# Forgetting, Ignorance or Myopia: Revisiting Key Challenges in Online Continual Learning

**Anonymous Authors**[1]

## Abstract

Online continual learning (OCL) requires learning from endless data streams. Despite progress by recent methods, achieving satisfactory performance remains challenging, especially for large-scale data streams. In addition to that well-known catastrophic forgetting, we identify two under-noticed limiting factors: *model's ignorance and myopia.* The first one arises from the single-pass nature of OCL, limiting its ability to fully utilize the semantic information in data streams even *without catastrophic forgetting*. The second one stems from the inconsistency between the optimal solution for the current and global task, as the constant emergence of new tasks necessitates the continuous evolution of feature representations and classifiers for each category. To tackle these issues, we first empirically study the role of pre-trained initialization in the model's ignorance and myopia. Then, we theoretically highlight its indispensability for efficient problem-solving. In particular, based on the pre-trained initialization, we propose a novel non-sparse classifier evolution framework (NCE) to further address model's myopia and enhance throughput. NCE incorporates a non-sparse maximum separation regularization and targeted experience replay techniques, enabling the model to rapidly learn new globally discriminative features. Extensive experiments demonstrate the substantial improvements of our framework in both model performance and throughput, while reducing constraints on memory buffer.

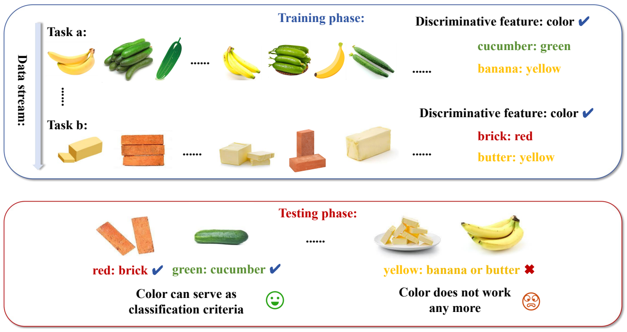
Training phase:
Data stream:
Task a:
......
Discriminative feature: color ✓
cucumber: green
banana: yellow
Task b:
Discriminative feature: color ✓
brick: red
butter: yellow
Testing phase:
red: brick ✓
green: cucumber ✓
yellow: banana or butter ✗
Color can serve as classification criteria
Color does not work any more

*Figure 1.* Color, which is the most discriminative feature for task a (banana vs. cucumber) and task b (butter vs. brick), is precisely the reason why the model confuses butter and banana.

## 1. Introduction

Online continual learning (OCL) is a learning paradigm that enables models to learn continuously from dynamic and non-stationary data streams. Although catastrophic forgetting has been extensively studied as the primary challenge in OCL(Mai et al., 2022; Wang et al., 2023; Zhou et al., 2023c), recent studies(Ghunaim et al., 2023) have shown that current OCL methods still struggle to achieve satisfactory performance even when utilizing a constantly replayed memory buffer. This struggle is particularly evident when handling large and volatile data streams. For instance, the accuracy of SOTA methods on the TinyImageNet dataset is reported to be only around 10%(Wei et al., 2023), let alone complex real-world scenarios. This raises the fundamental question of what factors limit the performance of OCL models and what are the underlying reasons behind them.

Our first focus is whether models have learned sufficient discriminative features during single-pass training, as this forms the basis for further analysis. If the model fails to develop substantial knowledge for individual task, concepts like forgetting is not even applicable. To examine this, we create a data stream comprising a single unified task, referred to as **single task setting**. Specifically, our testing encompasses CIFAR10, CIFAR100, EuroSat, CLEAR10, CLEAR100, and ImageNet datasets. We ensure that the data sampled from each task adheres to the same distribution, effectively minimizing interference between tasks. We test the model with and without pretrained initialization, considering different replay frequencies. We use the MAE pre-trained initialization for ImageNet and supervised pre-trained initialization for other datasets. The results, shown in Figure 2, indicate that models without pretrained initialization exhibit significantly lower classification accuracy than expected, especially on datasets with more categories like ImageNet and CIFAR100. Specifically, the accuracy of the model on these datasets is below 10%, which is more than 5 times lower than normal offline training. Plus, we find that even if the memory buffer is continuously used for replay, this problem remains difficult to alleviate. The single-pass nature of OCL prevents the model from fully leveraging the semantic information from the data stream, we call it as the **model's ignorance**. This issue emerges as a critical bottleneck that impedes the performance of existing OCL methods.

Our second focus is to delve deeper to the performance degradation phenomenon in OCL. We evaluate the model's learning using standard online continual learning setups by dividing the dataset into tasks. We analyze feature evolution and classification using nearest mean and linear softmax classifiers with detailed experimental settings in Appendix B.1). While previous studies often attribute this degradation to the model forgetting what it has learned, we argue that this explanation is not entirely true. Model's classification error for a specific class occurs abruptly during the training process. As illustrated in Figure 3 (blue), while a model initially maintains good classification accuracy for a specific class (eg. car) after a series of new tasks are introduced, there suddenly comes such a moment that model becomes completely confused, mistaking it for a new arrived class (eg. truck). It is counter-intuitive to solely attribute this phenomenon to the model forgetting past knowledge, as the process of forgetting should be gradual rather than sudden. Upon careful examination of the confused categories, we discover that this issue primarily stems from the fact that the discriminative features extracted by the model for a specific task (car vs. airplane) are often not transferable (it fails on car vs. truck). Models actually maintains the ability to distinguish between different categories within each individual task. *However, when only a limited range of categories is accessible to the model, its narrow focus on the current task restricts its capability to acquire desired features with broader discriminative power.* We term this limitation as the **model's myopia**. Intuitively, discriminative features for the current task may exactly be the cause of confusion when dealing with future classes, which means the model's cognition for each class must dynamically evolve with the arrival of new data, as demonstrated in Figure 1. We believe this issue is not exclusive to OCL. For any model requiring continuous learning of new knowledge, it is crucial to address the **model's myopia**.

In addition to performance, we also explore the throughput of current replay-based OCL models. As demonstrated in Figure 2), constantly replaying previous seen data (referred to as ER) with the goal of mitigating forgetting has shown to benefit the overall performance of the model. However, these approaches suffer from a significant reduction in model throughput and increased memory requirements, detailed in Figure 8 and Figure 9 (Appendix C), rendering them impractical for OCL (Jung et al., 2018; Ko et al., 2010). Inspired by how humans learn quickly and effectively, we realize our ability to recognize new class is typically built upon fundamental cognitive abilities and prior knowledge(Starr et al., 2013).It motivates us to explore the idea of leveraging additional knowledge from pre-trained models to effectively compensate for the model's ignorance and myopia. Our experiments, as illustrated in Figure 2, provide empirical evidence that employing an appropriate pre-trained initialization yields substantial enhancements in performance and efficiency across datasets, with improvements exceeding 50% on some datasets, effectively addressing the issue of **model's ignorance**. In addition to empirical verification, we also adopt a Pac-Bayes perspective to offer theoretical insights into why pre-training may be the few efficient solution when considering model throughput. Concerning **model's myopia**, we find that solely using pre-trained initialization does not completely solve the problem. While the pre-trained initialization can offer the model a broader perspective through prior knowledge, the emergence of an *overly sparse classifier* during training still causes the model to excessively focus on few discriminative features specialized for the current task. This exacerbates the issue of the model's myopia and limits its ability to consider a wider range of informative features. To tackle this issue, we introduce a novel framework called Non-sparse Classifier Evolution (NCE) that builds on the foundation of pre-trained initialization. NCE incorporates a non-sparse regularization term and employs maximum separation criteria between classes. This combination aims to address the issue of sparsity while preserving the model's ability to rapidly distinguish new classes. To further enhance model's throughput and reduce the requirements for a real-time memory buffer, we refine the replay scheme by selectively replaying past experiences with limited access. Our focus is on data that involve class confusion or highlight biases in the model.

By reidentifying and analyzing key challenges in OCL, our study not only enhances model's performance but also paves the way for the development of a high-throughput OCL system that is applicable in real-world scenarios.

## 2. Related Works

**Continual Learning.** Continual learning is a research field dedicated to learning continuously from incoming data while mitigating the forgetting of previously acquired knowledge(Oren & Wolf, 2021; Mirza et al., 2022; Garg et al., 2022; Mirza et al., 2022; Garg et al., 2022; Van de Ven & Tolias, 2019). Most continual learning approaches employ three types of techniques: regularization-based approaches, memory-based approaches and architecture-based approaches. Regularization-based approaches introduce regularization terms or constraints to the learning process to preserve previously learned knowledge(Kirkpatrick et al., 2017; Zenke et al., 2017; Aljundi et al., 2018; Dhar et al., 2019). Memory-based approaches utilize external memory buffers or replay mechanisms to store and replay past data, allowing the model to retain access to previous experiences(Guo et al., 2020; Zhu et al., 2021; Chaudhry et al., 2019a;b; Shim et al., 2021; Wang et al., 2019). Architecture-based approaches involve modifying the model architecture to facilitate continual learning(Sodhani et al., 2020; Yoon et al., 2018; Mallya & Lazebnik, 2018; Kim et al., 2022). Additionally, reducing storage overhead and minimizing dependence on hardware devices are issues of concern in the research community(Wang et al., 2022; Zhou et al., 2023a).

**Online Continual Learning.** Online continual learning (OCL) serves as a more realistic extension of continual learning. Unlike traditional batch learning, where the entire dataset for each task is available upfront, OCL operates in scenarios where data distributions dynamically change over time. In OCL, similar to memory-based approaches in CL, most methods leverage a real-time accessible memory buffer and employ various experience replay methods to mitigate the issue of forgetting(Chaudhry et al., 2019a;b; Aljundi et al., 2019a;c; Shim et al., 2021; Aljundi et al., 2019b; Chrysakis & Moens, 2020; Chaudhry et al., 2020; Rebuffi et al., 2017; de Masson D'Autume et al., 2019; Sokar et al., 2021; Wang et al., 2022; Caccia et al., 2022). Besides, other OCL methods aim to improve the learning of better features and classifiers in a single-pass training manner(Rebuffi et al., 2017). Techniques like contrastive learning(Mai et al., 2021; Cha et al., 2021), mutual information maximization(Gu et al., 2022; Guo et al., 2022b), and prototype learning(Zhu et al., 2021; Wei et al., 2023) have been employed to enhance the discriminative abilities of the model and improve its performance. Moreover, there are other works that focus a more proper evaluation of existing algorithms(Koh et al., 2022; Ghunaim et al., 2023). Compared with methods that aim for better performance, we focus on rethinking key challenges in OCL and then design a framework under more realistic throughput and storage constraints.

**Online Continual Learning with Pre-trained Models.** The utilization of pre-trained models has become a common approach in various machine learning tasks, including transfer learning(You et al., 2021; Chen et al., 2021), natural language processing(Devlin et al., 2018) and class-incremental learning(Mehta et al., 2023; Zhou et al., 2023b). While the effectiveness of pre-trained models has been well-established for these applications, only a few works(Lee et al., 2023) have explored their impact on OCL. These studies reveal underperforming algorithms can become very competitive when considering when using pre-trained models (He et al., 2022; Chen et al., 2020; Radford et al., 2021; Guo et al., 2022a). In this paper, we go beyond evaluating different pre-trained models in OCL settings. Our primary focus is to gain a deeper understanding of their impact on OCL, which sets our work apart from previous researches.

## 3. Problem Description and Preliminaries

In OCL, we aim to learn a sequence of tasks incrementally from a single-pass data stream $\mathfrak{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_T\}$, where $\mathcal{D}_t = \{x_i, y_i\}_{i=1}^{N_t}$ is the dataset of task $t$ following a distribution $\mu_t$, and $T$ is the total number of tasks. In a replay-based method, we denote the memory buffer as $\mathcal{B}$. For traditional methods that allow a real-time accessible memory buffer, at each time step $t$, the model receives a mini-batch data $X \cup X^b$ i.i.d drawn from $\mathcal{D}_t$ and memory buffer $\mathcal{B}$ respectively. Following (Chaudhry et al., 2018), we adopt the single-head evaluation setup where a classifier must choose from all seen classes when making inferences, to comprehensively measure overall performance. In addition, we introduce a predictor space $\mathcal{H}$ and a loss function $\ell : \mathcal{H} \times \mathcal{Z} \rightarrow \mathbb{R}^+$, which is bounded by a constant $K > 0$, $\mathcal{Z}$ denotes the training data. We denote $\mathcal{M}_1(\mathcal{H})$ as the set of all probability distributions on $\mathcal{H}$. Following the idea in online PAC-Bayes(Alquier et al., 2016; Haddouche & Guedj, 2022), we denote a sequence of distributions $(Q_i)_{i=0..T}$ as the learning process of an online continual model and $Q_0$ stands for the initial distribution on $\mathcal{H}$.

## 4. Key Challenges in OCL

In the context of OCL, the most extensively studied challenges are the issue of forgetting past knowledge when learning new tasks or acquiring new knowledge. However, as demonstrated in recent literature(Ghunaim et al., 2023; Wei et al., 2023), even SOTA algorithms struggle to achieve satisfactory performance, particularly when dealing with large volatile data streams. Despite the availability of techniques like gradient regularization, experience replay, and knowledge distillation to mitigate catastrophic forgetting, most methods still suffer from limited current task performance and significant performance degradation. In this section, we delve deeper into the specific factors causing such performance bottlenecks in OCL.

### 4.1. Ignorance: Dilemma of Insufficient Learning

Firstly, we demonstrate that in addition to catastrophic forgetting, the single-pass nature of OCL introduces significant insufficient learning problem that limits the performance of the model, denoted as **model's ignorance**. To isolate this challenge, we construct a data stream comprising a single unified task called **single task setting**. Each task $\mathcal{D}_i$ in the stream follows the same distribution $\mu_0$, eliminating any inter-task interference. By doing so, we create a unified task where data points from any class arrive at random timestamps, ensuring that class changes

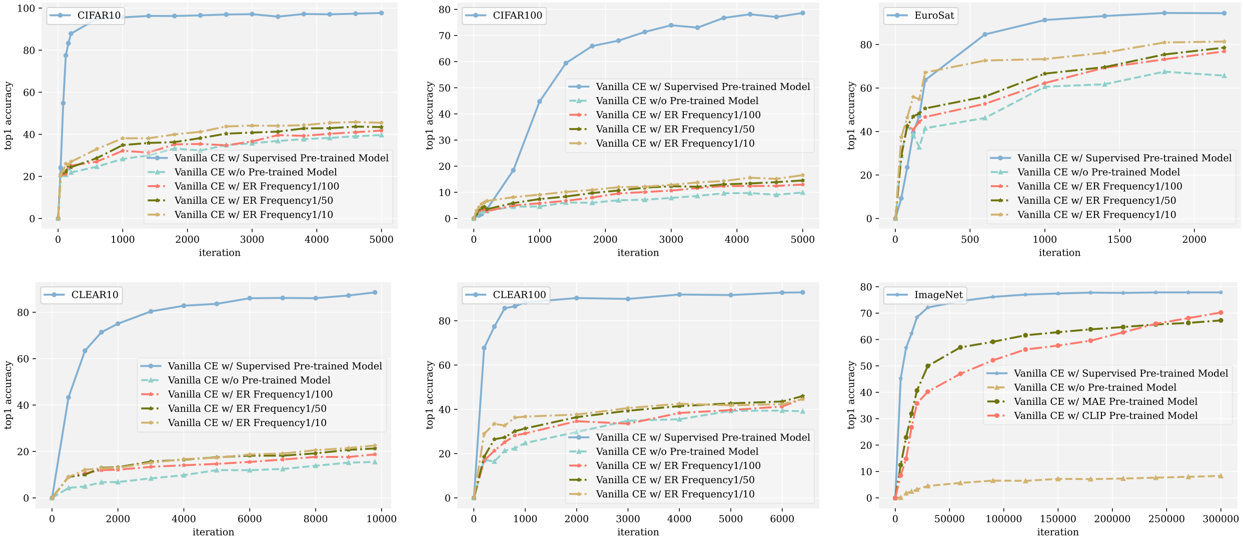

*Figure 2.* We evaluate the real-time accuracy of models on currently seen classes (w/) and (w/o) pre-trained models under our designed **single task setting**, as well as the impact of experience replay frequency on CIFAR, EuroSAT, CLEAR and ImageNet. More discussions on the effect of different pre-trained models and implementation details are in Appendix B.2.

in the data stream do not result in catastrophic forgetting. In this controlled setup, we train a simple supervised model optimized using cross-entropy loss $\mathcal{L}_{ce} = -\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\phi(f_c(x_i)))$ where $\phi(\cdot)$ represents the linear classifier, $f(\cdot)$ denotes the feature extractor. We then evaluate its performance both with and without the utilization of a pre-trained model to demonstrate the dilemma of insufficient learning in OCL and assess the impact of pre-trained models on the performance of the supervised model trained on our single-task stream.

As shown in Figure 2, even in the absence of inter-task interference, models trained from scratch struggle to achieve satisfactory performance (at least 2 times lower than offline training). The limited exposure to incoming data restricts model parameter updates to a single iteration, resulting in under-utilization of the available semantic information present in data stream. Continuous replay of the memory buffer can provide some relief to the **model's ignorance**, but it comes at the expense of significantly reduced model throughput and increased demands for memory buffer accessibility. However, in the context of OCL, the data stream does not wait for the model to finish training before presenting the next data for predictions. Therefore, efficiency becomes crucial, especially in scenarios that impose strict requirements on both model training and inference speed. Moreover, achieving a memory buffer that can be continuously accessed in real time is challenging in reality due to factors like network connectivity and privacy protection. These drawbacks all make it impractical to apply OCL in real-world applications, a more detailed discussion on the efficiency and feasibility is provided in Appendix C.

In such scenarios, leveraging additional knowledge becomes a natural choice. As depicted in our **single task setting** experiment, pre-trained initialization offers two advantages, as shown in Figure 2 and detailed in Appendix B.2. Firstly, it significantly enhances the performance upper limit of the OCL model. Secondly, unlike replay-based approaches that necessitate extensive training time to revisit seen data, pre-trained initialization provides abundant prior knowledge, facilitating much faster adaptation to new data. In addition to empirical analysis, we also provide theoretical evidence from the Pac-Bayes perspective, as outlined in Theorem 4.1. Although this bound is specifically used to illustrate the generalization risk for each task rather than a global risk, it serves as a valuable example of why a good initialization is one of the few beneficial strategies to **model's ignorance** when strictly adhering to the single-pass setting of OCL. The theoretical evidence highlights the significance of an effective initialization strategy in mitigating the challenges of OCL and improving overall performance.

**Theorem 4.1.** *For any distributions $\mu_1, ..., \mu_T$ over $\mathcal{Z}$, let $\mathcal{D}_t = (z_1^t, ..., z_{m_t}^t)$ be an iid set sampled from $\mu_t$ as the dataset of task t, for any $\lambda > 0$ and any online predictive sequence $(Q_0, Q_1, ..., Q_T)$, the following inequality holds*

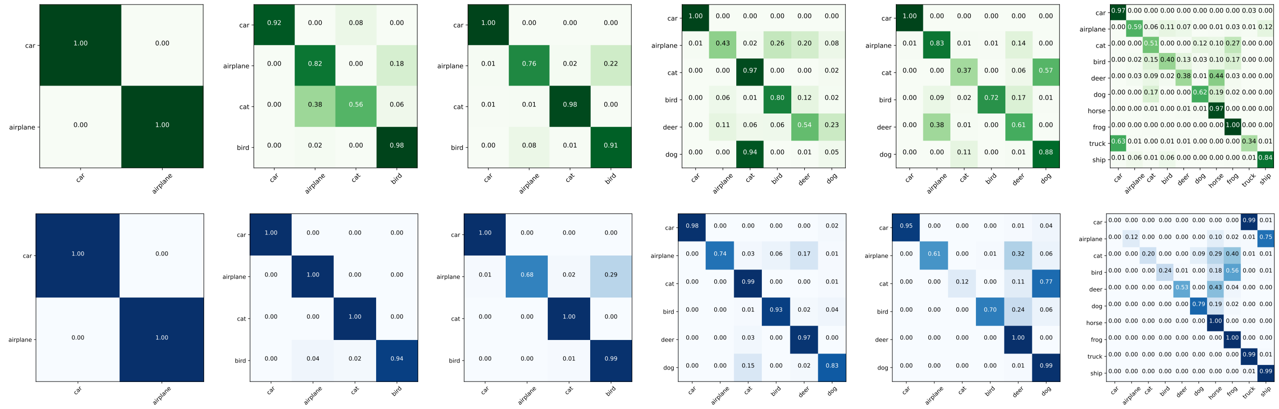

*Figure 3.* The normalized confusion matrix (CIFAR10) evolution of NCM classifier (green) and linear softmax classifier (blue) with supervised pre-trained models on ImageNet. Due to space limitations, we only present a partial training process in the main text. We provide a comprehensive display of the training process in Appendix D.

*with probability* $1-\delta$*:*

$$\sum_{t=1}^{T}\mathbb{E}_{h_t\sim Q_t}\left[\mathbb{E}_{z_t\sim\mu_t}[\ell(h_t,z_t)]\right] \le \sum_{t=1}^{T}\frac{\mathrm{KL}(Q_t\|Q_{t-1})}{\lambda} + \sum_{t=1}^{T}\sum_{j=1}^{m_t}\frac{\mathbb{E}_{h_t\sim Q_t}\left[\ell(h_t,z_j^t)\right]}{m_t} + \frac{\lambda T K^2}{2m_{\min}} + \frac{T\log(T/\delta)}{\lambda}. \tag{1}$$

Here $m_{\min}$ represents the minimum number of samples that models can receive for a task, $K$ denotes the upper bound of the loss function $\ell(.)$ and $T$ is the total task number.

*Remark* 4.2. In addition to empirical risk, the initial prior distribution $Q_0$ is one of the few selectable components before the training process, contributing to improved generalization. By effectively tightening the upper bound on the right-hand side, it highlights the importance of a good initialization. Furthermore, the third term in the bound emphasizes the significance of model throughput, as it directly impacts $m_{\min}$. Many existing OCL techniques, such as augmentation, knowledge distillation and gradient constraints all increase the training time, consequently reducing the amount of data ($m_{min}$) that the model can process. Proof and more detailed discussions are refered to Appendix A.

### 4.2. Myopia: Key Factor for Performance Degradation

We already discussed the ability of pre-trained models to alleviate the issue of **model's ignorance**. However, it is notable that solely relying on pre-trained initialization cannot guarantee a satisfying overall performance. Degradation in performance still frequently occurs. Current research on continual learning often attributes this to the phenomenon of catastrophic forgetting, which is caused by an excessive emphasis on the current task or class, leading to interference with previously learned knowledge (Wang et al., 2023). Additionally, in the context of OCL, some studies suggest it is due to the model learning ungeneralized trivial features(Guo et al., 2022b; Wei et al., 2023). In this section, our objective is to examine the issue of performance degradation when using pre-trained representations and conduct a comprehensive analysis of the factors contributing to it. To achieve this, we train the model using pre-trained initialization and vanilla cross-entropy loss (Equation (2)), without employing any techniques to mitigate forgetting or trivial learning.

$$\mathcal{L}_{ce} = -\sum_{t=1}^{T}\sum_{i=1}^{N_t}\sum_{c=1}^{C_t} y_i^c \log(\phi^c(f(x_i))). \tag{2}$$

We assess the discrimination ability of the classifier and feature extractor by evaluating the model's performance using a linear softmax classifier(Chaudhry et al., 2019b) and an online updating NCM (Nearest Class Mean) classifier(Rebuffi et al., 2017) which are both very common in OCL. Specifically, when the model receives new classes, the softmax classifier need to be updated: $\phi_c(.) \rightarrow \phi_{c+1}(.)$. In practice, researchers often simplify the updates by using a pre-designed linear classifier with parameters that are typically higher than the number of classes. For NCM classifier, we compute a dynamic class mean prototype for each class:

$$\begin{gathered}\mu_c^{new} = (1-\lambda)\mu_c^{old} + \lambda\frac{1}{n_c}\sum_i f(x_i)\cdot\mathbb{I}\{y_i = c\},\\ y^* = \underset{c=1\ldots C}{\arg\min}\,||f(x)-\mu_c||.\end{gathered} \tag{3}$$

Class mean prototype $\mu_c$ is based on the $n_c$ data points from class $c$ in the current data stream and old prototypes. We employ a momentum update scheme with the parameter $\lambda$.

Even though pre-trained initialization offers a broader perspective to the model, enabling it to leverage prior knowledge, as depicted in Figure 3, performance degradation remains a common occurrence when new tasks or data are introduced. Specifically, the linear softmax classifier (blue) demonstrates a rapid acquisition of improved discriminative abilities for data within the current task. However, it is more prone to misclassifying categories from previous tasks as belonging to the current task, leading to a decline in overall performance. On the other hand, the NCM classifier (green) is more susceptible to achieving sub-optimal classification performance for the current class. It is also noticeable that both two classifiers get confused by classes sharing similar semantic information, such as color, texture, or background style. When we take a close look at those classes that model gets confused during training, such as car and cat. Distinguishing them suddenly becomes challenging when highly discriminative features from past tasks (e.g., shape, background for airplane vs. car, texture for cat vs. bird) appear in new classes (car and truck share similar shapes and backgrounds, cat and dog share similar texture). Such confusions exist at both the classifiers head (linear classifier) and the features (NCM classifier).

Previous literature often attributes such confusion in different classes to what is known as catastrophic forgetting or interference from spurious relationships. However, we propose a different perspective. We believe that the confusion arises because the discriminative features or knowledge acquired from previous tasks may not be helpful in distinguishing these classes from some new classes in future tasks from the outset. In other words, the previously learned representations may not capture the essential characteristics necessary for effectively differentiating these classes with new classes and this is something perfectly normal even for us as humans. We still take task 1 (airplane vs. car) as an example. In fact, the model never loses the ability to distinguish between airplanes and cars. After all, a highly discriminative feature must be considered in the context of a specific task but the single pass nature of OCL makes it impossible for some categories to coexist simultaneously for the training. During the training process, the model naturally focuses on features and discriminant criteria that are more important for the current task. The independent arrival of tasks results in such **a myopic model**, which is the key factor in the decline of OCL performance. For a comprehensive visualization of the training process and further analysis (with or without pre-trained initialization), please refer to Figure 14 in Appendix D.

In addition to feature-level confusions, a biased classifier can also lead to performance degradation. As depicted in Figure 3, we can observe that, during the training process, the classification results of NCM and softmax classifier are inconsistent. Specifically, we notice that while the extracted features from the model are often separable, a biased classifier can still result in significant confusion between categories. Multiple prior studies in continual learning(Belouadah & Popescu, 2019; Hou et al., 2019; Wu et al., 2019) have consistently witnessed that the models utilizing the softmax classifier exhibit a pronounced prediction bias towards the recent task. This bias is often attributed to the decreasing mean weights for the new classes in the final linear layer (fully connected layer in classification head).

In this paper, we propose an alternative viewpoint that the direct reason for the model losing its classification ability in OCL is not solely due to the decrease in classifier weights, especially when using pre-trained initializations. As demonstrated in Figure 4, regardless of using pre-trained models or not, the weights of the classifier for previous tasks decrease with the continuous arrival of new tasks. However, unlike training from scratch, when using a pre-trained initialization, the model does not arbitrarily classify class 1 (car) as a category in the current task. This indicates that the decrease in weights alone is not the primary reason for the bias observed in the classifier.

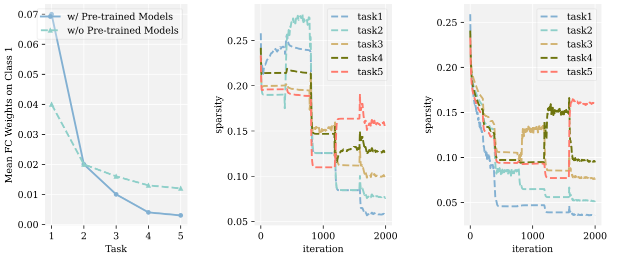

*Figure 4.* The **left** figure displays the weight means in the classifier (final FC layer) for class 1 (car) on CIFAR10. The **middle** figure shows the sparsity of the classifier for classes corresponding to different tasks, with a pre-trained initialization. The **right** figure represents the sparsity of the classifier for the same classes as **middle**, without a pre-trained initialization.

As previously analyzed, models naturally prioritizes the features and discriminative criteria that are crucial for the current task at hand. However, an excessive emphasis on the current task often leads to the model relying solely on a small number of discriminative features for classification. To evaluate the model's risk of myopia, we measure the sparsity of parameters in the linear classifier for each task using the method outlined in Equation (4). We represent the final fully connected layer as a matrix $W \in \mathcal{R}^{d\times C}$, where $d$ stands for the feature dimension and $C$ is the number of classes. In Eq.4, the variable $w$ denotes a column vector extracted from $W$, with $w \in \mathcal{R}^{d}$.

$$S(w) = \frac{(|w_1| + |w_2| + \cdots + |w_d|)/d}{\max(|w_1|, |w_2|, \cdots, |w_d|)} \tag{4}$$

We pose the question of whether, during the training process in OCL, the model's criteria for judging a category gradually become simpler. In other words, the model may start focusing solely on a limited set of discriminative features that it deems beneficial for the current task. This simplification of criteria, as illustrated in Figure 4, is characterized by the increasing sparsity of parameters associated with old tasks as new tasks continue to arrive. While relying on few discriminative features may not be problematic in traditional supervised learning paradigms, in the context of OCL, it leads to inevitable confusion between categories when these few discriminative features appear in new task categories.

## 5. Method

After conducting a series of analysis on the key challenges in OCL, we emphasize the significance of leveraging prior knowledge in this context, particularly when dealing with large-scale data streams. The framework we propose below is built upon the utilization of pre-trained initialization.

**Non-sparse regularization.** Unlike some previous methods that aim to acquire task-specific features capable of generalization, in this study, we acknowledge the unrealistic expectation of obtaining a model with absolute discriminative ability within a limited scope of classes, even with the rich prior knowledge provided by a pre-trained model. Instead, our focus lies in ensuring the diversity of discriminant features during training and enabling the model to swiftly develop the ability to differentiate between categories from different tasks.

As posited in the previous section, in the context of OCL, contrary to traditional settings where a sparse classifier is often considered desirable for achieving high classification performance(Dedieu et al., 2021; Levy & Abramovich, 2023), the overly sparse parameters can cause the model to focus solely on a limited set of highly discriminative features, increasing the risk of model's myopia. To mitigate this issue, we propose a straightforward idea of constraining the sparsity calculated in Equation (4). Our goal is to ensure that the model maintains a diverse set of discriminative features during training, allowing it to effectively handle different tasks without being overly biased towards specific features. However, considering that the $\max(\cdot)$ function is easily affected by a small number of outliers in the parameters, we opt to replace it with the $l2$ norm as a more robust alternative in our sparsity regularization:

$$\mathcal{L}_s = -\frac{(|w_1| + |w_2| + \cdots + |w_d|)/d}{\sqrt{w_1^2 + w_2^2 + \cdots + w_d^2}}. \tag{5}$$

It should be noted that our intention is to only constrain the corresponding classifier parameters (with dimension of $d$) of non-current task categories here. During experiments, we observed that excessive sparsity of task-corresponding parameters also occurs during the training of the current task. In such cases, an alternative solution could be to halt the gradient backpropagation for a portion of the network. More discussions are provided in Appendix B.3.3.

**Maximum separation.** While a smooth classifier can help to mitigate the model's myopia, it sometimes hinders the model's ability to rapidly perform classification in the current task. Moreover, in OCL, it is also hard to have simultaneous access to data from all categories, especially when there are restrictions on the use of memory buffers. This leads to severe class imbalance during the learning process, which is a well-recognized challenge in the context of continual learning(Yang et al., 2023b; Kasarla et al., 2022). Meanwhile, it can be proved that the popular consistency-based or contrastive learning techniques used in OCL methods further exacerbate this problem(Zhong et al., 2022).

Thus, we draw inspiration from the famous Neural Collapse (Papyan et al., 2020) and Maximum Class Separation criterion(Kasarla et al., 2022). For learned representations from different categories $\{f(x_1), f(x_2), \cdots f(x_{C_t})\}$, their cosine similarity should satisfy a maximum separation criterion and converge to an ideal simplex ETF, $\forall_{i,j,i\neq j}\langle f(x_i), f(x_j)\rangle = -\frac{1}{C_t - 1}$.

$$\begin{aligned} \mathcal{L}_p =& \frac{1}{C_t^2}\sum_{i,j=1}^{C}(\langle f(x_i), f(x_j)\rangle - p_{ij})^2, \\ p_{ij} =& \frac{C_t}{C_t - 1}\delta_{i,j} - \frac{1}{C_t - 1} \end{aligned} \tag{6}$$

where $\delta_{i,j}$ is Kronecker delta symbol that designates the number $1$ if $i = j$ and $0$ if $i \neq j$. To address categories not present in the current task, we use the class mean in Equation (3) to replace the representation of the corresponding category. Thus, we denote our total loss function as:

$$\mathcal{L} = \mathcal{L}_{ce} + \gamma(\mathcal{L}_p + \mathcal{L}_s) \tag{7}$$

**Targeted experience replay.** Despite our efforts to mitigate model's myopia, occasional class confusion still occurs. To address this promptly, we prioritize the categories that the model has previously struggled to distinguish when we have access to the memory buffer. During experience replay, we compute a confusion matrix to identify these frequently confused categories. To tackle each group of confused categories, we devise separate binary classification loss which is designed to expedite the model's acquisition of discriminative abilities between these confusion classes.

$$\begin{aligned} \mathcal{L}_b = -\sum_{i=1}^{|\mathcal{B}|}\sum_{m,n=1}^{C}\mathbb{I}\{\mathcal{C}_{m,n}^b > \tau\} \cdot [y_i^m \log(\phi^m(f(x_i))) \\ + y_i^n \log(\phi^n(f(x_i)))], m \neq n. \end{aligned} \tag{8}$$

Here, $\mathcal{C}^b$ represents a normalized confusion matrix, and the condition $\mathcal{C}^b_{m,n} > \tau$ indicates that the proportion of data belonging to class $m$ being classified as class $n$ exceeds the threshold $\tau$.

## 6. Experiments

Before delving into the specifics of experimental results, we highlight a significant distinction in the utilization of memory buffers between our study and other works. In this work, we impose limitations on the number of requests allowed to retrieve data from the memory buffer. We evaluate our models on six image datasets, incorporating realistic task overlaps to mimic practical scenarios. Training is harmonized using ViT architectures and the AdamW optimizer, with consistent batch size and initialization (MAE pre-training for ImageNet, supervised pre-training for others). We compare our NCE method against 13 baselines, ensuring fairness by applying vanilla experience replay for a single epoch to all. To better capture models' performance across time, we use $A_{AUC}$ and Last Accuracy as metrics, enhancing the understanding of models' anytime inference capabilities beyond the traditional Average accuracy ($A_{avg}$). Due to limited space, we leave detailed descriptions of the implementation, evaluation metrics, datasets and comparison methods in Appendix B.1.

### 6.1. Main Results and Analysis

Here, we conduct a comprehensive evaluation of our method by comparing its performance with several existing state-of-the-art OCL methods as well as various continual learning variants. Table 1 displays the $A_{AUC}$ (area under the accuracy curve) for three synthetic benchmark datasets, showcasing the impact of different memory buffer sizes and replay frequencies. This evaluation metric offers a more comprehensive assessment compared to the commonly used average accuracy(Koh et al., 2022). The results demonstrate that our proposed method, NCE, consistently outperforms other approaches. Notably, the performance improvement achieved by NCE is particularly significant when the memory buffer size is relatively small and the number of memory buffer access times is very limited. This finding highlights the effectiveness of NCE in scenarios where memory capacity or access frequency are constrained. In addition to the commonly used synthetic datasets, we further evaluate our methods on two real-world domain incremental datasets and a large-scale image classification dataset. These inclusions allow us to assess the performance and generalizability of our approach in real-world scenarios, where the challenges and characteristics may differ. Table 2 demonstrates that NCE can also enhance the performance in real-world domain incremental settings and complex data streams. Due to limited space, more experimental results including model's last time accuracy, sensitivity analysis and evaluation on model throughput are provided in Appendix B.3.

### 6.2. Ablation Studies

To investigate the specific effects of different proposed components, we conduct a series of ablation studies. From Table 3, we can draw several observations: (1) For class incremental scenarios, each component we propose provides performance improvements, among which targeted ER has the most obvious effect. (2) Constraints on classifier sparsity, as defined by $\mathcal{L}_s$, do not result in improved model performance when new tasks do not involve the addition of new categories. It aligns with the intuition that in scenarios where there are no new categories introduced, the model does not face the myopia challenge.

## 7. Conclusion

In this study, we conduct a thorough reevaluation of the major challenges in current OCL methods. We delve into the underlying causes of these challenges and explore the potential influence of pre-trained models. Our analysis highlights two critical limiting factors: **model's ignorance and myopia**, which can have a more significant impact than the widely recognized issue of catastrophic forgetting in the context of OCL. Furthermore, we propose the NCE framework, which takes into account performance, throughput, and practicality considerations. Our work aims to provide a fresh perspective and inspire the OCL field to prioritize efficient learning in practical real-world scenarios.

*Table 1.* $A_{AUC}$ on online class-incremental setting. Best is highlighted in **bold**, second best is shown underlined.

| Method | CIFAR-10 | | | CIFAR-100 | | | EuroSat | | |
|---|---|---|---|---|---|---|---|---|---|
| | $M=0.1k$ | $M=0.2k$ | $M=0.5k$ | $M=0.5k$ | $M=1k$ | $M=2k$ | $M=0.1k$ | $M=0.2k$ | $M=0.5k$ |
| | $Freq=1/100$ | $Freq=1/50$ | $Freq=1/10$ | $Freq=1/100$ | $Freq=1/50$ | $Freq=1/10$ | $Freq=1/100$ | $Freq=1/50$ | $Freq=1/10$ |
| iCaRL(Rebuffi et al., 2017) | 80.6±0.5 | 83.9±0.4 | 88.2±0.4 | 55.1±0.2 | 57.9±0.4 | 67.7±0.2 | 58.7±0.4 | 75.2±1.1 | 80.4±0.7 |
| EWC(Kirkpatrick et al., 2017) | 81.7±0.7 | 85.5±1.2 | 91.2±0.7 | 60.7±0.8 | 62.9±3.2 | 67.2±1.1 | 61.0±0.8 | 72.6±0.5 | 83.9±1.1 |
| DER++(Buzzega et al., 2020) | 81.5±1.2 | 86.7±0.8 | 89.9±1.0 | 59.2±0.9 | 61.1±0.8 | 69.2±1.4 | 45.0±6.0 | 78.2±2.4 | 81.9±2.1 |
| PASS(Zhu et al., 2021) | 82.0±0.8 | 85.2±0.6 | 90.3±1.2 | 61.2±1.3 | 62.9±0.9 | 67.0±1.5 | 50.1±3.1 | 78.1±1.2 | 83.5±0.8 |
| AGEM(Chaudhry et al., 2019a) | 78.6±0.7 | 81.2±1.1 | 85.7±0.9 | 50.2±0.7 | 58.9±1.1 | 67.4±0.8 | 56.4±0.7 | 67.7±0.9 | 81.9±1.0 |
| ER(Chaudhry et al., 2019b) | 82.6±0.5 | 85.4±0.4 | 91.2±0.2 | 61.3±0.3 | 64.6±0.4 | 71.2±1.2 | 58.6±0.8 | 70.5±0.6 | 84.0±0.8 |
| MIR(Aljundi et al., 2019a) | 82.4±0.4 | 85.7±0.7 | 89.9±1.0 | 62.9±0.6 | 63.3±0.7 | 71.2±1.4 | 59.0±1.0 | 71.1±0.9 | 84.2±1.1 |
| ASER(Shim et al., 2021) | 80.7±1.2 | 84.4±0.6 | 87.2±1.0 | 60.1±0.8 | 62.3±0.9 | 70.9±1.4 | 59.4±1.0 | 72.0±0.8 | 83.7±0.4 |
| SCR(Mai et al., 2021) | 83.8±0.2 | 85.9±1.4 | 90.5±0.9 | 61.5±0.4 | 62.7±0.2 | 71.2±0.4 | 52.1±0.8 | 75.9±0.7 | 84.8±0.6 |
| DVC w/o Aug(Gu et al., 2022) | 80.5±0.2 | 85.9±0.3 | 89.2±0.7 | 57.9±0.6 | 58.9±0.6 | 67.4±0.8 | 52.0±0.9 | 69.1±0.8 | 82.7±1.1 |
| DVC(Gu et al., 2022) | 81.1±0.2 | 85.8±0.4 | 90.3±0.5 | 61.6±0.8 | 62.9±1.0 | 70.7±0.9 | 53.6±0.7 | 72.7±1.1 | 85.3±1.0 |
| OCM w/o Aug(Guo et al., 2022b) | 79.1±1.5 | 83.3±1.4 | 90.1±2.0 | 60.9±0.8 | 58.4±1.6 | 69.5±0.4 | 46.7±1.2 | 78.6±1.0 | 83.1±0.4 |
| OCM w/ Aug(Guo et al., 2022b) | 82.1±2.9 | 85.2±2.1 | 90.2±2.7 | 61.3±1.5 | 60.3±1.1 | 70.2±0.7 | 51.9±1.4 | 76.8±1.2 | 84.0±0.9 |
| OnPro w/ Aug(Wei et al., 2023) | 81.1±0.6 | 86.1±0.7 | 90.1±0.8 | 62.9±0.7 | 63.7±0.8 | 70.5±0.9 | 52.8±6.8 | 75.2±1.0 | 83.7±0.9 |
| MC-SGD w/ SAM(Mehta et al., 2023) | 81.8±0.5 | 83.9±1.2 | 90.5±0.4 | 60.2±0.8 | 63.1±0.8 | 71.8±0.8 | 61.3±0.9 | 78.9±0.5 | 84.6±1.0 |
| NCE | **89.9±0.4** | **90.4±0.4** | **90.7±1.0** | **74.1±0.7** | **75.5±0.8** | **79.7±0.9** | **75.7±0.4** | **83.4±0.7** | **86.3±0.4** |

*Table 2.* $A_{AUC}$ on real-world online domain-incremental setting and large scale data stream. Best is highlighted in **bold**, second best is shown underlined.

| Method | CLEAR-10 | | CLEAR-100 | | ImageNet |
|---|---|---|---|---|---|
| | $M=0.1k$ | $M=0.2k$ | $M=1k$ | $M=2k$ | $M=10k$ |
| | $Freq=1/100$ | $Freq=1/50$ | $Freq=1/100$ | $Freq=1/50$ | $Freq=1/500$ |
| ER | 87.3±1.0 | 87.9±0.5 | 80.1±0.6 | 82.0±0.8 | 55.6±0.4 |
| DER++ | 87.4±0.5 | 88.1±0.9 | 78.5±1.1 | 80.4±0.6 | 46.5±0.4 |
| EWC | 88.0±0.4 | 89.0±0.2 | 81.1±0.3 | 81.2±0.5 | 50.9±1.0 |
| iCarl | 86.2±0.8 | 87.1±1.1 | 77.1±0.8 | 80.4±0.9 | 57.1±1.8 |
| SCR | 84.1±0.7 | 85.9±0.6 | 67.2±0.7 | 79.7±0.4 | 52.5±2.4 |
| OCM | 88.1±0.5 | 89.2±0.6 | 80.0±0.9 | 82.5±1.0 | ××××× |
| DVC | 86.4±0.3 | 87.0±0.7 | 79.4±0.2 | 80.2±0.7 | 53.1±0.6 |
| OnPro | 86.9±1.4 | 88.1±0.9 | 80.4±0.3 | 82.3±0.2 | 52.2±0.4 |
| NCE | **89.2±0.7** | **91.4±0.8** | **84.3±0.4** | **85.7±0.3** | **61.6±0.7** |

*Table 3.* Ablation study.

| Model | CIFAR10 | CIFAR100 | EuroSat | CLEAR100 | ImageNet |
|---|---|---|---|---|---|
| | $M=0.1k$ | $M=0.5k$ | $M=0.1k$ | $M=1k$ | $M=10k$ |
| | $Freq=1/100$ | $Freq=1/100$ | $Freq=1/100$ | $Freq=1/100$ | $Freq=1/500$ |
| vanilla $\mathcal{L}_{ce}$ | 82.6±0.5 | 61.3±0.3 | 58.6±0.8 | 80.1±0.6 | 55.6±0.4 |
| vanilla w/ $\mathcal{L}_s$ | 84.5±1.3 | 64.5±0.7 | 62.0±0.4 | 77.6±0.8 | 56.2±0.7 |
| vanilla w/ $\mathcal{L}_s$ & $\mathcal{L}_p$ | 86.2±0.8 | 66.1±0.4 | 66.9±0.7 | 81.3±1.1 | 59.4±1.1 |
| vanilla w/ targeted ER | 87.2±0.9 | 71.9±0.9 | 72.4±0.4 | 84.1±0.9 | 58.2±1.3 |
| NCE | **89.9±0.4** | **74.1±0.7** | **75.7±0.4** | **84.3±0.4** | **61.6±0.7** |

# References

Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., and Tuytelaars, T. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 139–154, 2018.

Aljundi, R., Belilovsky, E., Tuytelaars, T., Charlin, L., Caccia, M., Lin, M., and Page-Caccia, L. Online continual learning with maximal interfered retrieval. In *Advances in Neural Information Processing Systems*, volume 32, 2019a.

Aljundi, R., Kelchtermans, K., and Tuytelaars, T. Task-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11254–11263, 2019b.

Aljundi, R., Lin, M., Goujaud, B., and Bengio, Y. Gradient based sample selection for online continual learning. *Advances in neural information processing systems*, 32, 2019c.

Alquier, P. User-friendly introduction to pac-bayes bounds. *arXiv preprint arXiv:2110.11216*, 2021.

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of gibbs posteriors. *The Journal of Machine Learning Research*, 17(1):8374–8414, 2016.

Belouadah, E. and Popescu, A. Il2m: Class incremental learning with dual memory. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 583–592, 2019.

Buzzega, P., Boschini, M., Porrello, A., Abati, D., and Calderara, S. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Caccia, L., Aljundi, R., Asadi, N., Tuytelaars, T., Pineau, J., and Belilovsky, E. New insights on reducing abrupt representation change in online continual learning. *ICLR*, 2022.

Cha, H., Lee, J., and Shin, J. Co2l: Contrastive continual learning. In *Proceedings of the IEEE/CVF International conference on computer vision*, pp. 9516–9525, 2021.

Chaudhry, A., Dokania, P. K., Ajanthan, T., and Torr, P. H. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 532–547, 2018.

Chaudhry, A., Ranzato, M., Rohrbach, M., and Elhoseiny, M. Efficient lifelong learning with a-gem. *ICLR*, 2019a.

Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P. K., Torr, P. H., and Ranzato, M. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019b.

Chaudhry, A., Khan, N., Dokania, P., and Torr, P. Continual learning in low-rank orthogonal subspaces. *Advances in Neural Information Processing Systems*, 33:9900–9911, 2020.

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., and Gao, W. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12299–12310, 2021.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.

Chrysakis, A. and Moens, M.-F. Online continual learning from imbalanced data. In *International Conference on Machine Learning*, pp. 1952–1961. PMLR, 2020.

De Lange, M. and Tuytelaars, T. Continual prototype evolution: Learning online from non-stationary data streams. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8250–8259, 2021.

de Masson D'Autume, C., Ruder, S., Kong, L., and Yogatama, D. Episodic memory in lifelong language learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Dedieu, A., Hazimeh, H., and Mazumder, R. Learning sparse classifiers: Continuous and mixed integer optimization perspectives. *The Journal of Machine Learning Research*, 22(1):6008–6054, 2021.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dhar, P., Singh, R. V., Peng, K.-C., Wu, Z., and Chellappa, R. Learning without memorizing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5138–5146, 2019.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Garg, P., Saluja, R., Balasubramanian, V. N., Arora, C., Subramanian, A., and Jawahar, C. Multi-domain incremental learning for semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 761–771, 2022.

Ghunaim, Y., Bibi, A., Alhamoud, K., Alfarra, M., Al Kader Hammoud, H. A., Prabhu, A., Torr, P. H., and Ghanem, B. Real-time evaluation in online continual learning: A new hope. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11888–11897, 2023.

Gu, Y., Yang, X., Wei, K., and Deng, C. Not just selection, but exploration: Online class-incremental continual learning via dual view consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7442–7451, 2022.

Guo, Y., Liu, M., Yang, T., and Rosing, T. Improved schemes for episodic memory-based lifelong learning. *Advances in Neural Information Processing Systems*, 33: 1023–1035, 2020.

Guo, Y., Hu, W., Zhao, D., and Liu, B. Adaptive orthogonal projection for batch and online continual learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 6783–6791, 2022a.

Guo, Y., Liu, B., and Zhao, D. Online continual learning through mutual information maximization. In *International Conference on Machine Learning*, pp. 8109–8126. PMLR, 2022b.

Haddouche, M. and Guedj, B. Online pac-bayes learning. *Advances in Neural Information Processing Systems*, 35: 25725–25738, 2022.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Helber, P., Bischke, B., Dengel, A., and Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

Hou, S., Pan, X., Loy, C. C., Wang, Z., and Lin, D. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 831–839, 2019.

Jung, M., McKee, S. A., Sudarshan, C., Dropmann, C., Weis, C., and Wehn, N. Driving into the memory wall: The role of memory for advanced driver assistance systems and autonomous driving. In *Proceedings of the International Symposium on Memory Systems*, pp. 377–386, 2018.

Kasarla, T., Burghouts, G., van Spengler, M., van der Pol, E., Cucchiara, R., and Mettes, P. Maximum class separation as inductive bias in one matrix. *Advances in Neural Information Processing Systems*, 35:19553–19566, 2022.

Kim, G., Esmaeilpour, S., Xiao, C., and Liu, B. Continual learning based on ood detection and task masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3856–3866, 2022.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Ko, J., Lu, C., Srivastava, M. B., Stankovic, J. A., Terzis, A., and Welsh, M. Wireless sensor networks for healthcare. *Proceedings of the IEEE*, 98(11):1947–1960, 2010.

Koh, H., Kim, D., Ha, J.-W., and Choi, J. Online continual learning on class incremental blurry task configuration with anytime inference. *ICLR*, 2022.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images, 2009.

Lee, K.-Y., Zhong, Y., and Wang, Y.-X. Do pre-trained models benefit equally in continual learning? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6485–6493, 2023.

Levy, T. and Abramovich, F. Generalization error bounds for multiclass sparse linear classifiers. *Journal of Machine Learning Research*, 24(151):1–35, 2023.

Lin, Z., Shi, J., Pathak, D., and Ramanan, D. The clear benchmark: Continual learning on real-world imagery. In *Thirty-fifth conference on neural information processing systems datasets and benchmarks track (round 2)*, 2021.

Mai, Z., Li, R., Kim, H., and Sanner, S. Supervised contrastive replay: Revisiting the nearest class mean classifier in online class-incremental continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3589–3599, 2021.

Mai, Z., Li, R., Jeong, J., Quispe, D., Kim, H., and Sanner, S. Online continual learning in image classification: An empirical survey. *Neurocomputing*, 469:28–51, 2022.

Mallya, A. and Lazebnik, S. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

Mehta, S. V., Patil, D., Chandar, S., and Strubell, E. An empirical investigation of the role of pre-training in lifelong learning. *Journal of Machine Learning Research*, 24(214):1–50, 2023.

Mirza, M. J., Masana, M., Possegger, H., and Bischof, H. An efficient domain-incremental learning approach to drive in all weather conditions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3001–3011, 2022.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. *Advances in Neural Information Processing Systems Workshop*, 2011.

Oren, G. and Wolf, L. In defense of the learning without forgetting for task incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2209–2218, 2021.

Papyan, V., Han, X., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

Prabhu, A., Torr, P. H., and Dokania, P. K. Gdumb: A simple approach that questions our progress in continual learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pp. 524–540. Springer, 2020.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Rivasplata, O., Kuzborskij, I., Szepesvári, C., and Shawe-Taylor, J. Pac-bayes analysis beyond the usual bounds. *Advances in Neural Information Processing Systems*, 33: 16833–16845, 2020.

Shim, D., Mai, Z., Jeong, J., Sanner, S., Kim, H., and Jang, J. Online class-incremental continual learning with adversarial shapley value. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 9630–9638, 2021.

Sodhani, S., Chandar, S., and Bengio, Y. Toward training recurrent neural networks for lifelong learning. *Neural computation*, 32(1):1–35, 2020.

Sokar, G., Mocanu, D. C., and Pechenizkiy, M. Learning invariant representation for continual learning. *arXiv preprint arXiv:2101.06162*, 2021.

Starr, A., Libertus, M. E., and Brannon, E. M. Number sense in infancy predicts mathematical abilities in childhood. *Proceedings of the National Academy of Sciences*, 110 (45):18116–18120, 2013.

Van de Ven, G. M. and Tolias, A. S. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. Glue: A multi-task benchmark and analysis platform for natural language understanding. *ICLR*, 2019.

Wang, L., Zhang, X., Yang, K., Yu, L., Li, C., Hong, L., Zhang, S., Li, Z., Zhong, Y., and Zhu, J. Memory replay with data compression for continual learning. *ICLR*, 2022.

Wang, L., Zhang, X., Su, H., and Zhu, J. A comprehensive survey of continual learning: Theory, method and application. *arXiv preprint arXiv:2302.00487*, 2023.

Wei, Y., Ye, J., Huang, Z., Zhang, J., and Shan, H. Online prototype learning for online continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18764–18774, 2023.

Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., and Fu, Y. Large scale incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 374–382, 2019.

Yang, J., Shi, R., Wei, D., Liu, Z., Zhao, L., Ke, B., Pfister, H., and Ni, B. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023a.

Yang, Y., Yuan, H., Li, X., Wu, J., Zhang, L., Lin, Z., Torr, P., Tao, D., and Ghanem, B. Neural collapse terminus: A unified solution for class incremental learning and its variants. *arXiv preprint arXiv:2308.01746*, 2023b.

Yoon, J., Yang, E., Lee, J., and Hwang, S. J. Lifelong learning with dynamically expandable networks. *ICLR*, 2018.

You, K., Liu, Y., Wang, J., and Long, M. Logme: Practical assessment of pre-trained models for transfer learning. In *International Conference on Machine Learning*, pp. 12133–12143. PMLR, 2021.

Zenke, F., Poole, B., and Ganguli, S. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pp. 3987–3995. PMLR, 2017.

Zhong, Z., Cui, J., Li, Z., Lo, E., Sun, J., and Jia, J. Rebalanced siamese contrastive mining for long-tailed recognition. *arXiv preprint arXiv:2203.11506*, 2022.

Zhou, D.-W., Wang, Q.-W., Ye, H.-J., and Zhan, D.-C. A model or 603 exemplars: Towards memory-efficient class-incremental learning. *ICLR*, 2023a.

Zhou, D.-W., Ye, H.-J., Zhan, D.-C., and Liu, Z. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *arXiv preprint arXiv:2303.07338*, 2023b.

Zhou, Z., Guo, L.-Z., Jia, L.-H., Zhang, D., and Li, Y.-F. Ods: test-time adaptation in the presence of open-world data shift. In *International Conference on Machine Learning*, pp. 42574–42588. PMLR, 2023c.

Zhou, Z.-H. Stream efficient learning. *arXiv preprint arXiv:2305.02217*, 2023.

Zhu, F., Zhang, X.-Y., Wang, C., Yin, F., and Liu, C.-L. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5871–5880, 2021.

## A. Detailed Proof

We first reintroduce the classical PAC-Bayes adapted from (Alquier, 2021; Alquier et al., 2016) as the Lemma.

**Lemma A.1** (Adapted from (Alquier et al., 2016), Thm 4.1)**.** *Let $\mathcal{D} = (z_1, ..., z_m)$ be an iid set sampled from the law $\mu$. For any data-free prior $P$, for any loss function $\ell$ bounded by $K$, any $\lambda > 0, \delta \in [0, 1]$, one has with probability $1 - \delta$ for any posterior $Q \in \mathcal{M}_1(\mathcal{H})$:*

$$\mathbb{E}_{h\sim Q}\mathbb{E}_{z\sim\mu}[\ell(h, z)] \leq \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}_{h\sim Q}[\ell(h, z_i)] + \frac{\mathrm{KL}(Q\|P) + \log(1/\delta)}{\lambda} + \frac{\lambda K^2}{2m},$$

*where $\mathcal{M}_1(\mathcal{H})$ denotes the set of all probability distributions on $\mathcal{H}$.*

*Remark* A.2. Lemma A.1 is a special case of the original theorem from (Alquier et al., 2016) as we take the case of a bounded loss which implies the subgaussianity of the random variables $\ell(., z_i)$ and then allows us to recover the factor $\frac{\lambda K^2}{m}$.

Following (Rivasplata et al., 2020), we introduce the notion of *stochastic kernel* which formalise properly data-dependent measures within the PAC-Bayes framework. First, for a fixed predictor space $\mathcal{H}$, we set $\Sigma_{\mathcal{H}}$ to be the considered $\sigma$-algebra on $\mathcal{H}$.

**Definition A.3** (Stochastic kernels(Rivasplata et al., 2020))**.** A *stochastic kernel* from $\mathcal{D} = \mathcal{Z}^m$ to $\mathcal{H}$ is defined as a mapping $Q : \mathcal{Z}^m \times \Sigma_{\mathcal{H}} \to [0; 1]$ where

- For any $B \in \Sigma_{\mathcal{H}}$, the function $\mathcal{D} = (z_1, ..., z_m) \mapsto Q(\mathcal{D}, B)$ is measurable,
- For any $\mathcal{D} \in \mathcal{Z}^m$, the function $B \mapsto Q(\mathcal{D}, B)$ is a probability measure over $\mathcal{H}$.

We denote by $\mathtt{Stoch}(\mathcal{D}, \mathcal{H})$ the set of all stochastic kernels from $\mathcal{S}$ to $\mathcal{H}$ and for a fixed $S$, we set $Q_{\mathcal{D}} := Q(\mathcal{D}, .)$ the data-dependent prior associated to the sample $S$ through $Q$.

Following Definition A.3, we provide a formal definition of the **online predictive sequence** as (Haddouche & Guedj, 2022):

**Definition A.4.** A sequence of stochastic kernels $(P_i)_{i=1..m}$ is denoted as an ***online predictive sequence*** if (i) for all $i \geq 1, S \in \mathcal{Z}^m, P_i(\mathcal{D}, .)$ is $\mathcal{F}_{i-1}$ measurable and (ii) for all $i \geq 2$, $P_i(\mathcal{D}, .) \gg P_{i-1}(\mathcal{D}, .)$.

For $P, Q \in \mathcal{M}_1(\mathcal{H})$, the notation $P \ll Q$ indicates that $Q$ is absolutely continuous wrt $P$ (i.e. $Q(A) = 0$ if $P(A) = 0$ for measurable $A \subset \mathcal{H}$). Before giving a detailed proof, let us first reclaim our theorem.

**Theorem A.5.** *For any distributions $\mu_1, ..., \mu_T$ over $\mathcal{Z}$, $\mathcal{D}_t = (z_1^t, ..., z_{m_t}^t)$ be an iid set sampled from $\mu_t$ as the dataset of task t, for any $\lambda > 0$ and any online predictive sequence (used as both priors and posteriors) $(Q_0, Q_1, ..., Q_T)$, the following inequality holds with probability $1 - \delta$:*

$$\sum_{t=1}^{T}\mathbb{E}_{h_t\sim Q_t}\left[\mathbb{E}_{z_t\sim\mu_t}[\ell(h_t, z_t)]\right] \leq \sum_{t=1}^{T}\sum_{j=1}^{m_t}\frac{\mathbb{E}_{h_t\sim Q_t}\left[\ell(h_t, z_j^t)\right]}{m_t} + \frac{\mathrm{KL}(Q_t\|Q_{t-1})}{\lambda} + \frac{\lambda T K^2}{2m_{\min}} + \frac{T\log(T/\delta)}{\lambda}.$$

*Proof.* For each task $t$ in OCL, we can consider $Q_{t-1}$ as a prior since it doesn't depend on the dataset $\mathcal{D}_t$. By applying Lemma A.1, we can have that let $\mathcal{D}_t = (z_1, ..., z_{m_t})$ be an iid set sampled from the law $\mu_t$. For data-free prior $Q_{t-1}$, for any loss function $\ell$ bounded by $K$, any $\lambda > 0, \tilde{\delta} \in [0, 1]$, one has with probability $1 - \tilde{\delta}$ for a data-dependent posterior $Q_t \in \mathcal{M}_1(\mathcal{H})$:

$$\mathbb{E}_{h_t\sim Q_t}\left[\mathbb{E}_{z_t\sim\mu_t}[\ell(h_t, z_t)]\right] \leq \sum_{j=1}^{m_t}\frac{\mathbb{E}_{h_t\sim Q_t}\left[\ell(h_t, z_j^t)\right]}{m_t} + \frac{\mathrm{KL}(Q_t\|Q_{t-1})}{\lambda} + \frac{\lambda K^2}{2m_t} + \frac{\log(\tilde{\delta})}{\lambda}.$$

We then make $\tilde{\delta} = \delta/T$ and take an union bound on all tasks to ensure with probability $1 - \delta$ for any $t \in 1, 2, ...T$:

$$\mathbb{E}_{h_t\sim Q_t}\left[\mathbb{E}_{z_t\sim\mu_t}[\ell(h_t, z_t)]\right] \leq \sum_{j=1}^{m_t}\frac{\mathbb{E}_{h_t\sim Q_t}\left[\ell(h_t, z_j^t)\right]}{m_t} + \frac{\mathrm{KL}(Q_t\|Q_{t-1})}{\lambda} + \frac{\lambda K^2}{2m_t} + \frac{\log(T/\delta)}{\lambda}.$$

Then, by taking a sum on all tasks, we can have the following result with probability $1-\delta$:

$$\sum_{t=1}^{T} \mathbb{E}_{h_t\sim Q_t}\left[\mathbb{E}_{z_t\sim\mu_t}[\ell(h_t,z_t)]\right] \leq \sum_{t=1}^{T}\sum_{j=1}^{m_t}\Big[\frac{\mathbb{E}_{h_t\sim Q_t}\left[\ell(h_t,z_j^t)\right]}{m_t} + \frac{\mathrm{KL}(Q_t\|Q_{t-1})}{\lambda} + \frac{\lambda K^2}{2m_t} + \frac{\log(T/\delta)}{\lambda}\Big].$$

$$\leq \sum_{t=1}^{T}\sum_{j=1}^{m_t}\frac{\mathbb{E}_{h_t\sim Q_t}\left[\ell(h_t,z_j^t)\right]}{m_t} + \frac{\mathrm{KL}(Q_t\|Q_{t-1})}{\lambda} + \frac{\lambda T K^2}{2m_{\min}} + \frac{T\log(T/\delta)}{\lambda}.$$

□

*Remark* A.6. First of all, we need to explain that such a bound is only used to illustrate the generalization risk for each task, not a global risk. Part of the reason is that this makes it easier to account for the problem of model ignorance (excluding problems of myopia and catastrophic forgetting). On the other hand, it is actually extremely challenging to directly establish the relationship between the expected risk of the final posterior distribution $Q_T$ and the empirical losses of the entire training process.

The flexibility of the classical PAC-Bayes bound allows the stochastic kernels $Q_t$ to be either data-dependent distributions or not, as stated in Lemma A.1. In the case of data-dependent distributions, the only available prior we can select in the predictive sequence $Q_t$ is the initial prior distribution $Q_0$. This emphasizes the significance of a good initialization and pre-trained model in achieving favorable results. Furthermore, by examining Theorem A.5, we can observe that the derived bound deteriorates as the number of tasks $T$ increases. This deterioration arises from the growing number of new tasks, which makes online continual learning more challenging. As the model needs to adapt and accommodate an expanding set of tasks, the learning process becomes increasingly complex and prone to performance degradation.

Furthermore, the third term $\frac{\lambda T K^2}{2m_{\min}}$ in the bound emphasizes the significance of model throughput, as it directly impacts $m_{\min}$. Many existing OCL techniques, such as augmentation, knowledge distillation and gradient constraints all increase the training time, consequently reducing the amount of data ($m_{min}$) that the model can process. A lower model throughput not only hampers the practicality of the OCL method but also restricts the model's generalization ability.

Compared to traditional generalization bounds that only consider the final output of an algorithm, the left side of Theorem A.5 evaluates the performance of the model at each time step. This distinction is crucial because in continual learning scenarios, the model's performance should be assessed and monitored throughout the learning process, rather than solely focusing on the final outcome. Indeed, considering the performance of the model at each time step can be seen as a compromise that aligns the generalization gap with a notion of regret. Compared with the regret bound provided in (Haddouche & Guedj, 2022), the deteriorated convergence rate is mainly caused by the fact that at each time step we don't have an access to all the past data to predict the future as the projected Online Gradient Descent (OGD) algorithm. In summary, this is just a simple and natural extension of (Alquier et al., 2016) in the context of OCL. However, we believe that Theorem A.5 already provides some theoretical guidance for the issues that OCL needs to address. In future work, we hope to obtain a more in-depth theoretical analysis specifically for this problem.

# B. Experiments

## B.1. Experiment Setups

**Memory buffers.** Before delving into the specifics experimental results, it is essential to highlight a significant distinction in the utilization of memory buffers between this paper and other works. Traditional methods typically employ a real-time accessible memory buffer, where at each time step $t$, the model receives a mini-batch of data $X \cup X^b$, drawn i.i.d from $\mathcal{D}_t$ and the memory buffer $\mathcal{B}$, respectively. However, in this work, we impose limitations on the number of requests allowed to retrieve data from the memory buffer. Furthermore, we will assess the time and storage overhead incurred by any additional computations, including data augmentation, knowledge distillation, and gradient calculations. For our experiments, we employ three distinct memory sizes along with their corresponding experience replay frequencies, as presented in each respective table. In contrast to existing replay-based methods that sample a small batch of data from the memory buffer at every training iteration, we evaluate the performance of OCL methods under the assumption that they only have access to the memory buffer every 10, 50, 100 or 500 training iterations. This approach allows us to test the methods under various throughput requirements and is more aligned with the off-site storage of data and models in real-world scenarios.

**Datasets.** We use 6 image classification datasets in the evaluation including CIFAR10, CIFAR100, EuroSat, CLEAR10, CLEAR100 and ImageNet(Krizhevsky et al., 2009; Helber et al., 2019; Lin et al., 2021; Deng et al., 2009). CIFAR10 has 10 classes with 40,000 for training and 10,000 for testing. It is split into 5 disjoint tasks with 2 classes per task. CIFAR100 has 100 classes with 40,000 for training and 10,000 for testing. It is split into 20 disjoint tasks with 5 classes per task. EuroSat has 10 classes with 17,799 for training and 7,000 for testing. It is split into 5 disjoint tasks with 2 classes per task. CLEAR10, CLEAR100, two continual image classification benchmark datasets with a natural temporal evolution of visual concepts in the real world that spans a decade (2004-2014). We adopt the ”streaming” protocols for CL that always test on both seen data and data in the (near) future. ImageNet has 1000 classes with 1,281,167 for training and 50,000 for testing. It is split into 200 disjoint tasks with 5 classes per task. All the methods are trained in a supervised manner and tested on seen classes at any given time. In our experiments, we employ a blurry task boundary as suggested by (Koh et al., 2022) instead of the conventional disjoint task boundary to better reflect realistic and practical scenarios. Specifically, in the process of data arrival, there is partial overlap (set at 10%) between the data at the boundaries of different tasks, rather than being completely disjoint.

**Implementation details.** For CIFAR10, CIFAR100, and EuroSat, we utilize the ViT-Tiny as a backbone and ViT-Base (Dosovitskiy et al., 2020) for CLEAR10, CLEAR100 and ImageNet. We train the model with AdamW optimizer for all the datasets and comparing methods. For all the methods compared, we set the same batch size (10) and replay batch size (10) for fair comparisons. We reproduce all baselines in the same environment with their source code and default settings. For the methods requiring a real-time memory buffer to compute some exclusive variables, we ensure the correct calculation of these variables by increasing the frequency of access to the memory while ensuring a same replay frequency. For the pre-trained models used in Table 1, Table 2, Table 3, Table 8 and Table 9, we use the MAE pre-trained initialization (He et al., 2022) for ImageNet and supervised pre-trained initialization for other datasets.

**Compared baselines.** We conducted a comparison of our NCE approach with 13 baselines, as shown in Table 9, consisting of 9 replay-based OCL baselines and 4 offline CL baselines. To ensure a fair comparison, we implemented a vanilla experience replay on the 3 offline CL baselines, running all approaches for one epoch.

**Evaluation metrics.** The traditional metric, Average accuracy ($A_{avg}$), is commonly used in continual learning. However, $A_{avg}$ only provides information about the model's performance at specific moments of task transitions, which may occur only 5 to 10 times in most OCL setups. This temporal sparsity of measurement makes it insufficient to deduce conclusions about the model's any-time inference capability. In this paper, we propose alternative evaluation metrics: Area Under the Curve of Accuracy ($A_{AUC}$) and Last Accuracy. Inspired by the work of (Koh et al., 2022), we measure accuracy more frequently by evaluating it after every $\Delta n$ samples, instead of only at discrete task transitions. This new metric is equivalent to the area under the curve (AUC) of the accuracy-to-# of samples curve for continual learning methods, with $\Delta n = 1$. We refer to it as Area under the curve of accuracy ($A_{AUC}$), calculated as $A_{AUC} = \sum_{i=1}^{t} f(i \cdot \Delta n) \cdot \Delta n$. Additionally, we include Last Accuracy as another evaluation metric. Last Accuracy simply refers to the model's accuracy after it has processed all the data in the data streams.

### B.2. Discussions on Utilization of Pre-trained Models

*Table 4.* Average accuracy of state-of-the-art methods without the pre-trained initialization.

| | CIFAR-10 | | | CIFAR-100 | | | TinyImageNet | | |
|---|---|---|---|---|---|---|---|---|---|
| Method | $M = 0.1k$ | $M = 0.2k$ | $M = 0.5k$ | $M = 0.5k$ | $M = 1k$ | $M = 2k$ | $M = 1k$ | $M = 2k$ | $M = 4k$ |
| iCaRL(Rebuffi et al., 2017) | 31.0±1.2 | 33.9±0.9 | 42.0±0.9 | 12.8±0.4 | 16.5±0.4 | 17.6±0.5 | 5.0±0.3 | 6.6±0.4 | 7.8±0.4 |
| DER++(Buzzega et al., 2020) | 31.5±2.9 | 39.7±2.7 | 50.9±1.8 | 16.0±0.6 | 21.4±0.9 | 23.9±1.0 | 3.7±0.4 | 5.1±0.8 | 6.8±0.6 |
| PASS(Zhu et al., 2021) | 33.7±2.2 | 33.7±2.2 | 33.7±2.2 | 7.5±0.7 | 7.5±0.7 | 7.5±0.7 | 0.5±0.1 | 0.5±0.1 | 0.5±0.1 |
| AGEM(Chaudhry et al., 2019a) | 17.7±0.3 | 17.5±0.3 | 17.5±0.2 | 5.8±0.1 | 5.9±0.1 | 5.8±0.1 | 0.8±0.1 | 0.8±0.1 | 0.8±0.1 |
| GSS(Aljundi et al., 2019c) | 18.4±0.2 | 19.4±0.7 | 25.2±0.9 | 8.1±0.2 | 9.4±0.5 | 10.1±0.8 | 1.1±0.1 | 1.5±0.1 | 2.4±0.4 |
| ER(Chaudhry et al., 2019b) | 19.4±0.6 | 20.9±0.9 | 26.0±1.2 | 8.7±0.3 | 9.9±0.5 | 10.7±0.8 | 1.2±0.1 | 1.5±0.2 | 2.0±0.2 |
| MIR(Aljundi et al., 2019a) | 20.7±0.7 | 23.5±0.8 | 29.9±1.2 | 9.7±0.3 | 11.2±0.4 | 13.0±0.7 | 1.4±0.1 | 1.9±0.2 | 2.9±0.3 |
| GDumb(Prabhu et al., 2020) | 23.3±1.3 | 27.1±0.7 | 34.0±0.8 | 8.2±0.2 | 11.0±0.4 | 15.3±0.3 | 4.6±0.3 | 6.6±0.2 | 10.0±0.3 |
| ASER(Shim et al., 2021) | 20.0±1.0 | 22.8±0.6 | 31.6±1.1 | 11.0±0.3 | 13.5±0.3 | 17.6±0.4 | 2.2±0.1 | 4.2±0.6 | 8.4±0.7 |
| SCR(Mai et al., 2021) | 40.2±1.3 | 48.5±1.5 | 59.1±1.3 | 19.3±0.6 | 26.5±0.5 | 32.7±0.3 | 8.9±0.3 | 14.7±0.3 | 19.5±0.3 |
| CoPE(De Lange & Tuytelaars, 2021) | 33.5±3.2 | 37.3±2.2 | 42.9±3.5 | 11.6±0.7 | 14.6±1.3 | 16.8±0.9 | 2.1±0.3 | 2.3±0.4 | 2.5±0.3 |
| DVC(Gu et al., 2022) | 35.2±1.7 | 41.6±2.7 | 53.8±2.2 | 15.4±0.7 | 20.3±1.0 | 25.2±1.6 | 4.9±0.6 | 7.5±0.5 | 10.9±1.1 |
| OCM(Guo et al., 2022b) | 47.5±1.7 | 59.6±0.4 | 70.1±1.5 | 19.7±0.5 | 27.4±0.3 | 34.4±0.5 | 10.8±0.4 | 15.4±0.4 | 20.9±0.7 |
| OnPro(Wei et al., 2023) | 57.8±1.1 | 65.5±1.0 | 72.6±0.8 | 22.7±0.7 | 30.0±0.4 | 35.9±0.6 | 11.9±0.3 | 16.9±0.4 | 22.1±0.4 |

**Model's ignorance and myopia: new perspectives.** The current performance bottleneck of the OCL method serves as the initial motivation for our study on the application of pre-trained models in OCL. As shown by results in Table 4 from (Wei et al., 2023), even the best methods can only reach about 30% on CIFAR100 and 20% accuracy on TinyImageNet. Although the exploration of these methods may be meaningful to the community, such performance is completely unworthy of discussion for practical problems. Furthermore, when we review previous work from the perspective of **model's ignorance and myopia**, we observe that many techniques originally developed for continuous learning, such as knowledge distillation and dark experience replay, may not be fully applicable to OCL scenarios. In OCL, the model requires more than just relying on past cognition. It needs the flexibility to dynamically adjust and update its features and classification criteria. The improvement in performance of OCL methods often stems from alleviating model ignorance through replaying past data and incorporating augmentation methods. These approaches help the model adapt and refine its representations, reducing the impact of myopia and enabling better performance in OCL settings.

**Inspiration of using pre-trained models.** Inspired by how humans learn quickly and effectively, we realize our ability to recognize new class is typically built upon fundamental cognitive abilities and prior knowledge. In fact, for most intelligent life forms, the process of learning begins with the acquisition of fundamental concepts and knowledge. For instance, humans possess innate abilities for perception and an instinctual drive to seek advantages while avoiding disadvantages. These foundational aspects of learning form the basis upon which more complex cognitive abilities and knowledge are built. We anticipate that pre-trained models, which have demonstrated success across various domains, can play a similar role as fundamental knowledge that enables a high-throughput, high-performance OCL model supporting any-time inference.

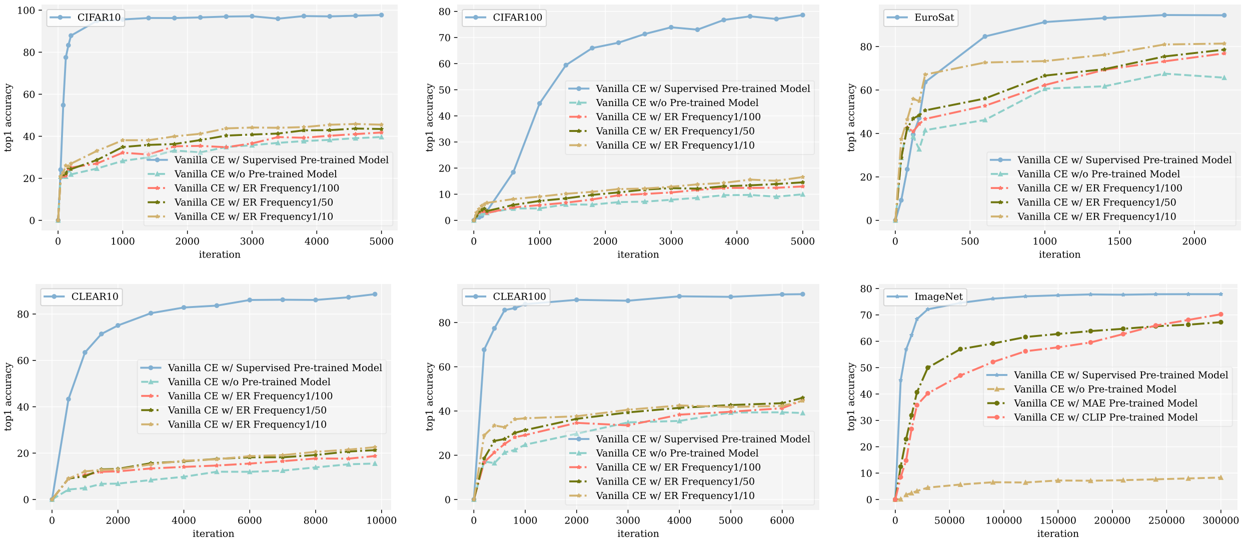

*Figure 5.* We evaluate the real-time accuracy of models on currently seen classes (w/) and (w/o) pre-trained models under our designed **single task setting**, as well as the impact of experience replay frequency on CIFAR, EuroSAT, CLEAR and ImageNet datasets. For CIFAR10, CIFAR100, and EuroSat, we utilize the ViT-Tiny as a backbone and ViT-Base (Dosovitskiy et al., 2020) for CLEAR10, CLEAR100 and ImageNet. More discussions on the effect of different pre-trained models are in Appendix B.2.

*Table 5.* Performance ($A_{AUC}$) under our **single task setting**. We illustrate the impact brought by pre-trained models (models pre-trained on ImageNet by Masked Auto Encoder(He et al., 2022)) with different network architectures over various datasets. The networks that exhibit the best performance with pre-trained models are highlighted in **bold**, while the networks that achieve the best performance without pre-trained models are shown underlined.

| Model | CIFAR10 | CIFAR100 | EuroSat | SVHN | TissueMNIST |
|---|---|---|---|---|---|
| ViT-T w/o pretrain | 31.31 | 9.77 | 55.71 | 54.16 | 43.68 |
| ViT-T w/ pretrain | 93.54 | 61.97 | 76.09 | 88.24 | 59.84 |
| Δ | **+62.23** | +52.20 | +20.38 | +34.08 | +16.16 |
| ViT-S w/o pretrain | 37.64 | 6.95 | 51.81 | 36.86 | 42.78 |
| ViT-S w/ pretrain | **90.38** | **79.49** | **78.02** | **93.33** | **60.04** |
| Δ | +52.74 | **+72.54** | **+26.21** | **+56.47** | **+17.26** |

**Pre-trained model is not a one-size-fits-all solution.** Although pre-trained models have demonstrated significant performance improvements across various datasets (as illustrated in Figure 5), it is crucial to acknowledge that they are not a one-size-fits-all solution. The limitations manifest in multiple aspects, as depicted in Table 5, Table 6 and Table 7, the

*Table 6.* Performance ($A_{AUC}$) under our **single task setting**. We illustrate the impact brought by pre-trained models (models pre-trained on CLIP(Radford et al., 2021)) with different network architectures over various datasets. The networks that exhibit the best performance with pre-trained models are highlighted in **bold**, while the networks that achieve the best performance without pre-trained models are shown underlined.

| Model | CIFAR10 | CIFAR100 | EuroSat | SVHN | TissueMNIST |
|---|---|---|---|---|---|
| Res50 w/o pretrain | 38.58 | 14.04 | 37.64 | 88.04 | 43.72 |
| Res50 w/ pretrain | **86.44** | **79.27** | **65.31** | **92.10** | **60.31** |
| Δ | +47.86 | +65.23 | **+27.67** | +4.06 | **+16.59** |
| ViT-S w/o pretrain | 37.64 | 6.95 | 51.81 | 36.86 | 42.78 |
| ViT-S w/ pretrain | 83.70 | 76.87 | 59.36 | 90.09 | 49.37 |
| Δ | **+55.24** | **+69.92** | +7.55 | **+53.23** | +6.59 |

benefits (or drawbacks) of utilizing a pre-trained model vary depending on the dataset and network architecture employed. Pre-trained models do not consistently yield substantial gains across all datasets. The effectiveness of pre-training depends on several factors, such as the degree of domain similarity between the pre-training and target tasks, the size and quality of the pre-training dataset, some specific characteristics of the target dataset and even the structure of backbone networks matters. For CIFAR, EuroSat, SVHN(Netzer et al., 2011), and TissueMNIST(Yang et al., 2023a), the distributional discrepancy between the pre-training and downstream task data gradually increases. It is evident that pre-trained models tend to provide more benefits when the pre-trained data share common semantics with the downstream tasks. Surprisingly, even for the same dataset and pre-training approach, the choice of model architecture can significantly impact the performance of the model, as observed in Table 5, Table 6 and Table 7 on SVHN and TissueMNIST. These findings highlight the nuanced nature of leveraging pre-trained models. When comparing networks based on convolutional neural networks (CNN), it is evident that transformer-based models tend to derive more benefits from pre-trained models. Furthermore, we also discover that different pre-training methods also exert a significant influence on the model's performance. However, despite conducting these experiments, we have not yet been able to discern definitive rules for selecting pre-trained models. Careful consideration and experimentation are necessary to identify the optimal combination of pre-training and downstream task settings for achieving the desired performance improvements. We believe the selection of appropriate pre-trained models remains an important open question for many areas including OCL.

*Table 7.* Performance ($A_{AUC}$) under our **single task setting**. We illustrate the impact brought by pre-trained models (supervised models pre-trained on ImageNet) with different network architectures over various datasets. The networks that exhibit the best performance with pre-trained models are highlighted in **bold**, while the networks that achieve the best performance without pre-trained models are shown underlined.

| Model | CIFAR10 | CIFAR100 | EuroSat | SVHN | TissueMNIST |
|---|---|---|---|---|---|
| Res18 w/o pretrain | 30.62 | 16.21 | 30.14 | 81.45 | 39.57 |
| Res18 w/ pretrain | 43.60 | 53.41 | 35.68 | 85.60 | 39.40 |
| Δ | +12.98 | +37.20 | +5.54 | +4.15 | -0.17 |
| Res50 w/o pretrain | 38.58 | 14.04 | 37.64 | 88.04 | 43.72 |
| Res50 w/ pretrain | 49.21 | 56.73 | 46.20 | 88.91 | 48.60 |
| Δ | +14.63 | +42.69 | +8.56 | +0.87 | +4.88 |
| WRN28-2 w/o pretrain | 32.10 | 11.17 | 29.52 | 87.69 | 42.38 |
| WRN28-2 w/ pretrain | 46.83 | 37.57 | 36.67 | 87.78 | 39.41 |
| Δ | +14.73 | +26.40 | +7.15 | +0.09 | -2.97 |
| WRN28-8 w/o pretrain | 26.66 | 13.22 | 21.24 | 90.17 | 44.35 |
| WRN28-8 w/ pretrain | 57.13 | 44.23 | 29.91 | 90.93 | 44.05 |
| Δ | +30.47 | +31.01 | +8.67 | +0.76 | -0.30 |
| ViT-T w/o pretrain | 31.31 | 9.77 | 55.71 | 54.16 | 43.68 |
| ViT-T w/ pretrain | 91.49 | 57.55 | 76.40 | 90.07 | 53.35 |
| Δ | **+60.18** | +47.78 | +10.70 | +35.91 | +9.67 |
| ViT-S w/o pretrain | 37.64 | 6.95 | 51.81 | 36.86 | 42.78 |
| ViT-S w/ pretrain | **92.88** | **76.87** | **79.40** | **95.11** | **57.64** |
| Δ | +55.24 | **+69.92** | **+17.59** | **+58.25** | **+14.86** |

## B.3. More Detailed Experimental Results

### B.3.1. Last Accuracy

In addition to $A_{AUC}$, we also evaluate the last accuracy of various OCL methods. We include these two evaluations because $A_{AUC}$ allows us to assess the real-time performance of the model, while the last accuracy measurement reflects the model's

*Table 8. Last Accuracy* on synthetic online class-incremental setting. Best is highlighted in **bold**, second best is shown underlined.

| Method | CIFAR-10 | | | CIFAR-100 | | | EuroSat | | |
|---|---|---|---|---|---|---|---|---|---|
| | $M = 0.1k$ | $M = 0.2k$ | $M = 0.5k$ | $M = 0.5k$ | $M = 1k$ | $M = 2k$ | $M = 0.1k$ | $M = 0.2k$ | $M = 0.5k$ |
| | $Freq = 1/100$ | $Freq = 1/50$ | $Freq = 1/10$ | $Freq = 1/100$ | $Freq = 1/50$ | $Freq = 1/10$ | $Freq = 1/100$ | $Freq = 1/50$ | $Freq = 1/10$ |
| iCaRL(Rebuffi et al., 2017) | 90.1±0.2 | 90.0±0.1 | 92.3±0.3 | 69.6±0.4 | 70.2±0.7 | 73.1±0.2 | 67.7±0.8 | 77.9±1.0 | 87.5±0.4 |
| EWC(Kirkpatrick et al., 2017) | 85.5±0.4 | 92.4±0.8 | 94.2±1.0 | 67.9±0.8 | 66.0±1.1 | 70.4±1.3 | 75.7±1.1 | 84.5±0.9 | 89.2±1.0 |
| DER++(Buzzega et al., 2020) | 87.7±1.4 | 91.1±0.9 | 93.0±1.1 | 67.8±1.7 | 71.5±1.0 | 73.7±0.9 | 66.9±4.3 | 84.7±1.4 | 87.4±2.0 |
| PASS(Zhu et al., 2021) | 91.2±1.1 | 92.0±0.9 | 94.4±0.7 | 69.0±1.4 | 71.7±1.2 | 72.0±0.9 | 70.9±0.8 | 84.5±1.0 | 88.7±1.0 |
| AGEM(Chaudhry et al., 2019a) | 84.3±0.3 | 90.4±0.2 | 93.7±0.9 | 68.2±0.4 | 67.8±0.3 | 73.5±0.1 | 75.6±1.1 | 87.6±0.8 | 91.8±0.7 |
| ER(Chaudhry et al., 2019b) | 91.3±0.7 | 92.0±0.4 | **94.9±0.2** | 73.5±0.6 | 73.3±0.5 | 73.6±0.4 | 76.3±0.8 | 89.8±1.0 | 93.4±0.9 |
| MIR(Aljundi et al., 2019a) | 92.0±1.0 | 92.1±0.8 | 94.1±1.1 | 68.4±0.8 | **74.1±0.8** | **74.7±0.8** | 74.4±2.4 | 88.4±1.6 | 90.0±0.8 |
| ASER(Shim et al., 2021) | 86.2±0.9 | 90.2±1.2 | 93.0±1.1 | 67.9±0.8 | 71.0±0.4 | 72.3±0.8 | 74.3±0.8 | 85.9±0.5 | 92.9±0.8 |
| SCR(Mai et al., 2021) | 89.9±0.6 | 92.2±0.5 | 93.5±1.0 | 69.9±0.9 | 71.7±0.5 | 73.1±0.3 | 75.8±0.8 | 87.8±0.7 | 94.2±1.1 |
| DVC w/o Aug(Gu et al., 2022) | 90.1±0.8 | 91.9±0.9 | 92.7±0.8 | 71.4±0.2 | 71.0±0.6 | 73.5±1.0 | 74.2±4.2 | 85.7±1.0 | 92.3±0.8 |
| DVC w/ Aug(Gu et al., 2022) | 90.6±0.9 | 91.3±0.7 | 94.1±0.4 | 72.7±0.6 | 72.8±0.8 | 73.4±0.6 | 75.1±0.4 | 88.9±0.8 | 92.4±1.2 |
| OCM w/o Aug(Guo et al., 2022b) | 84.1±1.3 | 83.3±1.4 | 92.0±1.1 | 64.2±0.8 | 55.0±1.6 | 64.7±0.4 | 72.8±1.2 | 78.7±1.0 | 80.4±0.6 |
| OCM w/ Aug(Guo et al., 2022b) | 85.6±1.1 | 89.5±1.4 | 93.6±0.5 | **73.7±0.8** | 73.5±1.0 | 73.9±0.5 | 74.1±1.0 | 86.4±1.5 | 93.5±0.7 |
| OnPro w/Aug(Wei et al., 2023) | 92.2±0.9 | 92.1±0.6 | 94.1±0.5 | 69.4±0.5 | 73.7±0.8 | 74.1±0.6 | 76.8±2.4 | 88.6±0.7 | 93.3±1.1 |
| MC-SGD w/ SAM(Mehta et al., 2023) | 90.7±0.6 | 91.5±0.7 | 94.2±0.4 | 70.0±0.8 | 73.1±0.8 | 74.0±1.6 | 75.3±0.4 | 88.9±0.7 | 94.0±1.5 |
| NCE | **93.1±0.5** | **93.0±1.4** | 93.1±1.2 | 70.8±1.5 | 73.9±1.7 | 73.7±1.4 | **84.9±1.7** | **91.7±0.8** | **94.4±0.6** |

*Table 9. Last Accuracy* on real-world online domain-incremental setting and large scale data stream. Best is highlighted in **bold**, second best is shown underlined.

| Method | CLEAR-10 | | CLEAR-100 | | ImageNet |
|---|---|---|---|---|---|
| | $M = 0.1k$ | $M = 0.2k$ | $M = 1k$ | $M = 2k$ | $M = 10k$ |
| | $Freq = 1/100$ | $Freq = 1/50$ | $Freq = 1/100$ | $Freq = 1/50$ | $Freq = 1/500$ |
| ER | 93.4±0.2 | 93.5±0.6 | 88.5±0.9 | 88.1±0.2 | 47.0±0.6 |
| DER | 92.7±0.4 | 93.4±0.8 | 87.9±0.4 | 88.9±0.3 | 43.8±0.7 |
| EWC | 93.0±0.6 | 92.1±0.7 | 89.3±1.4 | **89.6±0.9** | 46.0±0.4 |
| iCarl | 91.4±0.9 | 92.8±1.2 | 84.9±1.3 | 85.9±0.8 | 47.9±1.0 |
| SCR | 89.4±0.6 | 89.8±0.6 | 83.4±0.6 | 87.0±1.1 | 46.3±1.4 |
| OCM | 92.1±0.5 | 92.7±1.3 | 87.9±0.9 | 86.7±0.6 | ××××× |
| DVC | 91.3±0.8 | 91.9±0.4 | 88.0±0.6 | 88.1±0.5 | 45.9±0.5 |
| OnPro | 92.2±1.2 | 93.5±1.6 | 88.0±0.9 | 88.3±0.7 | 47.1±0.6 |
| NCE | **93.9±0.7** | **94.2±1.1** | **90.1±0.8** | 89.2±1.6 | **49.8±2.4** |

performance after processing the entire data stream. As shown in Table 8 and Table 9, Even without employing data augmentation and knowledge distillation, our NCE framework still achieves comparable results. This is particularly evident when faced with more stringent constraints on memory buffer size and replay frequency.

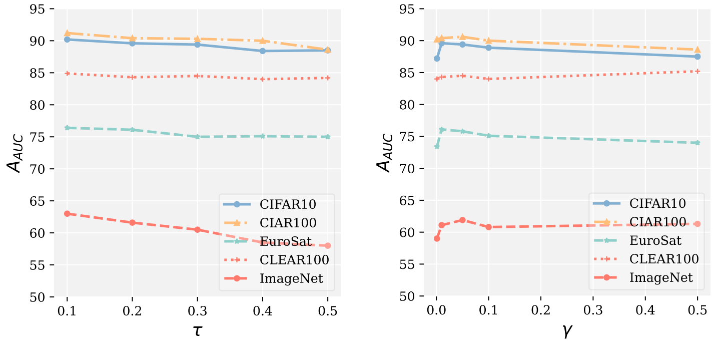

*Figure 6.* Sensitivity analysis on $\tau$ and $\gamma$.

### B.3.2. Sensitivity analysis.

We analyze the impact of the threshold in targeted experience replay and the coefficient on the non-sparse maximum separate regularization. As depicted in Figure 6, we observe that as the threshold $\tau$ increases, the normalized cross-entropy (NCE) has a relatively lower area under the accuracy curve ($A_{AUC}$). This trade-off between performance and efficiency is expected, as higher values of $\tau$ lead to fewer samples being replayed, resulting in improved model throughput but potentially compromising performance.

Furthermore, our approach demonstrates robust outcomes when the coefficient $\gamma$ is not too small, and it basically achieves the best performance when $\gamma = 0.01$.

*Table 10.* Comparison results between our proposed NCE and NCE Lite ($A_{AUC}$). The methods that exhibit the best performance with pre-trained models are highlighted in **bold**.

| Model | CIFAR10 | CIFAR100 | EuroSat | CLEAR10 | CLEAR100 | ImageNet |
|---|---|---|---|---|---|---|
| | $M = 0.1k, Freq = 1/100$ | $M = 0.5k, Freq = 1/100$ | $M = 0.1k, Freq = 1/100$ | $M = 0.1k, Freq = 1/100$ | $M = 1k, Freq = 1/100$ | $M = 10k, Freq = 1/500$ |
| NCE | 89.9±0.4 | **74.1±0.7** | 75.7±0.4 | **89.2±0.7** | **84.3±0.4** | **61.6±0.7** |
| NCE Lite | **90.7±0.6** | 72.9±0.9 | **76.1±0.5** | 88.0±0.6 | 82.9±1.2 | 60.9±0.8 |

**Algorithm 1** NCE (Lite)

```
Input: Data stream $\mathfrak{D}$, encoder $\theta_f$, classifier $\phi$
Initialization: Memory buffer $\mathcal{M} \leftarrow \{\}$,$acc_t = 0$
for $t = 1$ to $T$ do
  for each mini-batch $X$ in $D_t$ do
    $M \leftarrow$Update$(M, X)$
    if $acc_t > threshold$ then
      $\theta_f$.detach()
    end if
    $p = \phi(\theta_f(X))$, $z = \theta_f(X)$
    Compute online class mean $\mu$ and $y^*$ by Equation (3)
    $\theta_f, \theta_\phi \leftarrow \mathcal{L}_{ce} + \gamma(\mathcal{L}_p + \mathcal{L}_s)$ by Equation (7)
    if Replay then
      Compute Confusion Matrix on Memory buffer
      $\theta_f, \theta_\phi \leftarrow \mathcal{L}_b$ by Equation (8)
    end if
    Caculate the accuracy $acc_t$ on current task by $y*$
  end for
end for
```

### B.3.3. NCE Lite.

In addition to enhancing model throughput through constraining memory replay, we also consider the possibility of not fine-tuning the entire network since we have already utilized a pre-trained model. However, when faced with large-scale data streams with changing data distributions, it becomes challenging for the model to adapt to new data without fine-tuning. In such cases, we evaluate the features learned by our model (Equation (3)) during training on the data stream. If the model has acquired sufficiently discriminative features for the current task, we believe that only updating the classifier layer would suffice to achieve optimal results. We refer to this lightweight framework as NCE Lite, detailed in Algorithm 1.

We conducted tests on our lightweight version of NCE using the smallest memory buffer and lowest replay frequency on six datasets, as presented in Table 10. In most cases, this lightweight framework also achieves comparable results with NCE, particularly on relatively simple datasets like CIFAR10 and EuroSat, where NCE Lite even outperforms the original NCE approach. The potential reason for this improvement may be that when NCE Lite encounters simpler datasets, despite constraining the sparseness of the classifier parameters with the dataset, model's myopia caused by intensified parameter sparsity exists not only in the classifier but also in the feature extraction process. This becomes especially apparent when the model acquires highly separable features. Further training may cause the model to excessively focus on discriminative features that lack generalization ability. Hence, introducing a detach operation to the feature extractor $f(\cdot)$ can effectively mitigate the model's myopia, especially when the model has attained satisfactory performance on the current task. By detaching the feature extractor, the model can retain the learned features while allowing for independent updates and adjustments to the classification layer or other components.

## C. Detailed Discussions on Efficiency and Feasibility of Current OCL Methods

In this section, we present empirical observations on the efficiency and feasibility of current OCL methods. We will discuss these observations from several key aspects: requirements on memory buffer, model throughput, and performance. By examining these aspects, we aim to provide insights into the practicality and effectiveness of existing OCL methods in real-world applications.

### C.1. Requirements on Real-time Memory Buffers

As we stated before, OCL serves as a more realistic extension of continual learning, unlike traditional batch learning, where the entire dataset for each task is available upfront, it operates in scenarios where data distributions dynamically change over time. However, we find that, there is very limited research that specifically addresses the accessibility of memory buffers during training in the context of OCL. In this study, we argue that such an assumption is highly unrealistic in a real-world environment. Most existing replay-based OCL methods exhibit some flaws when applied to real-world applications:

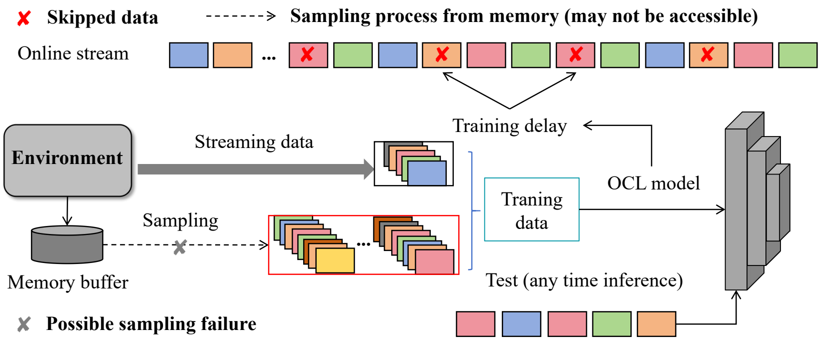

*Figure 7.* Common framework of replay-based OCL methods. Possible sampling failure and training delay due to the mismatch between model training speed and the data stream flow rate are two primary concerns.

(1) As illustrated in Figure 7, a notable characteristic of replay-based methods is their tendency to sample a larger proportion of data from the memory buffer compared to the incoming data stream. However, such a continuous sampling process significantly restricts the throughput of the model for streaming data. It has been observed that the time required to retrain memory data is typically 3-5 times longer than that for new data. Similarly, some data augmentations also significantly reduce the model throughput. This prolonged training time not only results in increased training delays but also leads to more skipped data, reducing the amount of available data that can be effectively utilized within a given time frame, as highlighted in(Ghunaim et al., 2023; Zhou, 2023).

(2) Furthermore, there is a typical assumption that the memory buffer needs to store dozens of samples for each class to help the model to enable efficient review of past classes. However, when dealing with large-scale datasets such as ImageNet (Deng et al., 2009), the storage overhead becomes impractical, particularly for data that needs to be real-time accessible and stored in local memory. Not to mention that in real-world scenarios, the number of categories in the data stream we encounter is constantly increasing. Such limitations become evident when the system lacks continuous access to past data, severely restricting the model's learning capacity, as illustrated in Figure 2. Additionally, most existing methods actually require the memory buffer, model and data being processed to be stored in GPU memory simultaneously to avoid latency during access. This discrepancy between the existing methods and the practical scenario further diminish the practicality of current OCL methods.

(3) In real-world applications, such as autonomous vehicles (Jung et al., 2018) or sensor networks (Ko et al., 2010), ensuring the real-time accessibility of the memory buffer presents a significant challenge, especially when the learning system is deployed in terminal equipment. The constraints of computing resources, privacy concerns and network connectivity all hinder the sampling process in the memory buffer, thereby diminishing the feasibility of existing OCL methods. For example, the latency caused by data transfer makes it difficult for the model to synchronize and obtain data from both the memory buffer and the incoming data stream. Additionally, due to privacy and copyright concerns, in most cases, we cannot store data from the data stream arbitrarily. For instance, in real-world autonomous driving scenarios, data cannot be uploaded to data centers at any time. Meanwhile, the data stored in data centers usually undergoes strict scrutiny to avoid the risk of infringing on privacy or the commercial copyright of another party.

In our research, we simulate the situation where the memory buffer, model and data stream are stored separately by limiting the number of accesses to the memory, which means we cannot replay the desired data at any given moment. Nevertheless, it is important to acknowledge that there may still be a gap between the scenarios we simulated and real-life applications. However, we firmly believe that taking this step is beneficial to the community.

## C.2. Model Throughput and Performance

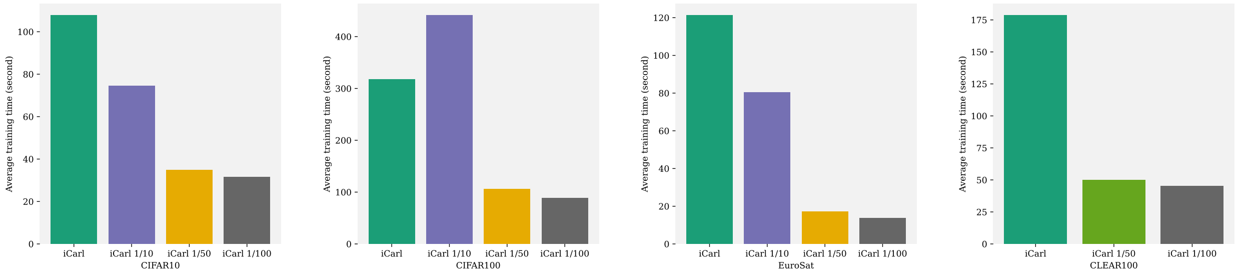

*Figure 8.* Training time of iCarl(Rebuffi et al., 2017) under different replay frequencies across datasets.

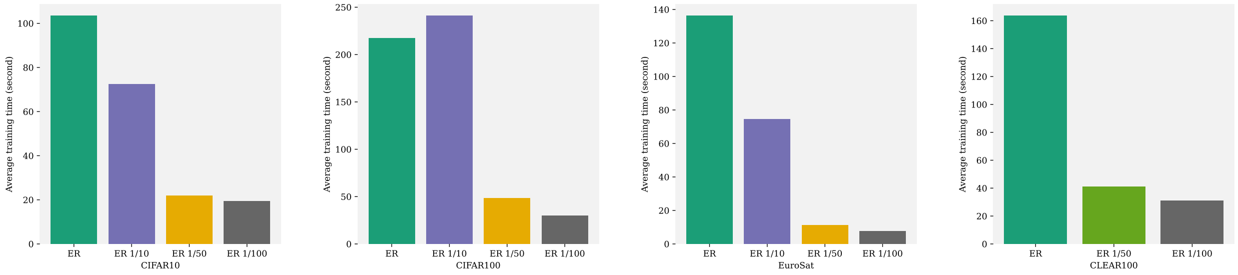

*Figure 9.* Training time of Experience Replay (ER)(Chaudhry et al., 2019b) under different replay frequencies across datasets.

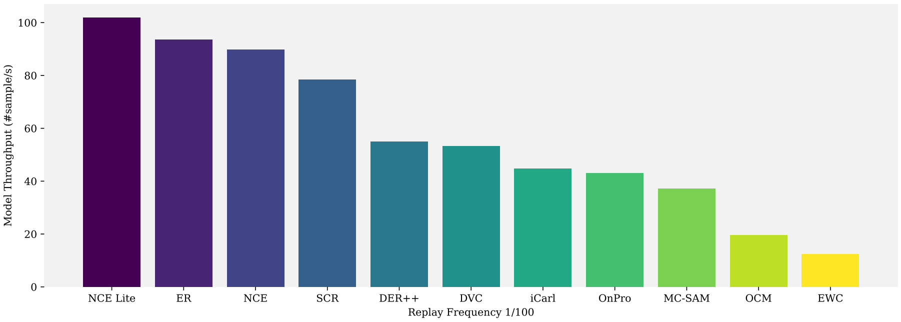

*Figure 10.* Averaged model throughput of 11 OCL methods on 6 datasets (Replay frequency as 1/100).

In recent years, various OCL methods have been proposed (Mai et al., 2022; Guo et al., 2022b; Wei et al., 2023; Wang et al., 2023; Zhou et al., 2023c), among them, replay-based techniques that interleave past experiences with new data have emerged as a predominant component. It aims to consolidate feature learning from earlier tasks while mitigating catastrophic forgetting through the constant re-exposure of few old data. However, the evaluation of OCL methods often overlooks assessment of model throughput, a critical metric especially for data streams with different coming speed. To assess the performance and efficiency of popular OCL methods more effectively, we first record the running time as shown in Figure 8 and Figure 9. It is evident that as the replay frequency increases, the training time of the model (every 200 training iterations) also significantly increases. Consequently, this leads to a substantial reduction in the model's throughput. In comparison, our experimental settings, including frequencies of $1/50$ and $1/100$, considerably shorten the training time and enhance the model's throughput compared to fetching data from the memory buffer for each training iteration. Moreover, we evaluate the averaged model throughput of 11 OCL methods on 6 datasets. As shown in Figure 10, our proposed NCE and NCE Lite improve throughput while ensuring model performance.

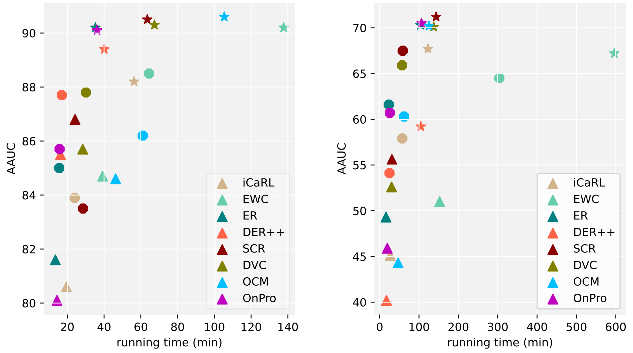

*Figure 11.* $A_{AUC}$ (Area Under the Curve of Accuracy) and running time on CIFAR10 and CIFAR100. ▲,●,★ represents for different replay frequency of $1/100, 1/50, 1/10$.

We also visualize the running time and $A_{AUC}$ of various OCL models with a pre-trained initialization. As shown in Figure 11, existing methods indeed achieved improvements in model performance, but they often come at the cost of slower training speed. Interestingly, as shown in prior work(Ghunaim et al., 2023), if we utilize the available extra time to increase the frequency of replay and train the model multiple times on the memory, the performance gains are often greater compared to using state-of-the-art techniques such as various regularization or knowledge distillation techniques.

### C.3. Conclusion

In addition to efficiency, achieving a balance between model performance, throughput, and practicality is of great concern. The aforementioned practical limitations highlight the necessity for innovative approaches that can adapt to resource-constrained environments and effectively address the challenges in OCL. Considering the difficulties encountered in real-world scenarios, a simple alternative is to limit the frequency of memory access throughout the entire training process. This approach not only improves training throughput but also eliminates the need for real-time storage, thereby alleviating requirements related to hardware specifications, network connectivity, and privacy concerns.

However, this alternative approach introduces new challenges in avoiding model myopia and potential forgetting. While pre-training models can expedite the learning of valuable features, sampling less from memory makes the model to excessively concentrate on the current task, increasing the risk of model myopia and catastrophic forgetting.

## D. Forgetting Phenomenon

We meticulously record the changes in model classification results when using a linear classifier without pre-training initialization, using a linear classifier with pre-training initialization, and using a prototype classifier with pre-training initialization. As demonstrated by Figure 12 and Figure 14, pre-trained initialization allows the model to retain previously learned discriminant information without indiscriminately dividing the data into the current class. In our evaluation, we specifically assess the classifier results of CIFAR10 using linear softmax and the NCM prototype classifier. To gain insights into the model's classification performance, we visualize the classification results on the test set at the beginning and end of each task, as shown in Figure 12. We calculate the model's classification confusion matrix to analyze the results.Our observations reveal the following:

- Pre-trained initialization helps the model rapidly achieve performance on the current task while providing a broader perspective to avoid mindlessly classifying past classes as part of the current task (as the comparison in red and blue in Figure 14).

- The linear softmax classifier demonstrates a quicker acquisition of improved discriminative abilities for data within the current task compared to the NCM classifier (as the comparison in green and blue in Figure 14). However, it is more prone to misclassifying categories from previous tasks as belonging to the current task, resulting in a decline in overall performance.

- When using a pre-trained model and having continuous access to the memory buffer, as illustrated in Figure 12, we can quickly achieve good overall performance.

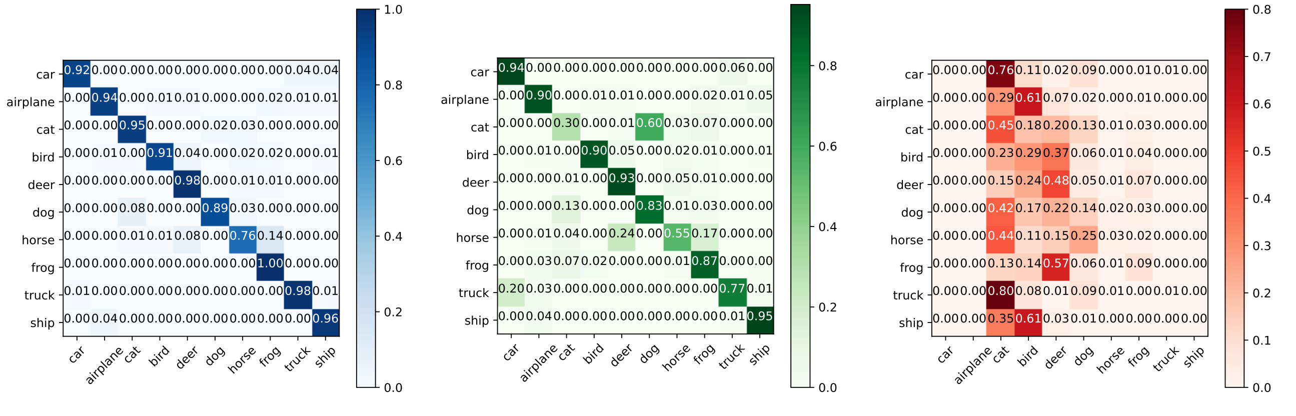

*Figure 12.* Normalized confusion matrix after fine-tuning on a memory buffer with a size of 500.

In fact, both humans and models heavily rely on familiar classes when acquiring features. Similar to how humans may overlook certain features of a specific bird's beak despite being familiar with that bird, these overlooked features could be the distinguishing factor when differentiating it from other unfamiliar birds. Similarly, models have the capability to distinguish between different categories within each individual task. However, when the model's access is limited to only a specific range of categories, its narrow focus on the current task restricts its ability to acquire more comprehensive and discriminative features. It means when familiar features exit in some new categories, confusion may be inevitable.

From our perspective, it is nearly impossible to have a classifier or model that can extract absolute robust features or conclude an absolutely correct classification criterion. Instead, our goal is to adapt the model to learn the necessary knowledge for overall classification, regardless of whether it pertains to seen or new classes.

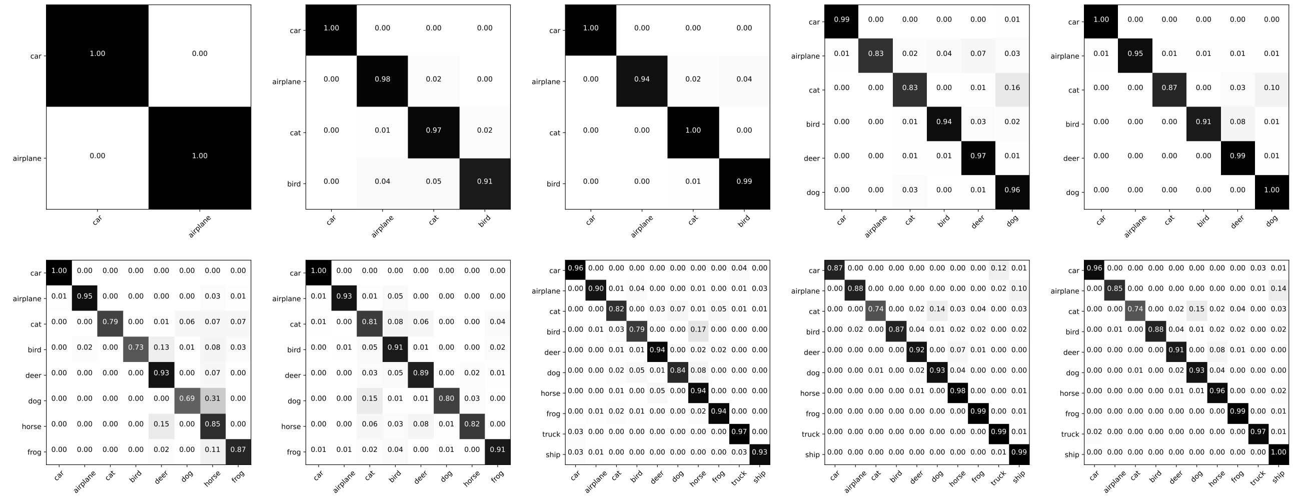

*Figure 13.* The normalized confusion matrix (CIFAR10) evolution of our proposed NCE framework (memory buffer size is 100 and replay frequency is 1/100).

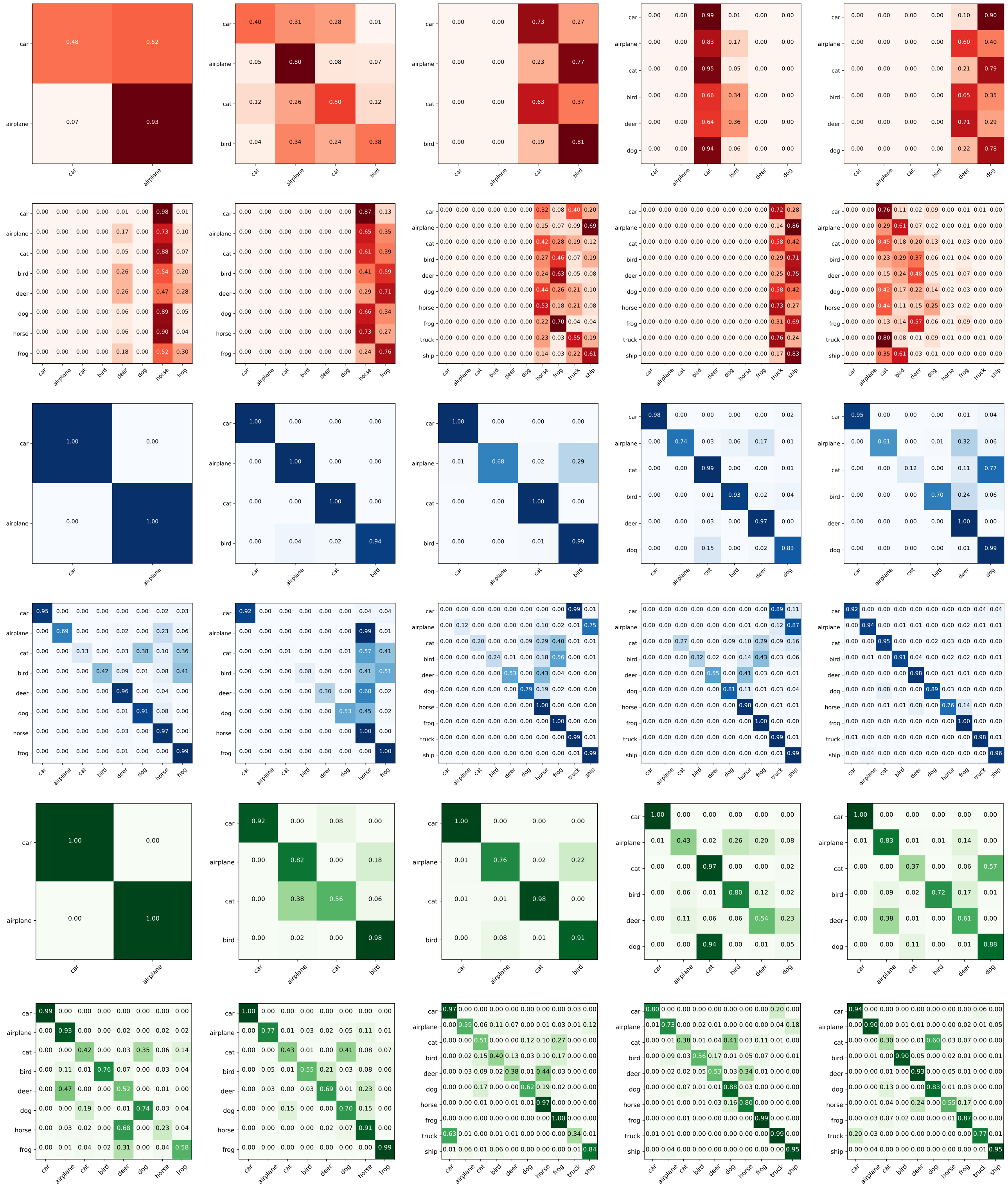

*Figure 14.* The normalized confusion matrix (CIFAR10) evolution of linear softmax classifier without pre-trained initialization (red) and NCM classifier (green), linear softmax classifier (blue) with supervised pre-trained models on ImageNet.