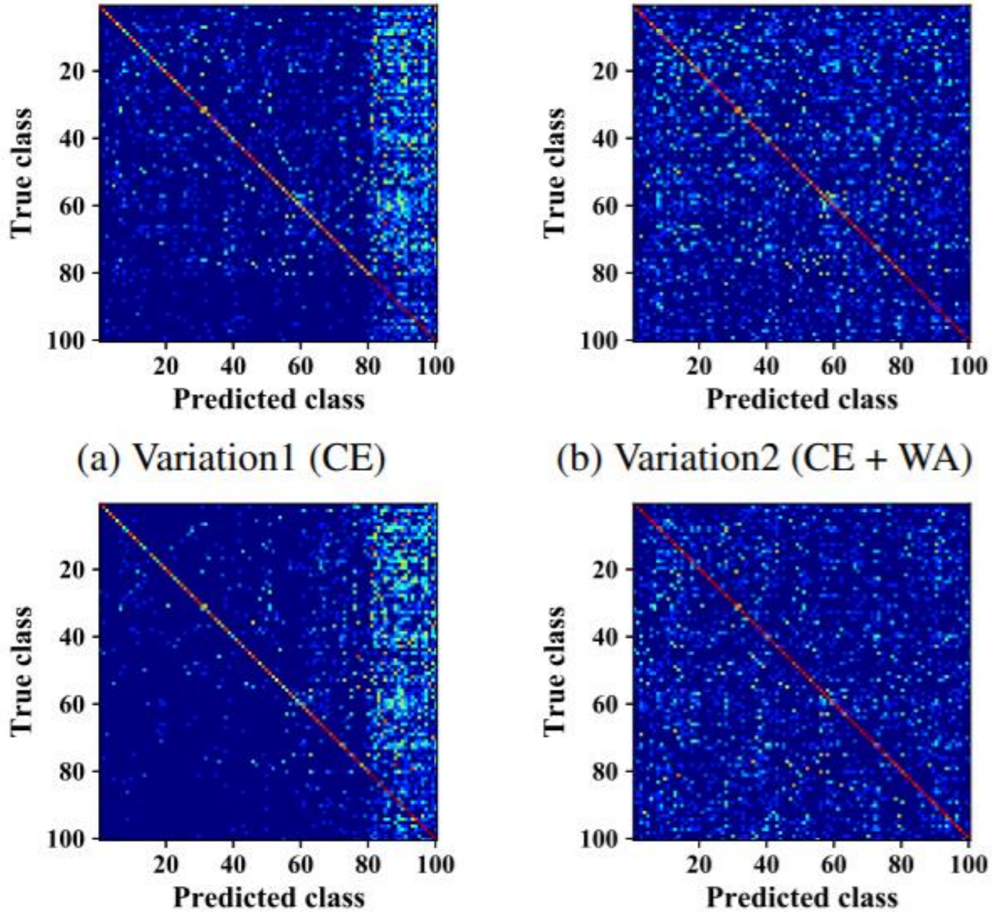


Table 4: Class incremental learning performance (top-1 accuracy %) on CIFAR100 with 2, 5, 10 and 20 incremental steps. The average results over all the incremental steps except the first step are reported.

| #incremental steps | 2 | 5 | 10 | 20 |
|---|---|---|---|---|
| LwF.MC [20, 26] | 52.6 | 47.1 | 39.7 | 29.7 |
| iCaRL [26] | 62.0 | 63.3 | 61.6 | 59.7 |
| EEIL [5] | 60.8 | 63.7 | 63.6 | **63.4** |
| BiC [33] | 64.9 | 65.1 | 63.5 | 62.1 |
| Ours | **65.1** | **66.6** | **64.5** | 62.6 |
| Full | 70.1 | | | |

# Experiment & Results



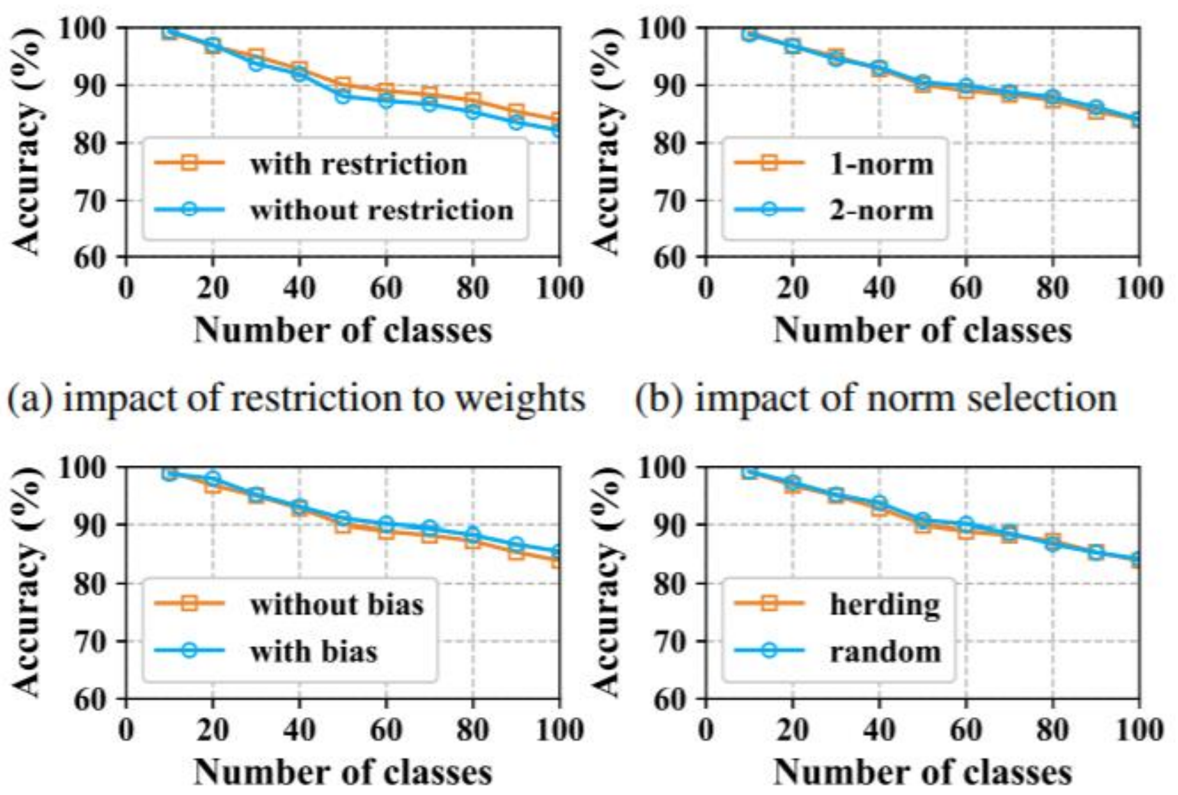
(a) Variation1 (CE)

(b) Variation2 (CE + WA)

(c) Variation3 (CE + KD)

(d) Ours (CE + KD + WA)

Figure 5: Confusion matrices of different approaches.



(a) impact of restriction to weights

(b) impact of norm selection

(c) impact of the bias term

(d) impact of exemplar selection

Figure 6: Class incremental learning performance (top-5 accuracy %) on ImageNet-100 for ablation study.

# Supplementary

# ImageNet result

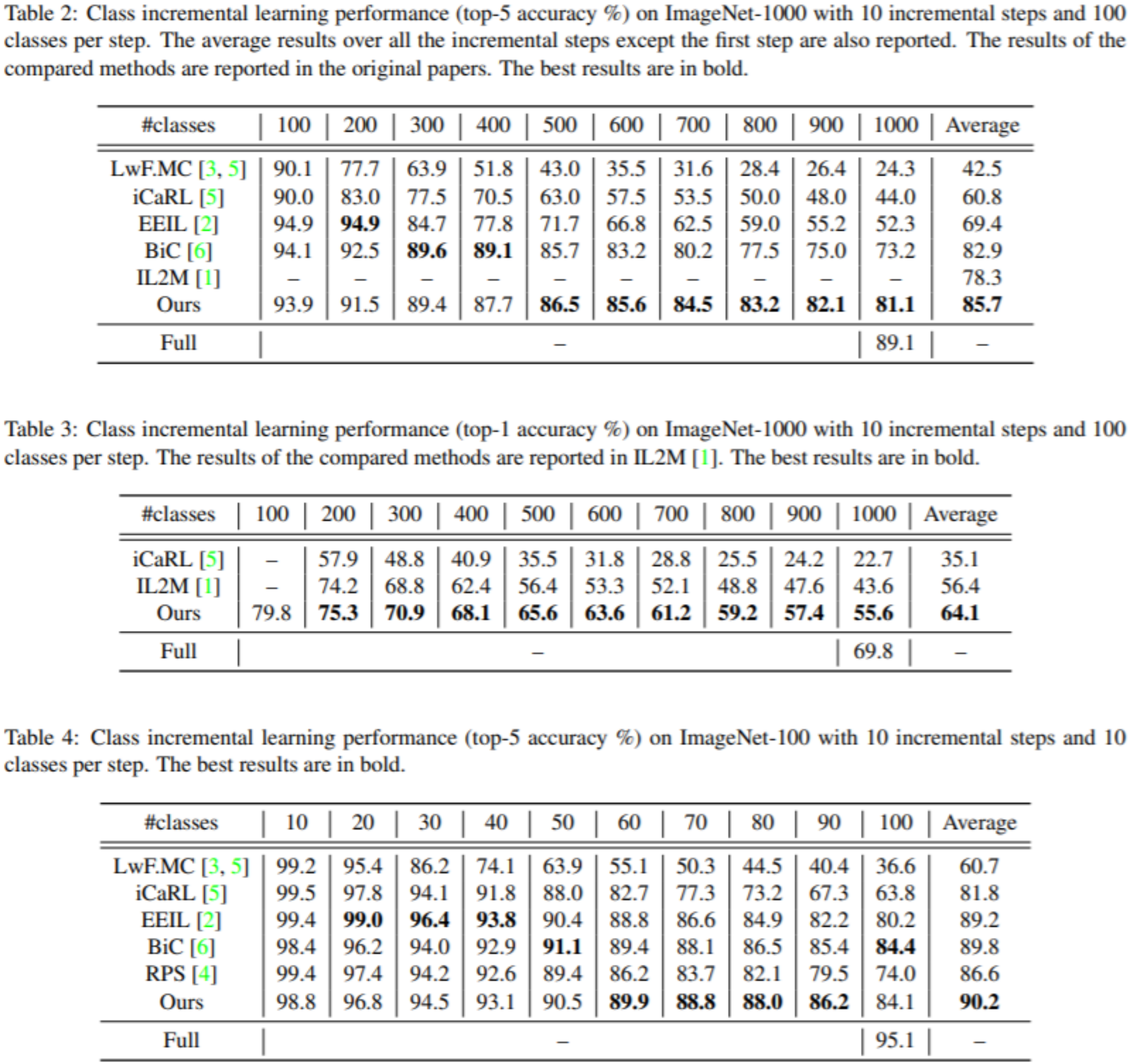


Table 2: Class incremental learning performance (top-5 accuracy %) on ImageNet-1000 with 10 incremental steps and 100 classes per step. The average results over all the incremental steps except the first step are also reported. The results of the compared methods are reported in the original papers. The best results are in bold.

| #classes | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LwF.MC [3, 5] | 90.1 | 77.7 | 63.9 | 51.8 | 43.0 | 35.5 | 31.6 | 28.4 | 26.4 | 24.3 | 42.5 |
| iCaRL [5] | 90.0 | 83.0 | 77.5 | 70.5 | 63.0 | 57.5 | 53.5 | 50.0 | 48.0 | 44.0 | 60.8 |
| EEIL [2] | 94.9 | **94.9** | 84.7 | 77.8 | 71.7 | 66.8 | 62.5 | 59.0 | 55.2 | 52.3 | 69.4 |
| BiC [6] | 94.1 | 92.5 | **89.6** | **89.1** | 85.7 | 83.2 | 80.2 | 77.5 | 75.0 | 73.2 | 82.9 |
| IL2M [1] | – | – | – | – | – | – | – | – | – | – | 78.3 |
| Ours | 93.9 | 91.5 | 89.4 | 87.7 | **86.5** | **85.6** | **84.5** | **83.2** | **82.1** | **81.1** | **85.7** |
| Full | – | | | | | | | | | 89.1 | – |

Table 3: Class incremental learning performance (top-1 accuracy %) on ImageNet-1000 with 10 incremental steps and 100 classes per step. The results of the compared methods are reported in IL2M [1]. The best results are in bold.

| #classes | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| iCaRL [5] | – | 57.9 | 48.8 | 40.9 | 35.5 | 31.8 | 28.8 | 25.5 | 24.2 | 22.7 | 35.1 |
| IL2M [1] | – | 74.2 | 68.8 | 62.4 | 56.4 | 53.3 | 52.1 | 48.8 | 47.6 | 43.6 | 56.4 |
| Ours | 79.8 | **75.3** | **70.9** | **68.1** | **65.6** | **63.6** | **61.2** | **59.2** | **57.4** | **55.6** | **64.1** |
| Full | – | | | | | | | | | 69.8 | – |

Table 4: Class incremental learning performance (top-5 accuracy %) on ImageNet-100 with 10 incremental steps and 10 classes per step. The best results are in bold.

| #classes | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LwF.MC [3, 5] | 99.2 | 95.4 | 86.2 | 74.1 | 63.9 | 55.1 | 50.3 | 44.5 | 40.4 | 36.6 | 60.7 |
| iCaRL [5] | 99.5 | 97.8 | 94.1 | 91.8 | 88.0 | 82.7 | 77.3 | 73.2 | 67.3 | 63.8 | 81.8 |
| EEIL [2] | 99.4 | **99.0** | **96.4** | **93.8** | 90.4 | 88.8 | 86.6 | 84.9 | 82.2 | 80.2 | 89.2 |
| BiC [6] | 98.4 | 96.2 | 94.0 | 92.9 | **91.1** | 89.4 | 88.1 | 86.5 | 85.4 | **84.4** | 89.8 |
| RPS [4] | 99.4 | 97.4 | 94.2 | 92.6 | 89.4 | 86.2 | 83.7 | 82.1 | 79.5 | 74.0 | 86.6 |
| Ours | 98.8 | 96.8 | 94.5 | 93.1 | 90.5 | **89.9** | **88.8** | **88.0** | **86.2** | 84.1 | **90.2** |
| Full | – | | | | | | | | | 95.1 | – |

# CIFAR result

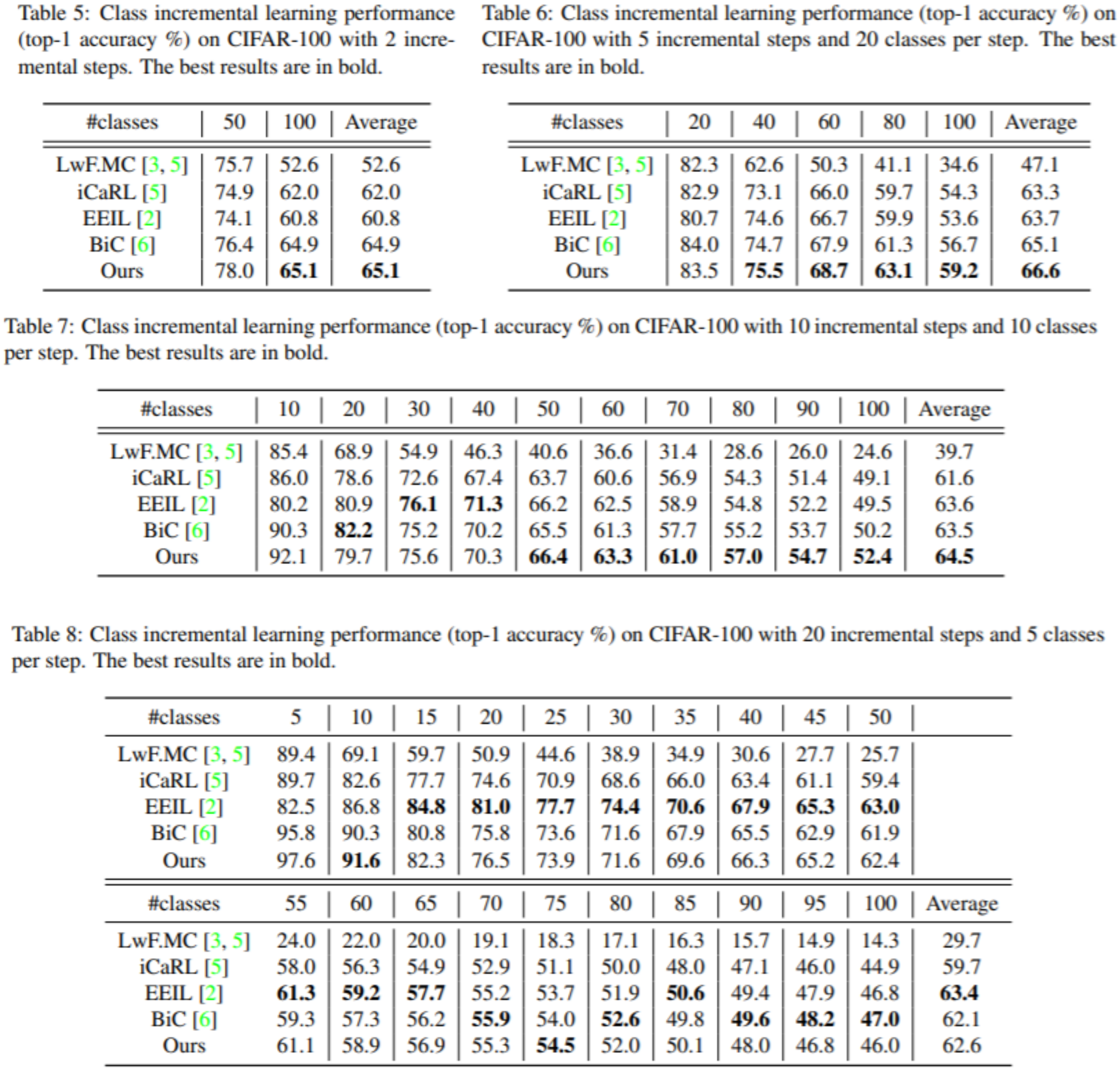


Table 5: Class incremental learning performance (top-1 accuracy %) on CIFAR-100 with 2 incremental steps. The best results are in bold.

| #classes | 50 | 100 | Average |
|---|---|---|---|
| LwF.MC [3, 5] | 75.7 | 52.6 | 52.6 |
| iCaRL [5] | 74.9 | 62.0 | 62.0 |
| EEIL [2] | 74.1 | 60.8 | 60.8 |
| BiC [6] | 76.4 | 64.9 | 64.9 |
| Ours | 78.0 | **65.1** | **65.1** |

Table 6: Class incremental learning performance (top-1 accuracy %) on CIFAR-100 with 5 incremental steps and 20 classes per step. The best results are in bold.

| #classes | 20 | 40 | 60 | 80 | 100 | Average |
|---|---|---|---|---|---|---|
| LwF.MC [3, 5] | 82.3 | 62.6 | 50.3 | 41.1 | 34.6 | 47.1 |
| iCaRL [5] | 82.9 | 73.1 | 66.0 | 59.7 | 54.3 | 63.3 |
| EEIL [2] | 80.7 | 74.6 | 66.7 | 59.9 | 53.6 | 63.7 |
| BiC [6] | 84.0 | 74.7 | 67.9 | 61.3 | 56.7 | 65.1 |
| Ours | 83.5 | **75.5** | **68.7** | **63.1** | **59.2** | **66.6** |

Table 7: Class incremental learning performance (top-1 accuracy %) on CIFAR-100 with 10 incremental steps and 10 classes per step. The best results are in bold.

| #classes | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LwF.MC [3, 5] | 85.4 | 68.9 | 54.9 | 46.3 | 40.6 | 36.6 | 31.4 | 28.6 | 26.0 | 24.6 | 39.7 |
| iCaRL [5] | 86.0 | 78.6 | 72.6 | 67.4 | 63.7 | 60.6 | 56.9 | 54.3 | 51.4 | 49.1 | 61.6 |
| EEIL [2] | 80.2 | 80.9 | **76.1** | **71.3** | 66.2 | 62.5 | 58.9 | 54.8 | 52.2 | 49.5 | 63.6 |
| BiC [6] | 90.3 | **82.2** | 75.2 | 70.2 | 65.5 | 61.3 | 57.7 | 55.2 | 53.7 | 50.2 | 63.5 |
| Ours | 92.1 | 79.7 | 75.6 | 70.3 | **66.4** | **63.3** | **61.0** | **57.0** | **54.7** | **52.4** | **64.5** |

Table 8: Class incremental learning performance (top-1 accuracy %) on CIFAR-100 with 20 incremental steps and 5 classes per step. The best results are in bold.

| #classes | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LwF.MC [3, 5] | 89.4 | 69.1 | 59.7 | 50.9 | 44.6 | 38.9 | 34.9 | 30.6 | 27.7 | 25.7 | |
| iCaRL [5] | 89.7 | 82.6 | 77.7 | 74.6 | 70.9 | 68.6 | 66.0 | 63.4 | 61.1 | 59.4 | |
| EEIL [2] | 82.5 | 86.8 | **84.8** | **81.0** | **77.7** | **74.4** | **70.6** | **67.9** | **65.3** | **63.0** | |
| BiC [6] | 95.8 | 90.3 | 80.8 | 75.8 | 73.6 | 71.6 | 67.9 | 65.5 | 62.9 | 61.9 | |
| Ours | 97.6 | **91.6** | 82.3 | 76.5 | 73.9 | 71.6 | 69.6 | 66.3 | 65.2 | 62.4 | |

| #classes | 55 | 60 | 65 | 70 | 75 | 80 | 85 | 90 | 95 | 100 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LwF.MC [3, 5] | 24.0 | 22.0 | 20.0 | 19.1 | 18.3 | 17.1 | 16.3 | 15.7 | 14.9 | 14.3 | 29.7 |
| iCaRL [5] | 58.0 | 56.3 | 54.9 | 52.9 | 51.1 | 50.0 | 48.0 | 47.1 | 46.0 | 44.9 | 59.7 |
| EEIL [2] | **61.3** | **59.2** | **57.7** | 55.2 | 53.7 | 51.9 | **50.6** | 49.4 | 47.9 | 46.8 | **63.4** |
| BiC [6] | 59.3 | 57.3 | 56.2 | **55.9** | 54.0 | **52.6** | 49.8 | **49.6** | **48.2** | **47.0** | 62.1 |
| Ours | 61.1 | 58.9 | 56.9 | 55.3 | **54.5** | 52.0 | 50.1 | 48.0 | 46.8 | 46.0 | 62.6 |