

# Online Continual Learning on a Contaminated Data Stream with Blurry Task Boundaries

Jihwan Bang<sup>1,2</sup> Hyunseo Koh<sup>2,3</sup> Seulki Park<sup>2,4</sup> Hwanjun Song<sup>1,2</sup> Jung-Woo Ha<sup>1,2</sup> Jonghyun Choi<sup>2,5,\*</sup>

<sup>1</sup>NAVER CLOVA <sup>2</sup>NAVER AI Lab <sup>3</sup>GIST <sup>4</sup>Seoul National University <sup>5</sup>Yonsei University

{jihwan.bang,hwanjun.song,jungwoo.ha}@navercorp.com, hyunseo8157@gm.gist.ac.kr,  
seulki.park@snu.ac.kr, jc@yonsei.ac.kr

## Abstract

*Learning under a continuously changing data distribution with incorrect labels is a desirable real-world problem yet challenging. A large body of continual learning (CL) methods, however, assumes data streams with clean labels, and online learning scenarios under noisy data streams are yet underexplored. We consider a more practical CL task setup of an online learning from blurry data stream with corrupted labels, where existing CL methods struggle. To address the task, we first argue the importance of both diversity and purity of examples in the episodic memory of continual learning models. To balance diversity and purity in the episodic memory, we propose a novel strategy to manage and use the memory by a unified approach of label noise aware diverse sampling and robust learning with semi-supervised learning. Our empirical validations on four real-world or synthetic noise datasets (CIFAR10 and 100, mini-WebVision, and Food-101N) exhibit that our method significantly outperforms prior arts in this realistic and challenging continual learning scenario. Code and data splits are available in <https://github.com/clovaai/puridiver>.*

## 1. Introduction

Continual learning (CL) is a practical learning scenario under a continuous and online stream of annotated data [5, 24, 27]. Due to continuously changing data distribution, the CL methods are known to suffer from the stability-plasticity dilemma due to catastrophic interference and resistance to learning new information [12, 25]. In addition, when the CL model is deployed in real-world such as e-commerce applications, the annotated labels are often unreliable due to less controlled data curation process, e.g., crowd-sourcing [36]. Although data labels are likely to be contaminated in real-world due to human errors, existing

CL studies have largely assumed that the given training data has no annotation error [19, 27], which might hinder practical usages of many CL methods in real-world applications.

A very recent work [17] relaxes this assumption by proposing an online CL setup under data stream with less reliable labels. However, they assume a disjoint class incremental scenario such that there is no class overlapping between task streams, which is argued as less practical for real-world applications in the literature [5, 27]. To step forward in addressing continuously changing data distribution that can be falsely labeled, we first design a novel CL task of *online continual learning on a contaminated data stream on blurry task boundaries* [27] as a more realistic and practical continual learning scenario.

Sample selection strategy is arguably important for episodic memory-based CL methods [1, 5, 17, 19, 27]. Specifically, diversity-aware sample selection policy is shown to be effective in the online blurry CL scenarios [5]. However, on the contaminated data stream, diversity-based memory construction may promote to include many examples with corrupted labels in the episodic memory, leading to poor performance thanks to very high capacity of deep models to overly fit all the falsely labeled examples [40].

To address this issue, we propose a unified framework of memory sampling policy with robust learning with the episodic memory and semi-supervised learning with unreliable data, named **PuriDivER** (**P**urity and **D**iversity aware **E**pisode **R**eplay). Fig. 1 illustrates an overview of the proposed method to address the new CL task set-up.

Specifically, for the memory sampling policy, we propose to construct a robust episodic memory that preserves a set of training examples that are *diverse* and *pure*. Unfortunately, maintaining the information diversity and purity is in *trade-off*; clean examples mostly exhibit smaller losses (less diverse) than noise examples due to the memorization effect of deep neural networks [4, 33]. Thus, emphasizing purity in memory sampling does not promote sample diversity and *vice versa*. To address the dilemma, we define a

\*: corresponding author.

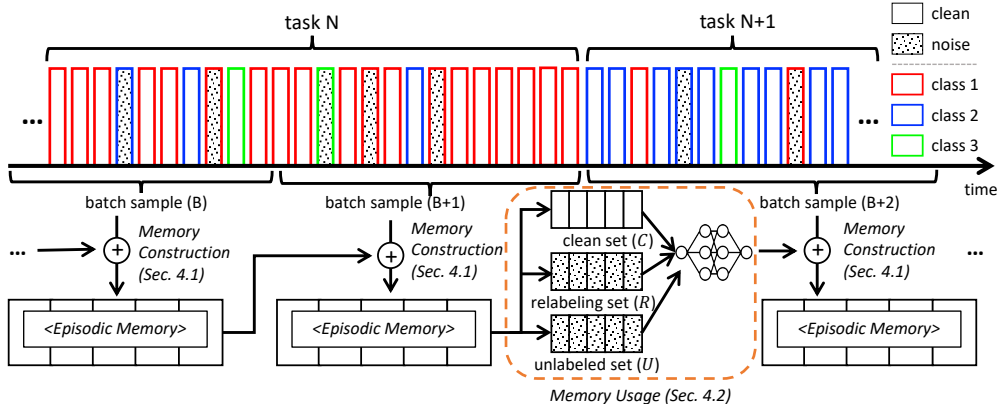


Figure 1. Overview of PuriDivER for the online blurry continual learning with noisy labels, where all the tasks share classes (*i.e.*, blurry tasks [27]) and examples might be falsely labeled. PuriDivER updates episodic memory by balancing examples’ diversity and purity, and unifies a robust training scheme based on semi-supervised learning not to be interfered by unreliable labels in the episodic memory.

score function of promoting the label purity with an additional term to promote diversity by optimizing the sample distribution to be similar to the one with noisy examples. In addition, we incorporate a novel robust learning scheme to further promote both purity and diversity when we use the samples from the episodic memory (Sec. 4.2). Thus, we enforce high diversity and purity around the memory twice; by its construction and usage.

Our empirical validations show that the proposed method outperforms combinations of prior arts to address the combined challenges by noticeable margins on multiple evaluations using CIFAR-10/100, WebVision and Food-101N. Our contributions are summarized as follows:

- Proposing the first online *blurry* CL set-up with contaminated data stream, which is realistic and challenging, and establishing a strong baseline for it.
- Proposing a unified framework of considering diversity and purity in memory updates and usage strategy.
- Proposing an adaptive balancing scheme for dynamic trade-off between diversity and purity in memory sampling for addressing various noise ratio in CL.
- Proposing a semi-supervised robust learning scheme with label cleaned up samples and unreliable samples regarded as unlabeled samples.

## 2. Related Work

There is a large body of literature to address the two challenges of online CL and learning from noisy data, separately. Here, we selectively review the most relevant ones.

### 2.1. Task-free Blurry Online Continual Learning

Recent studies [2, 5, 27] argue that the disjoint CL setup is less realistic but prevalent in literature and propose a new setup where task boundaries share classes, called ‘blurry’. While the disjoint setup assumes each task has non-overlapping classes, the blurry setup allows that all

tasks possibly share the same set of classes with different class distributions. Moreover, we categorize CL methods by the way of using the data; online and offline. Offline setup allows a model to use data many times, while online setup only allows a model to observe a data stream once at the current moment, which is more difficult but practical.

Task-free setup refers to the setup that does not allow the access of task id at inference. There have been many studies on task-free online continual learning (either blurry or disjoint), which are categorized into two; 1) *rehearsal-based approaches* [5, 27, 29], where episodic memory is used to store a few examples of old tasks, and 2) *regularization-based approaches* [8, 19], which regularizes neural network parameters from drastic updates for new information to preserve the information of old tasks for less forgetting. Since rehearsal-based approaches generally outperform the regularization ones, we use it.

For better rehearsal with an episodic memory, many strategies have been proposed for data sampling. Unfortunately, most of them are not suitable for the online continual learning setup except the following few. MER [29] applied *reservoir sampling* for updating memory in online continual learning, and it can approximately optimize over the stationary distribution of all seen examples. GDumb [27] proposed the *greedy balancing sampler* such that it focuses on balancing the number of examples for each class. *Rainbow memory* [5] and OCS [37] are proposed to diversified sampling by regular interval of uncertainty and constructing a ‘Core-Set’ [31], respectively. However, proposed methods assume reliable labels only.

### 2.2. Learning with Unreliable Labels

A number of approaches have been proposed to prevent a model from overfitting to unreliable or noisy labels. One of the typical methods is ‘sample selection’, which trains the model for selected small-loss examples whose labels are

likely to be correct. MentorNet [16] maintains a teacher network to provide the student network with the small-loss examples. Similarly, Co-teaching(+) [14, 38] utilizes two models, where each model selects a certain number of small-loss examples and feeds them to its peer model for training. Alternatively, ‘re-labeling’ corrects noisy labels to reliable ones. SELFIE [32] re-labels the examples by predicted labels’ consistency. SEAL [9] averages the model’s softmax output of each example over the whole training set.

Although they utilize the entire data, when the number of classes or noise rate increases, a risk of overfitting to false labels still persists due to possible false correction. To address this, there have been efforts to combine ‘semi-supervised learning’ with ‘sample selection.’ SELF [26] progressively filters out falsely labeled examples from noisy data by using self-ensemble predictions. DivideMix [7] fits bi-modal univariate Gaussian mixture models (GMMs) to the loss of training examples for sample selection, and then applied the MixMatch algorithm [22]. However, this family of robust methods assumes the offline setup with sufficient amount of training data whereas the training data is limited in the blurry online continual learning setup.

### 2.3. Continual Learning with Unreliable Labels

A straightforward approach for CL with unreliable labels is combining a memory sampling method for online learning with a robust learning method for handling the unreliable labels. However, this naive integration may not resolve the combined challenge because of aforementioned trade-off between diversity and purity of the examples. Recently, SPR [17] is proposed to solve online continual learning with contaminated data via self-purifying scheme. However, the experiments are only conducted on disjoint setup that is less realistic, and there are large rooms to improve in accuracy even in that setup (see Sec. 5 in the supplementary material). In contrast, we approach this problem by constructing the episodic memory with high diversity and purity, and mitigate the potential risk of memory sampling by using a complementary robust learning approach (see Tab. 3) in more realistic CL scenario.

### 3. Task Definition

We design a new online CL setup that is on a contaminated data stream with *blurry* task boundaries following [5]. Let  $C$  and  $T$  be the number of classes and tasks, respectively. At each task  $t$ , the set of classes are split into a set of *major classes*,  $M_t$  and a set of *minor classes*,  $m_t$ . For describing the imbalanced degree of major and minor classes, we define the hyperparameter ( $L$ ) as the number of minor classes over the number of all the classes, and those tasks are called *blurry-L*. Each class could be either a major class once in a certain task or a minor class in the other tasks. Formally, tasks should satisfy the following conditions:

---

#### Algorithm 1 Purity and Diversity aware Episode Replay

---

- 1: **Input:**  $S_t$ : stream data at task  $t$ ,  $\mathcal{M}$ : exemplars stored in a episodic memory,  $T$ : the number of tasks,  $\theta_0$ : initial model.
  - 2: **for**  $t = 1$  **to**  $T$  **do**
  - 3:     **for** each mini-batch  $\mathcal{B} \in S_t$  **do**
  - 4:          $\theta \leftarrow \theta - \alpha \nabla \sum_{i \in \mathcal{B}} \ell(\theta(x_i), \tilde{y}_i)$      ▷ online training
  - 5:         **for**  $x_i, \tilde{y}_i$  in  $\mathcal{B}$  **do**
  - 6:             Update  $\mathcal{M}$  from  $\mathcal{M} \cup (x_i, \tilde{y}_i)$      ▷ Sec. 4.1
  - 7:     **for**  $e = 1$  **to** MaxEpoch **do**
  - 8:         Split  $\mathcal{C}, \mathcal{R}, \mathcal{U}$  from  $\mathcal{M}$  via Eqs. (5) and (7)
  - 9:         **for** each mini-batch  $\mathcal{B}_C \in \mathcal{C}, \mathcal{B}_R \in \mathcal{R}, \mathcal{B}_U \in \mathcal{U}$  **do**
  - 10:              $\theta \leftarrow \theta - \alpha \nabla [\ell_{cls} + \eta \ell_{reg}]$      ▷ Sec. 4.2
- 

$$\begin{aligned} \bigcup_{t=1}^T M_t &= \bigcap_{t=1}^T M_t \cup m_t = \{1, \dots, C\}, \\ \bigcup_{i,j=1, i \neq j}^T M_i \cap M_j &= \emptyset. \end{aligned} \tag{1}$$

To configure blurry tasks, we split the streaming data  $S_t$  for the task  $t$  into subsets of different classes;  $S_t = \{\mathcal{S}_{(t,c)} : c \in \{1, \dots, C\}\}$ . For a given task  $t$ ,  $L = |\cup_{c \in m_t} \mathcal{S}_{(t,c)}| / |S_t|$ , where  $L$  is the specified class imbalance ratio at task  $t$ . In particular, in the presence of noisy labels, the streaming data  $S_t$  for a task  $t$  include falsely labeled examples, thus  $S_t = \{(x_i, \tilde{y}_i)\}_{i=1}^{|S_t|}$  where  $x_i$  is an example and  $\tilde{y}_i$  is its noisy label which may be different from the true (*i.e.*, not contaminated) label  $y_i$ .

### 4. Pure and Diverse Episode Replay

To address the online blurry continual learning with unreliable labels, we propose a method to construct and use the episodic memory. Specifically, we propose (1) a sampling method ‘diverse’ exemplars from the online stream for episodic memory with label purifying scheme (Sec. 4.1), and (2) to use the ‘purified’ diverse examples from the memory by semi-supervised learning with noisy labels for extra diversity with purity (Sec. 4.2). We call this method Purity and Diversity aware Episode Replay (**PuriDivER**). We describe a brief version of algorithm in Alg. 1 and detailed version in the supplementary material (Sec. 1).

#### 4.1. Episodic Memory Construction

Selecting a diverse set of examples for the episodic memory is arguably one of the beneficial steps to improve accuracy of CL model [5]. Since labels are unreliable in our setup, however, this approach would expedite overfitting to falsely labeled examples because they are likely regarded as the ones that are highly diverse thus included in the memory and used in training. On the other hand, to address purity, we may use a small-loss trick for purity aware sampling, but it only selects the examples with low diversity. Thus, the diversity and purity are in trade-off.

To address the dilemma, we define a sampling score function that considers both aspects for the episodic memory, and it is lower the better. Specifically, the proposed

score function consists of two factors; (1) purity of an example by assuming that the ‘small-loss examples’ are likely correctly labeled, and (2) the similarity in representation space between the ‘relevant representation’ of the sample to all other samples with the same noisy label in the memory for diversity. The representation vector is defined as:

**Definition 1** Let  $f(x_i) \in \mathbb{R}^n$  be the representation vector of an example  $x_i$ , where  $f(\cdot)$  be a representation learner. A **relevant representation**  $f_{rel}(x_i; j)$  for a class  $j$  is defined as a subset of elements  $e \in f(x_i)$  such that  $f_{rel}(x_i; j) = \{e \in f(x_i) \mid w(e, j) > \frac{1}{|C|} \sum_{c=1}^C w(e, c)\}$ , where  $w(e, c)$  is a weight parameter of the classification FC layer associated with the element  $e$ , its inputs to the FC layer, and class  $j$ .

Formally, we define and use the score function of an example  $x_i$  to select samples, considering both the purity and diversity with its noisy label  $\tilde{y}_i$  as:

$$S(x_i, \tilde{y}_i) = (1 - \alpha_k) \overbrace{\ell(x_i, \tilde{y}_i)}^{\text{purity}} + \alpha_k \overbrace{\frac{1}{|\mathcal{M}[\tilde{y}_i]|} \sum_{x_i \in \mathcal{M}[\tilde{y}_i]} \cos(f_{rel}(x_i; \tilde{y}_i), f_{rel}(\hat{x}_i; \tilde{y}_i))}_{\text{diversity}}, \quad (2)$$

where  $\mathcal{M}$  is the episodic memory at the current moment,  $k$  is minibatch index to which  $x_i$  is belong and  $\mathcal{M}[\tilde{y}_i]$  is a set of examples annotated as  $\tilde{y}_i$  in the memory. We use the cosine similarity (*i.e.*,  $\cos(\cdot, \cdot)$ ) to measure the similarity between two representations.  $\alpha_k$  is the balancing coefficient of diversity and purity. When the score is high, the sample is more likely corrupted in label and does not contribute to diversity much.

Specifically, the high value of the first term (for purity) implies that  $\tilde{y}_i$  can be a corrupted label with high probability. The high value of the second term (for diversity) implies that the examples annotated with  $\tilde{y}_i$  in the memory are similar to  $x_i$ , which can have less impact to train the model. By the computed score, we update the memory by dropping a single example with the highest score when we add a new sample from the data stream. Finding a proper  $\alpha_k$  is another non trivial issue.

**Adaptive Balancing Coefficient.** We argue that the optimal balancing coefficient  $\alpha_k$  could vary depending on the label noisiness or learning difficulty of each example (Sec. 5.3.3 for empirical analysis). To make the algorithm adaptive to learning difficulty and noisiness of an example, we additionally propose an adaptive  $\alpha_k$ .

As argued in curriculum learning literature [6, 13], easy or clean examples are favored in the early stage of training of a model. Similarly, we conjecture that diverse examples are more important at the later stage of training for better generalization. To implement this idea that implicitly considers learning difficulty and noise ratio, we propose an

adaptive balancing coefficient for a  $k^{th}$  minibatch based on the example’s training loss that is a proxy of label noisiness and learning difficulty (*i.e.*, if loss is large, learning that sample is difficult) as:

$$\alpha_k = 0.5 \cdot \min\left(1/\ell(\mathbf{X}_k, \tilde{\mathbf{Y}}_k), 1\right), \quad (3)$$

where  $\ell(\mathbf{X}_k, \tilde{\mathbf{Y}}_k)$  is the average loss for all examples in the  $k^{th}$  batch. We empirically validate its effectiveness in Sec. 5.3.3.

## 4.2. Episodic Memory Usage

Even with the memory of diverse and expectedly pure samples, the episodic memory may include noisy examples, since the label corrupted examples are mostly one with high diversity. However, due to the small size of the episodic memory, robust learning with the memory does not bring significant accuracy improvement. Here, we propose a hybrid learning scheme that combines re-labeling and unsupervised learning with unreliably labeled examples.

Specifically, we first split the examples in the episodic memory into clean and noisy subsets based on the small-loss trick that is widely used when handling noisy labels [3, 22]. We then fit a GMM to the training loss of all examples in the memory such that it estimates the probability of an example being  $p_G(\cdot)$  [22]. Given a noisy example  $x_i$  with its unreliable label  $\tilde{y}_i$ , we obtain probability of label purity by the posterior probability of GMMs as:

$$p_G(g|\ell(x_i, \tilde{y}_i; \Theta)) = \frac{p_G(\ell(x_i, \tilde{y}_i; \Theta)|g) \cdot p_G(g)}{p_G(\ell(x_i, \tilde{y}_i; \Theta))}, \quad (4)$$

where  $g$  denotes a Gaussian modality for the small-loss examples,  $\Theta$  denotes weight parameters of the model,  $\ell(x_i, \tilde{y}_i; \Theta) = -\log(p_m(x_i, \tilde{y}_i; \Theta))$  and  $p_m(x_i, c)$  is the softmax output of the model for class  $c$ . We finally obtain the clean set  $\mathcal{C}$  and noisy set  $\mathcal{N}$  as:

$$\begin{aligned} \mathcal{C} &:= \{(x_i, \tilde{y}_i) \in \mathcal{M} : p_G(g|\ell(x_i, \tilde{y}_i; \Theta)) \geq 0.5\}, \\ \mathcal{N} &:= \{(x_i, \tilde{y}_i) \in \mathcal{M} : p_G(g|\ell(x_i, \tilde{y}_i; \Theta)) < 0.5\}. \end{aligned} \quad (5)$$

In modern approaches of handling noisy labels [14, 33, 38], only the clean set is used to train the model in a supervised manner, while the noisy set is discarded for robust learning. However, in continual learning, as an episodic memory only contains a few number of samples, we need to utilize as many samples as possible, instead of discarding any. A straightforward way of using them is to purify them.

However, considering that the memory contains samples with various classification confidences (*i.e.*, logit values of a model), if we use the low confidence examples for re-labeling, they are likely incorrectly re-labeled (*i.e.*, purified) due to the perceptual consistency of a neural net. [28].



To address this issue, we further split the noisy set into two subsets; one with high confidence to be re-labeled and the other with low confidence to be used for unsupervised learning. Then, we re-label the former but consider the latter as an unlabeled examples and employ the unsupervised learning; specifically, we use ‘consistency regularization.’

In particular, to split the noise set into the two sets, we define *predictive uncertainty* as:

$$U(x_i) = 1.0 - \max_c (p_m(x_i, c; \Theta)), \quad (6)$$

where  $c \in \{1, \dots, C\}$ . According to our empirical analysis (see Fig. 3-(b)), the uncertainty of the two sets follows *bi-modal* distributions similar to the loss distribution of clean and noise examples. To distinguish two modalities without knowing ground-truth labels, we again fit a GMM to the uncertainty of the (expected) noisy examples in the noise set  $\mathcal{N}$ . Similar to Eq. 5, we split the noise set into the ‘re-labeling set’ ( $\mathcal{R}$ ) and the ‘unlabeled set’ ( $\mathcal{U}$ ) as:

$$\begin{aligned} \mathcal{R} &:= \{(x_i, \tilde{y}_i) \in \mathcal{N} : p_G(u|U(x_i)) \geq 0.5\}, \\ \mathcal{U} &:= \{(x_i, \tilde{y}_i) \in \mathcal{N} : p_G(u|U(x_i)) < 0.5\}, \end{aligned} \quad (7)$$

where  $u$  is the Gaussian component for the low-uncertainty examples. Since the examples with high uncertainty are difficult to estimate the true label, we treat them as the unlabeled set instead to prevent false supervision [30].

**Re-labeling on  $\mathcal{R}$ .** We refurbish the example with low uncertainty (*i.e.*, high confidence) by the convex combination of its given noisy label  $\tilde{y}_i$  and the model’s prediction  $p_m(x_i)$ , whose coefficient is its confidence, *i.e.*, the posterior probability of the mixture component  $u$  as:

$$\hat{y}_i = p_G(u|U(x_i)) \cdot p_m(x_i) + (1.0 - p_G(u|U(x_i))) \cdot \tilde{y}_i. \quad (8)$$

Employing a soft re-labeling approach to progressively refine the given noisy label based on the model’s prediction evolving during training, we mitigate the overfitting to incorrectly modified labels at the beginning of training. Hence, the training loss for the clean set ( $\mathcal{C}$ ) and re-labeling set ( $\mathcal{R}$ ) is formulated with their label supervision as:

$$\ell_{cls} = \frac{1}{|\mathcal{C} \cup \mathcal{R}|} \cdot \left( \sum_{(x_i, \tilde{y}_i) \in \mathcal{C}} \ell(x_i, \tilde{y}_i; \Theta) + \sum_{(x_j, \hat{y}_j) \in \mathcal{R}} \ell(x_j, \hat{y}_j; \Theta) \right). \quad (9)$$

**Semi-supervised Learning with  $\mathcal{U}$ .** We finally learn information from less confident samples in  $\mathcal{U}$ . As their labels are very unreliable, relabeling might not correct them. Thus, we consider them as unlabeled data and use consistency regularization, which is widely used in the semi-supervised learning literature [7, 35]. It helps learn useful knowledge from the examples without their labels by penalizing the prediction difference between the two examples

transformed from the same one as:

$$\ell_{reg} = \frac{1}{|\mathcal{U}|} \sum_{x_i \in \mathcal{U}} \|p_m(s(x_i); \Theta) - p_m(w(x_i); \Theta)\|_2, \quad (10)$$

where  $s(\cdot)$  and  $w(\cdot)$  are strong and weak augmentation functions for an example  $x_i$ . We use AutoAugment [10] as  $s(\cdot)$  and random horizontal flipping as  $w(\cdot)$ .

Thus, our final loss function on the three meticulously designed subsets, namely a clean set  $\mathcal{C}$ , a re-labeled set  $\mathcal{R}$ , an unlabeled set  $\mathcal{U}$ , and  $\mathcal{M} = \mathcal{C} \cup \mathcal{R} \cup \mathcal{U}$ , can be written as:

$$\ell(\mathcal{M}) = \ell_{cls}(\mathcal{C} \cup \mathcal{R}) + \eta \cdot \ell_{reg}(\mathcal{U}). \quad (11)$$

## 5. Experiments

### 5.1. Experimental Set-up

**Datasets.** We empirically validate our PuriDivER on image classification tasks on the blurry CL setups [5] with corrupted labels. We use CIFAR-10, CIFAR-100 [20], a real-world noisy data of large-scale crawled food images; Food-101N [21], and a subset of real-world noisy data consisting of large-scale web images whose classes are same as ImageNet [11]; WebVision [23].

**Noisy Label Set-ups.** Following [14, 22, 32, 38], we inject two types of synthetic noise that is widely used in the literature to emulate noisy label in CIFAR-10/100; namely symmetric (SYM) and asymmetric (ASYM) label noise. Symmetric noise flips the ground-truth (GND) label into other possible labels with equal probability, while asymmetric noise flips the GND into one with high probability.

For thorough evaluations, we adjust the ratio of label noise from 20% to 60%. We use the real label noise in Food-101N and WebVision datasets. According to [34], estimated noise ratio of both datasets are 20.0%. We provide the detailed experimental configurations for online CL with noisy labels in the supplementary material (Sec. 2).

**Metrics.** As a main performance metric, we report the *last* test (or validation) accuracy, which is the most widely used metric in CL literature [8, 14], where ‘‘last’’ refers to the moment when all tasks arrive. To quantitatively evaluate robustness in CL with noisy labels, we propose metrics of ‘purity’ and ‘diversity’ for the episodic memory  $\mathcal{M}$  as:

$$\begin{aligned} \text{Purity} &= \frac{\sum_{(x_i, \tilde{y}_i) \in \mathcal{M}} \mathbb{1}(\tilde{y}_i = y_i)}{|\mathcal{M}|}, \\ \text{Div.} &= \frac{1}{C} \sum_{c=1}^C \frac{2}{|\mathcal{M}_c|(|\mathcal{M}_c| - 1)} \sum_{\substack{x_i, x_j \in \mathcal{M}_c \\ x_i \neq x_j}} \|f(x_i) - f(x_j)\|_2, \end{aligned} \quad (12)$$

where  $f(x_i)$  is the representation vector of an example  $x_i$  by a jointly trained (using all tasks, not in CL setup) model, and  $\mathcal{M}_c$  is a subset of  $\mathcal{M}$  consisting of training examples whose ground-truth label is  $c$ , and  $C$  is the number

Table 1. Last test accuracy on CIFAR-10 (K=500) and CIFAR-100 (K=2,000) with SYM- $\{20\%, 40\%, 60\%\}$  and ASYM- $\{20\%, 40\%\}$ , where  $K$  denotes size of episodic memory.

Methods	CIFAR-10					CIFAR-100				
	Sym.			Asym.		Sym.			Asym.	
	20	40	60	20	40	20	40	60	20	40
RSV [29]	54.5 ± 2.1	39.2 ± 0.9	28.7 ± 0.4	53.6 ± 1.6	40.0 ± 1.2	29.4 ± 0.1	19.3 ± 1.2	10.5 ± 0.4	29.8 ± 0.8	20.3 ± 1.0
+ SELFIE [32]	54.5 ± 1.7	39.2 ± 1.1	28.8 ± 2.9	51.8 ± 4.1	40.4 ± 1.4	28.9 ± 0.4	19.5 ± 1.5	10.5 ± 0.4	29.8 ± 0.9	20.1 ± 1.2
+ Co-teaching [14]	56.1 ± 2.3	39.8 ± 4.4	30.5 ± 3.6	53.5 ± 2.9	38.7 ± 1.3	30.4 ± 0.3	22.0 ± 1.2	13.3 ± 0.3	30.8 ± 1.4	20.3 ± 0.6
+ DivideMix [22]	56.1 ± 1.3	43.7 ± 0.3	35.1 ± 1.9	56.1 ± 0.6	38.9 ± 2.5	30.5 ± 0.7	23.1 ± 1.6	13.1 ± 0.5	30.2 ± 0.5	20.1 ± 0.7
GBS [27]	54.8 ± 1.2	41.8 ± 0.9	27.3 ± 2.1	54.2 ± 1.7	40.4 ± 1.1	28.0 ± 0.2	18.8 ± 1.7	10.2 ± 0.8	29.5 ± 0.9	20.3 ± 0.5
+ SELFIE [32]	55.4 ± 0.8	41.6 ± 0.3	27.8 ± 1.8	51.3 ± 4.7	40.7 ± 0.9	28.0 ± 0.6	18.4 ± 1.9	9.9 ± 0.7	29.6 ± 2.1	10.4 ± 0.3
+ Co-teaching [14]	55.1 ± 0.9	42.7 ± 0.9	31.4 ± 3.6	54.0 ± 0.9	39.5 ± 1.3	29.5 ± 0.7	20.9 ± 1.3	12.8 ± 1.5	29.0 ± 1.6	20.5 ± 1.3
+ DivideMix [22]	57.8 ± 1.9	48.8 ± 1.9	34.3 ± 1.3	57.4 ± 0.6	44.6 ± 5.5	29.6 ± 0.8	21.4 ± 1.1	13.2 ± 0.2	28.7 ± 0.7	19.6 ± 1.1
RM [5]	57.1 ± 0.1	46.5 ± 2.6	33.5 ± 3.0	58.3 ± 2.6	46.2 ± 1.9	31.7 ± 1.3	23.9 ± 0.9	14.2 ± 0.4	32.0 ± 1.0	21.6 ± 1.2
+ SELFIE [32]	56.8 ± 1.3	44.8 ± 0.6	31.8 ± 4.4	57.9 ± 1.5	46.9 ± 1.9	32.1 ± 0.8	22.1 ± 1.0	12.4 ± 0.7	31.7 ± 0.4	20.9 ± 0.9
+ Co-teaching [14]	57.5 ± 1.8	47.6 ± 0.7	35.1 ± 2.0	58.5 ± 1.5	45.9 ± 2.0	31.9 ± 0.9	22.8 ± 0.5	14.4 ± 0.2	32.4 ± 0.4	21.3 ± 1.1
+ DivideMix [22]	61.3 ± 0.8	50.9 ± 3.3	34.9 ± 3.1	60.6 ± 1.7	46.4 ± 5.1	31.2 ± 0.7	23.4 ± 0.9	14.8 ± 0.8	30.6 ± 0.3	21.1 ± 0.6
<b>PuriDivER (ours)</b>	<b>61.3 ± 2.1</b>	<b>59.2 ± 0.3</b>	<b>52.4 ± 2.0</b>	<b>61.6 ± 1.6</b>	<b>47.1 ± 3.2</b>	<b>35.6 ± 0.4</b>	<b>33.4 ± 0.3</b>	<b>28.8 ± 0.6</b>	<b>34.6 ± 0.3</b>	<b>25.7 ± 1.1</b>

of classes. ‘Purity’ measures how many clean examples are correctly included in the memory. ‘Diversity’ measures how much the examples in the memory are spread out in the representation space per class. We describe the trade-off of purity and diversity in the supplementary material (Sec. 4).

**Baselines.** We compare our PuriDivER with the combination of representative memory sampling methods for continual learning and robust learning methods for noisy labels. As summarized in Sec. 2, the memory sampling methods include *reservoir sampling* (RSV) [29], *greedy balancing sampler* (GBS) [27], and *rainbow memory* (RM) [5]. Since RM originally updates memory per every task, we update memory of RM at every batch for online CL scenario.

For the robust learning baselines, we consider three widely used approaches in the literature for addressing noisy labels; a re-labeling approach *SELFIE* [32], a sample selection approach *Co-teaching* [14], and a semi-supervised learning approach *DivideMix* [22]. We compared PuriDivER to them; three memory only sampling methods with three robust learning methods combined plus three using memory sampling only, total of twelve.

**Implementation Details.** We use ResNet [15] family as a backbone for all compared algorithms; ResNet18, ResNet32, ResNet34 and ResNet34 are used for CIFAR-10/100, Food-101N, and WebVision, respectively. For CIFARs, we train ResNet using an initial learning rate of 0.05 with cosine annealing and a batch size of 16 for 256 epochs, following [5]. For Food-101N and WebVision, we use the same training hyper-parameters for CIFARs except the training epochs (256  $\rightarrow$  128). In all experiments, we use 1.0 for  $\eta$  in Eq. 11, and use CutMix [39] and AutoAugment [10] for data augmentation with the memory. For reliable validation, we repeat all the experiments thrice with a single NVIDIA P40 GPU and report the average results.

Table 2. Last validation accuracy on WebVision (K=1,000) and Food-101N (K=2,000), where  $K$  is episodic memory size.

Methods	WebVision	Food-101N
RSV [29]	18.4 ± 0.8	11.6 ± 1.0
RSV+ SELFIE [32]	19.1 ± 0.4	11.1 ± 1.3
RSV+ Co-teaching [14]	16.5 ± 0.3	11.3 ± 1.0
RSV+ DivideMix [22]	11.7 ± 1.1	7.3 ± 0.4
GBS [27]	20.3 ± 2.1	10.5 ± 0.4
GBS + SELFIE [32]	20.0 ± 2.1	10.4 ± 0.3
GBS + Co-teaching [14]	17.8 ± 1.8	10.5 ± 0.2
GBS + DivideMix [22]	14.0 ± 0.6	6.9 ± 0.1
RM [5]	23.1 ± 0.1	11.7 ± 0.2
RM+ SELFIE [32]	21.4 ± 1.1	12.0 ± 0.2
RM+ Co-teaching [14]	21.8 ± 0.8	12.0 ± 0.4
RM+ DivideMix [22]	18.2 ± 0.8	9.2 ± 0.9
<b>PuriDivER (Ours)</b>	<b>25.8 ± 0.7</b>	<b>13.4 ± 0.3</b>

## 5.2. Results

We compare the proposed PuriDivER with twelve baselines on the two CIFARs datasets with varying symmetric and asymmetric noise, and summarize the results in Tab. 1. We also compare methods on real-world noisy datasets; WebVision and Food-101N datasets in Tab. 2.

As shown in the two tables, PuriDivER consistently outperforms all other compared methods. Comparing PuriDivER with other methods, in particular, accuracy increases up to 16% on CIFARs, 11% and 3% on WebVision and Food-101N, respectively. Interestingly, the performance gain becomes large as the noise ratio increases or the difficulty of training data increases from CIFAR-10 to 100.

Noticeably, SELFIE [32] does not improve the accuracy in average when combined with the three memory sampling approaches. We believe that the SELFIE rather produces many false corrections in the online learning setup, since it requires an abundant number of data for re-labeling unlike episodic memory that contains a small number of data.

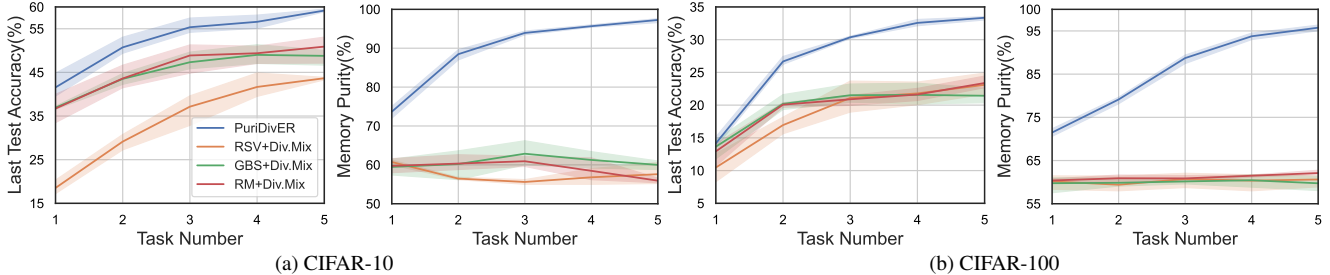


Figure 2. Last test accuracy as a function of memory purity when the task injection progresses on CIFAR-10/100 with SYM-40%. We provide more results with various noise ratio in the supplementary material (Sec. 3).

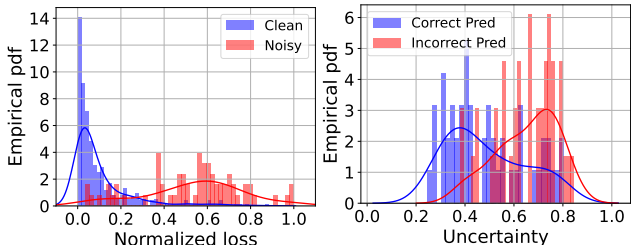


Figure 3. Distribution of examples in the episodic memory training on CIFAR-10 with SYM-20%. GMMs effectively distinguish between (left) clean and noisy labels by loss, and (right) correct and incorrect predictions by uncertainty (Eq. 6)

Despite the use of multiple networks, Co-teaching [14] marginally improves the performance. Unlike SELFIE and Co-teaching, DivideMix considerably improves the accuracy in CIFAR-10, but not in CIFAR-100 with asymmetric noise and real noise datasets, which have more realistic label noise. We expect that the size of (expected) noisy dataset is larger when splitting whole dataset into the clean and the noisy data, as noise ratio becomes larger or learning becomes more difficult. As such, it has too small (expected) clean data to train the model due to the small size of episodic memory. In Tab. 2, we observe that PuriDivER presents a competitive capability on noisy CL tasks in real-world label corruption while outperforming comparable methods.

### 5.3. Detailed Analyses

#### 5.3.1 Purity of Episodic Memory

We first investigate the effect of memory purity on the performance of our method. We measure the last test accuracy as a function of the memory purity (Eq. 12). We compare our PuriDivER with RSV, GBS and RM integrated with DivideMix in Fig. 2. As shown in the figure, ours significantly outperforms them in both accuracy and memory purity on CIFAR-10 and CIFAR-100 by large margins.

Particularly, the performance gap between PuriDivER and other methods gradually increase as more tasks arrive by a synergistic effect of our episodic memory construction (Sec. 4.1) and memory usage schemes (Sec. 4.2). Interestingly, the performance gain of PuriDivER increases

Table 3. Ablation study for PuriDivER on CIFAR-10 and CIFAR-100 with various noise ratios and types. SYM and ASYM refer to the symmetric and asymmetric label noise, respectively.

Robust Learning	Re-label	Consistency	CIFAR-10 (K=500)				CIFAR-100 (K=2,000)					
			SYM		ASYM		SYM		ASYM			
			20	40	60	20	40	20	40	60	20	40
			<b>61.8</b>	55.4	46.9	60.6	46.4	33.1	26.8	18.6	31.5	21.3
✓			57.7	55.4	44.1	61.0	46.6	34.7	31.5	26.2	33.2	22.7
		✓	48.6	46.2	31.3	52.1	36.7	31.7	24.3	18.1	29.9	20.3
✓	✓		61.3	<b>59.2</b>	<b>52.4</b>	<b>61.6</b>	<b>47.1</b>	<b>35.6</b>	<b>33.4</b>	<b>28.8</b>	<b>34.6</b>	<b>25.7</b>

as data becomes more difficult. It implies that the purity of other baselines relatively much worsen in CIFAR-100, while ours maintains its significant dominance. There are similar gains on various noise ratio as shown in the supplementary material (Sec. 3).

#### 5.3.2 Ablations on Robust Learning Components

We now investigate the contribution of each component of the proposed robust learning approach to the accuracy. The both sides of Fig. 3 show the loss distribution of all examples in the memory  $\mathcal{M}$  and the uncertainty distribution (Eq. 7) of the examples in (expected) noisy set  $\mathcal{N}$ , respectively. The loss distributions in Fig. 3-(left) are bi-modal for clean and noisy examples, thus clearly separating them by fitting GMMs. Likewise, the uncertainty distributions in Fig. 3-(right) are bi-modal as well for the correctness of model’s predictions. Hence, re-labeling using the model’s prediction for the left Gaussian components (*i.e.*, certain examples) ensures that the proposed re-labeling method achieves high precision.

Tab. 3 summarizes the contribution of the re-labeling and consistency regularization in PuriDivER. Using either re-labeling or consistency regularization alone does not improve the accuracy in many experiments because re-labeling would make many false correction, and consistency regularization utilizes only a small size of clean set  $\mathcal{C}$ . For those reasons, they also drastically decrease the performance as the noise ratio increases.

In contrast, training the model for the clean and re-labeled set  $\mathcal{C} \cup \mathcal{R}$  with the consistency loss for the unlabeled set  $\mathcal{U}$  significantly improves the accuracy by up to 10.2% compared to the baseline without them. By using the both methods, since a good number of examples in the

Table 4. Last test accuracy over various static  $\alpha$  on CIFAR-10 and CIFAR-100 with SYM-{20%, 40%, 60%} and ASYM-{20%, 40%}.  $K$  is the size of episodic memory.

$\alpha$	CIFAR-10 (K=500)					CIFAR-100 (K=2,000)				
	Sym.			Asym.		Sym.			Asym.	
	20	40	60	20	40	20	40	60	20	40
0.1	57.4	55.1	51.6	58.9	47.4	36.1	33.6	<b>28.1</b>	34.0	<b>25.6</b>
0.3	58.1	55.3	<b>52.6</b>	58.4	<b>49.0</b>	35.5	<b>33.6</b>	27.3	34.6	25.5
0.5	<b>60.5</b>	<b>57.0</b>	47.1	<b>60.4</b>	46.7	<b>37.0</b>	32.3	23.0	<b>36.6</b>	25.5
0.6	29.5	30.3	24.4	36.0	33.7	25.3	18.9	13.5	26.7	18.2
1.0	26.1	19.4	19.3	33.0	28.1	15.2	10.5	7.2	20.7	16.6

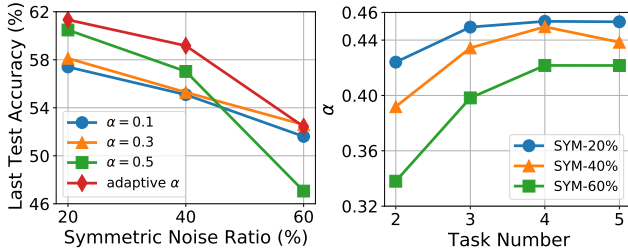


Figure 4. Accuracy by different  $\alpha$  on CIFAR-10 with SYM-{20%, 40%, 60%}; (left) comparison of the last test accuracy with and without the adaptive strategy, (right) comparison of the mean of  $\alpha$  for each task with the adaptive strategy.

re-label set  $\mathcal{R}$  would be correctly labeled, the  $\mathcal{R}$  can also be utilized to train the model better with clean set  $\mathcal{C}$ . Further, as we re-split the three sets  $\mathcal{C}$ ,  $\mathcal{R}$  and  $\mathcal{U}$  at every epoch, it is likely that many data will have both effects of re-labeling and consistency regularization over the epochs, which leads to further improvement.

### 5.3.3 Balancing Coefficient

Tab. 4 illustrates the effect of balancing coefficient  $\alpha$  on the accuracy. As shown in the table, the test accuracy decreases drastically when  $\alpha > 0.5$  because diversity dominantly influences memory sampling rather than memory purity. In the presence of label noise, this suggests that memory purity plays a significant role in performance, as opposed to the conventional CL methods that mainly emphasize example diversity [5]. Hence, we set the range of adaptive balancing coefficient  $\alpha$  to (0, 0.5) instead of (0, 1). Additionally note that the best  $\alpha$  value depends on the noise ratio. Generally, with increased noise ratios, a smaller  $\alpha$  incurs higher accuracy since memory purity is far more important than diversity when label noise is severe.

**Efficacy of Adaptive Strategy.** We compare the performance of PuriDivER with and without using the proposed adaptive balancing coefficient strategy, and summarize the results in Fig. 4-(left). PuriDivER with the adaptive strategy always outperforms others. In addition, as shown in Fig. 4-(right), the  $\alpha$  values are properly determined by PuriDivER with the adaptive strategy for three difference noise scenarios. The  $\alpha$  tends to increase as the task number increases except for the initial task. It implies that small-loss (clean) examples are favored at the earlier stage of training for ro-

bust learning, while diverse examples are favored at the later stage of training for better generalization. Therefore, the proposed adaptive strategy allows PuriDivER to exploit the best  $\alpha$  value for the various noise scenario.

### 5.3.4 Time Cost

In online continual learning, time cost is significant for the model to adapt the distribution change of data stream. Unfortunately, there is a trade-of between time and accuracy. In Fig. 5, RSV, GBS and RM have small amount of training time but poor accuracy, while SPR takes too much time to be utilized in online setup. Despite the additional time cost for memory construction compared to RSV and GBS, PuriDivER exhibits a good trade-off between the accuracy and time cost, compared to others.

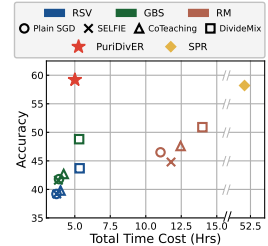


Figure 5. Total training time and accuracy on CIFAR-10 SYM-40% for various methods.

## 6. Conclusion

We address a setup of online and blurry CL with noisy labels, which occurs frequently in real world AI deployment scenarios. Specifically, we propose a method to address this practical task by adaptively balancing diversity and purity by an episodic memory management scheme followed by a robust learning with semi-supervised learning for additional diversity and purity enforcement for the memory usage. We also define an adaptively balancing coefficient to accommodate various noise ratio and sample-wise learning efficiency. We empirically validate that our PuriDivER not only improves accuracy over combinations of prior arts but also helps better sampling for memory.

**Limitation.** Although PuriDivER outperforms prior arts by large margins, developing methods in noisy CL setups are in its nascent steps; *e.g.*, no new class is added as the task progresses. Also, scaled-up empirical validation is important and desirable.

**Potential Negative Societal Impact.** As the CL method is for deploying AI to broader context, the data can come from any sources thus the data privacy issue may arise. Although the proposed method has *no intention* to allow such consequences, the method may carry such negative effects. Efforts to prevent such concerns would be a focus of secure machine learning research.

**Acknowledgement.** All authors thank NAVER Smart Machine Learning (NSML) [18] for experimental environments. This work was partly supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2022R1A2C4002300) and Institute for Information & communications Technology Promotion (IITP) grants funded by the Korea government (MSIT) (No.2020-0-01361-003 and 2019-0-01842, Artificial Intelligence Graduate School Program (Yonsei University, GIST), and No.2021-0-02068 Artificial Intelligence Innovation Hub).



## References

- [1] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. *NeurIPS*, 32:11849–11860, 2019. [1](#)
- [2] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. *NeurIPS*, 32:11816–11825, 2019. [2](#)
- [3] Eric Arazo, Diego Ortego, Paul Albert, Noel E O’Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *ICML*, 2019. [4](#)
- [4] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *ICML*, pages 233–242, 2017. [1](#)
- [5] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *CVPR*, pages 8218–8227, 2021. [1](#), [2](#), [3](#), [5](#), [6](#), [8](#)
- [6] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, pages 41–48, 2009. [4](#)
- [7] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *NeurIPS*, 32, 2019. [3](#), [5](#)
- [8] Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *ECCV*, 2018. [2](#), [5](#)
- [9] Pengfei Chen, Junjie Ye, Guangyong Chen, Jingwei Zhao, and Pheng-Ann Heng. Beyond class-conditional assumption: A primary attempt to combat instance-dependent label noise. In *AAAI*, volume 35, pages 11442–11450, 2021. [3](#)
- [10] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, pages 113–123, 2019. [5](#), [6](#)
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. [5](#)
- [12] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013. [1](#)
- [13] Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *ICML*, pages 1311–1320. PMLR, 2017. [4](#)
- [14] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W Tsang, and Masashi Sugiyama. Co-teaching: robust training of deep neural networks with extremely noisy labels. In *NeurIPS*, pages 8536–8546, 2018. [3](#), [4](#), [5](#), [6](#), [7](#)
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. [6](#)
- [16] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, pages 2304–2313. PMLR, 2018. [3](#)
- [17] Chris Dongjoo Kim, Jinseo Jeong, Sangwoo Moon, and Gunhee Kim. Continual learning on noisy data streams via self-purified replay. In *ICCV*, pages 537–547, 2021. [1](#), [3](#)
- [18] Hanjoo Kim, Minkyu Kim, Dongjoo Seo, Jinwoong Kim, Heungseok Park, Soeun Park, Hyunwoo Jo, KyungHyun Kim, Youngil Yang, Youngkwan Kim, et al. Nsmi: Meet the mlaas platform with a real-world case study. *arXiv preprint arXiv:1810.09957*, 2018. [8](#)
- [19] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *PNAS*, 114(13):3521–3526, 2017. [1](#), [2](#)
- [20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. [5](#)
- [21] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In *CVPR*, pages 5447–5456, 2018. [5](#)
- [22] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *ICLR*, 2019. [3](#), [4](#), [5](#), [6](#)
- [23] Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data. *arXiv preprint arXiv:1708.02862*, 2017. [5](#)
- [24] David Lopez-Paz and Marc’Aurelio Ranzato. Gradient episodic memory for continual learning. *NeurIPS*, 30:6467–6476, 2017. [1](#)
- [25] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. [1](#)
- [26] Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi Phuong Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. Self: Learning to filter noisy labels with self-ensembling. In *ICLR*, 2019. [3](#)
- [27] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *ECCV*, pages 524–540, 2020. [1](#), [2](#), [6](#)
- [28] Scott E Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *ICLRW*, 2015. [4](#)
- [29] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *ICLR*, 2018. [2](#), [6](#)
- [30] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An

- uncertainty-aware pseudo-label selection framework for semi-supervised learning. In *ICLR*, 2020. 5
- [31] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, 2018. 2
- [32] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. SELFIE: Refurbishing unclean samples for robust deep learning. In *ICML*, pages 5907–5915. PMLR, 2019. 3, 5, 6
- [33] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Robust learning by self-transition for handling noisy labels. In *KDD*, 2021. 1, 4
- [34] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022. 5
- [35] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, pages 1195–1204, 2017. 5
- [36] Wenqiang Xu, Liangxiao Jiang, and Chaoqun Li. Improving data and model quality in crowdsourcing using cross-entropy-based noise correction. *Information Sciences*, 546:803–814, 2021. 1
- [37] Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. *arXiv preprint arXiv:2106.01085*, 2021. 2
- [38] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor W Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *ICML*, 2019. 3, 4, 5
- [39] Sangdoon Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, pages 6023–6032, 2019. 6
- [40] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021. 1

# Supplementary Material for Online Continual Learning on a Contaminated Data Stream with Blurry Task Boundaries

Jihwan Bang<sup>1,2</sup> Hyunseo Koh<sup>2,3</sup> Seulki Park<sup>2,4</sup> Hwanjun Song<sup>1,2</sup> Jung-Woo Ha<sup>1,2</sup> Jonghyun Choi<sup>2,5,\*</sup>  
NAVER CLOVA<sup>1</sup> NAVER AI Lab<sup>2</sup> GIST<sup>3</sup> Seoul National Univ.<sup>4</sup> Yonsei Univ.<sup>5</sup>

{jihwan.bang, hwanjun.song, jungwoo.ha}@navercorp.com, hyunseo8157@gm.gist.ac.kr,  
seulki.park@snu.ac.kr, jc@yonsei.ac.kr

**Note:** We use blue color to refer to figures, tables, section numbers and citations in the main paper (e.g., [17]). All red or green characters refer to figures, tables, section numbers and citations in this supplementary material.

## 1. Detailed Algorithm

Algorithm 1 describes overall procedure of PuriDivER. For each task, a model is trained with online data stream  $S_t$  via SGD optimizer (Lines 3–5). Since we can see the data at once by definition of online learning, episodic memory is updated every batch. To get the diversified examples in the memory, we consider the examples of memory and a new example from batch as memory candidates, then remove one that has the maximum value of score function (Lines 7–9). After constructing memory, the model trains with memory. Since the memory might have contaminated data by selecting examples considering both diversity and purity, we split the memory as three parts every epoch; clean set  $\mathcal{C}$ , noisy but high model’s confident set  $\mathcal{R}$ , noisy and low model’s confident set  $\mathcal{U}$  (Lines 11–13). Finally, we apply to the different loss function of each set that is appropriate for handling noisy labels (Lines 14–15).

## 2. Detailed Configuration for Online Continual Learning

To split dataset into several tasks, we follow [5] to make *blurry-CL*. *Blurry-CL* contains two types of classes, major and minor classes. We set  $L = 0.1$ , so the number of minor classes are 10% of the entire of classes at each task. Since some labels might be incorrect, real class distribution of each task might be different. We configure CIFAR-10/100, WebVision and Food-101N as 5, 5, 10 and 5 tasks, respectively. Since Food-101N has 101 classes, the number of major classes is 21 at the last task. Furthermore, we

\*corresponding author

---

### Algorithm 1 Purity and Diversity aware Episode Replay

---

```

1: Input:  $S_t$ : stream data at task  $t$ ,  $\mathcal{M}$ : exemplars stored in a
   episodic memory,  $T$ : the number of tasks,  $\theta_0$ : initial model.
2: for  $t = 1$  to  $T$  do
3:   for each mini-batch  $\mathcal{B} \in S_t$  do
4:     /* Online Training */
5:      $\theta \leftarrow \theta - \alpha \nabla \sum_{i \in \mathcal{B}} \ell(\theta(x_i), \tilde{y}_i)$ 
6:     /* Memory Construction */
7:     for  $x_i, \tilde{y}_i$  in  $\mathcal{B}$  do
8:        $\mathcal{M} \leftarrow \mathcal{M} \cup (x_i, \tilde{y}_i)$ 
9:        $\mathcal{M}.\text{remove}(\arg \max_{(x, \tilde{y}) \in \mathcal{M}} S(x, \tilde{y}))$  by Eq. 2
10:    /* Memory Usage */
11:    for  $e = 1$  to MaxEpoch do
12:      Split  $\mathcal{C}, \mathcal{N}$  from  $\mathcal{M}$  via Eq. 5
13:      Split  $\mathcal{R}, \mathcal{U}$  from  $\mathcal{N}$  via Eq. 7
14:      for each mini-batch  $\mathcal{B}_{\mathcal{C}} \in \mathcal{C}, \mathcal{B}_{\mathcal{R}} \in \mathcal{R}, \mathcal{B}_{\mathcal{U}} \in \mathcal{U}$  do
15:         $\theta \leftarrow \theta - \alpha \nabla [\ell_{cls} + \eta \ell_{reg}]$ 

```

---

consider online-CL which the incoming samples are presented to a model only once except for the examples from the memory.

## 3. Accuracy and memory purity by various noise ratios

We add experimental results of the last accuracy and memory purity of PuriDivER on CIFAR-10/100 with the symmetric noise 20% and 60% in Fig. 1 and 2. We can find consistent results as in Fig. 2. On different scenarios, PuriDivER outperforms the other baselines for both accuracy and memory purity over the entire task stream. Especially, in memory purity, PuriDivER greatly outperforms the other baselines, and it shows that our method can effectively find clean labels from corrupted data for the online continual learning setup. Since we believe that memory sampling and robust learning from PuriDivER are the relationship to help each other, so the last accuracy and mem-

Table 1. Comparison of our method and SPR in disjoint setup.

Methods	CIFAR-10					WebVision
	Sym.			Asym.		Real
	20	40	60	20	40	
Self-Centered filter [17]	36.5	35.7	32.5	37.1	36.9	33.0
Self-Replay [17]	40.1	31.4	22.4	44.1	43.2	48.0
SPR [17]	43.9	43.0	40.0	44.5	43.9	40.0
<b>PuriDivER (Ours)</b>	<b>61.2</b>	<b>60.9</b>	<b>56.0</b>	<b>62.4</b>	<b>46.4</b>	<b>51.8</b>

ory purity increases and have larger gap as the task number increases.

#### 4. Diversity vs. Purity

To analyze how diversity and purity in a memory change according to coefficient  $\alpha$ , we plot the diversity and purity metrics (defined in Eq. 12) in Fig. 3 and 4 for various noise ratios. As hyper-parameter  $\alpha$  increases, the diversity score increases while the purity score decreases in all figures. When  $\alpha$  is increased beyond a certain value, it can be seen that purity is sacrificed for the sample diversity. If there are too many noisy labels in a memory, a trained model would inevitably learn the noisy labels, which can lead to performance degradation. Therefore, the best strategy is to set the  $\alpha$  before the abrupt decrease in the purity score. For example, in Fig. 3a and 3b, memory purity is over 95% until  $\alpha = 0.5$ , and then falls to less than 50% when  $\alpha$  is set to 0.6. Thus, the best strategy for balancing the diversity and memory purity can be obtained at  $\alpha = 0.5$ . Meanwhile, in the case of Fig. 3c, memory purity drops to 70% at  $\alpha = 0.5$ , so it can be expected that the best performance is obtained at  $\alpha = 0.3$ , before memory purity is too much degraded. This is because memory purity becomes more important as the ratio of noisy labels contained in memory increases. These results are consistent with the results in Tab. 4.

#### 5. Comparison to SPR in Disjoint Tasks

As SPR [17] is proposed for similar task setup, we compare our PuriDivER to SPR [17] in the disjoint setup for which the SPR is proposed, and summarize the results in Tab. 1. Following their experimental setup, we configure five tasks with randomly paired classes for CIFAR-10, and use the top 14 largest classes in seven tasks with randomly paired classes for WebVision.

SPR uses two different memories; a delayed buffer, which the incoming data is saved temporarily, and a purified buffer which maintains purified data, and a fixed memory size of 500 or 1,000 for CIFAR-10 or WebVision, respectively. As the delayed buffer is only utilized to auxiliary model (called Self-Replay Model) to use only purified examples for model training, their actual memory size is  $2 \times$  of the episodic memory. Hence, we set memory size as 1,000 and 2,000 in CIFAR-10 and WebVision for fair comparison.

Interestingly, PuriDivER outperforms SPR by 3%-18% in all experiments (Tab. 1). We believe it is because the SPR always purifies incoming contaminated data stream before training the model, but it has a high risk of overfitting by false model predictions.



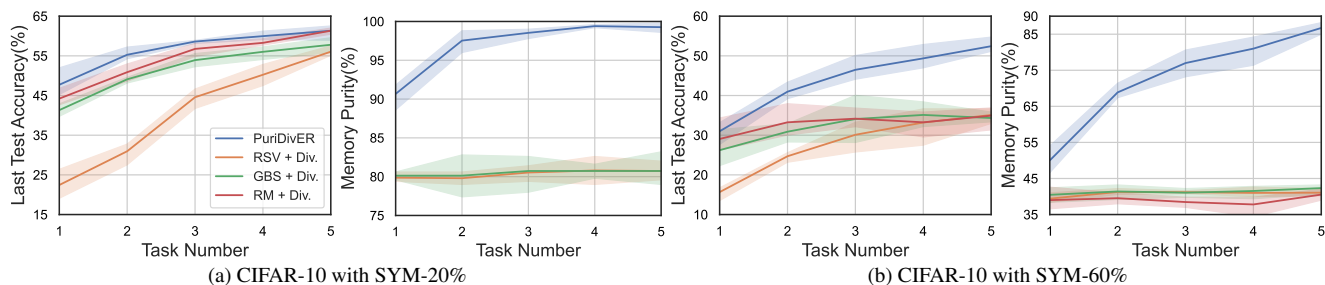


Figure 1. Illustration of last accuracy and memory purity changes as the task number increases on CIFAR-10 with SYM- $\{20\%, 60\%$ .

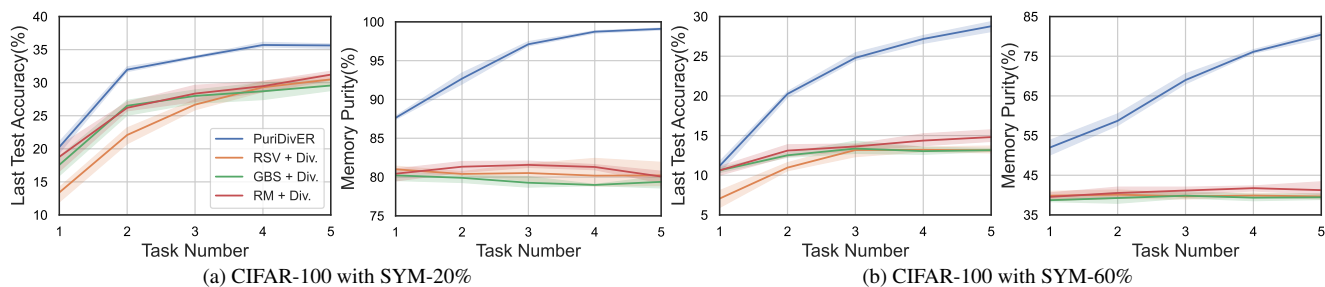


Figure 2. Illustration of last accuracy and memory purity changes as the task number increases on CIFAR-100 with SYM- $\{20\%, 60\%$ .

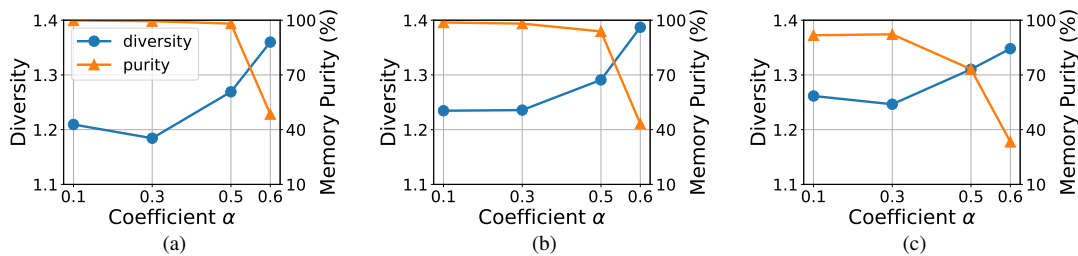


Figure 3. Illustration of diversity and purity in the memory as the coefficient  $\alpha$  changes on CIFAR-10 with (a) SYM-20%, (b) SYM-40%, and (c) SYM-60%.

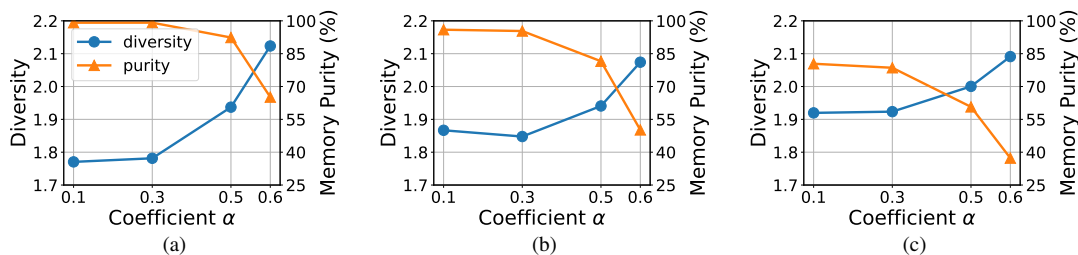


Figure 4. Illustration of diversity and purity in the memory as the coefficient  $\alpha$  changes on CIFAR-100 with (a) SYM-20%, (b) SYM-40%, and (c) SYM-60%.