A study of the Dream Net model robustness across continual learning scenarios

Marion Mainsant, Martial Mermillod, Christelle Godin, Marina Reyboz

A study of the Dream Net model robustness across continual learning scenarios

Marion MAINSANT
Univ. Grenoble Alpes, CEA, List, F-38000 Grenoble, France
marion.mainsant@cea.fr

Martial MERMILLOD
LPNC, Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, 38000, Grenoble, France
martial.mermillod@univ-grenoble-alpes.fr

Christelle GODIN
Univ. Grenoble Alpes, CEA, Leti, F-38000 Grenoble, France
christelle.godin@cea.fr

Marina REYBOZ
Univ. Grenoble Alpes, CEA, List, F-38000 Grenoble, France
marina.reyboz@cea.fr

***Abstract*—Continual learning is one of the major challenges of deep learning. For decades, many studies have proposed efficient models overcoming catastrophic forgetting when learning new data. However, as they were focused on providing the best reduce-forgetting performance, studies have moved away from real-life applications where algorithms need to adapt to changing environments and perform, no matter the type of data arrival. Therefore, there is a growing need to define new scenarios to assess the robustness of existing methods with those challenges in mind. The issue of data availability during training is another essential point in the development of solid continual learning algorithms. Depending on the streaming formulation, the model needs in the more extreme scenarios to be able to adapt to new data as soon as it arrives and without the possibility to review it afterwards. In this study, we propose a review of existing continual learning scenarios and their associated terms. Those existing terms and definitions are synthesized in an atlas in order to provide a better overview. Based on two of the main categories defined in the atlas, "Class-IL" and "Domain-IL", we define eight different scenarios with data streams of varying complexity that allow to test the models robustness in changing data arrival scenarios. We choose to evaluate Dream Net - Data Free, a privacy-preserving continual learning algorithm, in each proposed scenario and demonstrate that this model is robust enough to succeed in every proposed scenario, regardless of how the data is presented. We also show that it is competitive with other continual learning literature algorithms that are not privacy preserving which is a clear advantage for real-life human-centered applications.**

***Index Terms*—continual learning, incremental learning, real-life scenarios, online learning, streaming learning, pseudo-rehearsal, replay, privacy**

I. Introduction

With advances in artificial intelligence, and in deep learning algorithms, the tools developed in those fields are becoming more and more attractive for real-life applications where data monitoring is essential (healthcare, autonomous driving, environment...). Nowadays, many smart devices are developed in order to monitor a wide array of data generated daily by the users. Traditional deep learning usually proposes to train models offline with unlimited quantity of stored data that is used in an independent and identically distributed (i.i.d.) and stationary fashion during the training phase. This setting is unusable for smart devices in a changing environment where data is not available at all time, notably because of its memory footprint, power consumption and privacy-preserving issues. This particular issue can be illustrated with the case of clinical applications where data storage is not always allowed because of patient's privacy concerns [1]. In addition, for those kind of applications, the i.i.d. assumption is not easy to maintain even for data coming from the same patient because of the number of sensors used, temporal changes in physiological data and other factors due to the recording environment [2]. In order to deal with those continuous data arrival, algorithms must continuously adapt while minimizing data storage and complexity.

However, unlike the human ability to keep and fine-tune concepts continuously, Artificial Neural Networks (ANNs) encounter the issue of catastrophic forgetting [3]. The catastrophic forgetting effect appears when the ANN is trained on new data and adapts its parameters in order to match with this new information without taking into consideration previously learned knowledge. Many studies have aimed to propose continual learning models that overcome this issue [4]. Some of the models proposed in those studies were designed as an answer to very specific scenarios of data arrival. In order to highlight differences between scenarios, [5] made an interesting comparative study on continual learning formulations used in literature (see section II for more details). They highlighted the fact that algorithms in the literature were not necessarily comparable in terms of accuracy and data treatment because the implemented scenarios could have drastic differences. For example, some formulations of continual learning need an oracle to give information about the task being performed during the test phase while others do not rely on it. This study will provide an overview of existing continual learning scenarios while clarifying terms and definitions in a global atlas. Besides the discrepancies between the existing continual learning scenarios, the issue of data availability during training is also fundamental. In fact, depending on the algorithm and its training environment, data can be treated in small batches or even one element at the time. This kind of setting, called online learning or streaming learning consists of dealing with

This work was done within the scope of the Carnot MIEL (Multimodal and Incremental Embedded Learning) project. It has been partially supported by MIAI@Grenoble Alpes, (ANR-19-P3IA-0003)

a restrictive quantity of data available at each learning step while not allowing the algorithm to review the same data several times [6]–[9]. Three formulations of streaming learning are presented in this study in order to explore different level of constraints in the data stream. The ability of a model to deal with a wide variety of data arrival is essential when real-life application are considered. For example, smart sensors deployed in a changing environment must be able to learn new data no matter the arrival scenario and stream constraints. There is a lack of homogeneity in continual learning model evaluation in literature, more particularly with regard to data arrival scenarios. That is why, after proposing an overview of continual learning scenarios and streaming formulations, we propose a general framework with eight different training scenarios in order to test and assess the robustness of different continual learning models. A promising privacy preserving continual learning model, Dream Net - Data free [10], will be tested in those eight configurations and compared to baseline and other literature models. In summary, this paper brings the following contributions:

- An overview of existing continual learning scenarios and models with an atlas that enables to synthesize terms and definitions.
- A framework of eight scenarios that allow to efficiently test the robustness of continual learning models over data arrival possibilities.
- A validation of Dream Net - Data Free [10] robustness in the proposed framework compared to other baselines and literature models.

The next section, provides an overview of different continual learning scenarios, streaming formulations, and methods while defining terms in order to better formulate our study settings. The experimental setup introducing the different test scenarios and considered databases is established in section III. In section IV we present the results of experiments on Dream Net - Data free model and discuss them. We finally conclude and propose some perspectives in section V.

II. Literature overview

This section presents the main challenges of continual learning for streaming data. On one side, during the last decades a lot of studies have been done on continual learning with the key challenge of avoiding catastrophic forgetting and a lot of different approaches have been explored [4]. On the other side, online or streaming learning is an emerging research field that challenges the types of data arrivals in order to propose algorithms able to learn on restrained continual data streams. We propose a taxonomy of the different concepts that will highlight one of the major issue of real-life application: the need for algorithms stable and agnostic to data arrival scenario.

A. Continual learning scenarios and streaming formulations

To begin, in continual learning literature, terms like incremental learning or lifelong learning are sometimes also used to talk about similar concepts. Some articles propose different definition for those three terms [11]. Here for more simplicity in the concept definition, we chose to consider them as synonyms. Beyond a set of methods for overcoming catastrophic forgetting, continual learning aims to give ANNs the ability to deal with a non stationary and never-ending stream of data [8]. This ability to adapt in a constantly changing environment is challenging because algorithms have to not only preserve acquired knowledge through past data but also adapt to a potential distribution shift. We provide here an overview of the existing terms and definitions.

1) Task-incremental learning vs Domain-incremental learning vs Class-incremental learning: In order to evaluate continual learning methods, several teams agreed on the definition of three evaluation protocols [12], [13]. In all this sub-section, the term ”task” refers to the process of splitting a given set of data. Each task can contain one or more classes depending on the type of scenario.

Task-incremental learning (Task-IL) is the easiest continual learning scenario. It consists of using an oracle that gives information about the considered task during training and testing phases. For example, when one trains a model with tasks that each contains two classes, during the test phase, the model has information about the considered task for the accuracy measurement. The model only has to choose between two classes and not between all the classes that have been learned so far. With this kind of protocol, ”multi-head” networks are usually used and are built with different outputs for each specific task.

For **Domain-incremental learning (Domain-IL)** approaches, information about the task identity is no longer available during the testing phase. However, the model does not takes into account changes in available labels. It only solves the task for which it has been designed and does not need to find out to which task the example belongs. This scenario is very useful in the case where classes represented in each task are always the same and where new instances of those classes appears.

Class-incremental learning (Class-IL) also works with the condition to not give the task identity during the testing phase. But unlike domain-incremental learning, the algorithm should determine task identity of the infered example among all labels available in tasks learned so far. This approach is considered as one of the most difficult one.

At the intersection between Domain-IL and Class-IL [14] proposed a scenario named **New Instances and Classes** (NIC). In this scenario, data of a new task belongs to both known and new classes.

As explained in [15] work, each of the four above approaches are placed in a framework where tasks are identified during the training phase and boundaries between them are well defined at all time, i.e. the algorithm has the information that current data belongs or not to a new task. From the definition of these three approaches they suggest a fifth approach named **Task-agnostic incremental learning** (Task-agnostic IL) which consists in not having any information about the task identity during the training (discrete task-agnostic) and even about task boundaries (continuous task-agnostic). This

approach is the most restrictive but also the closest to real-life applications and to the functioning of the human brain which is able to learn new tasks in a completely agnostic manner (i.e. without any prior information about the task). However, we will not deal with this approach in this article as it requires to mix traditional continual learning architecture with novelty detection ones [15], [16].
Beyond the type of scenario to be used to process the data, it is important to take into account how the data will evolve over time, i.e. **data drift possibilities**. In classic online deep-learning studies, neural networks require data independent and identically distributed (i.i.d) and come from stationary distributions. In continual learning this assumption is relaxed in order to create new models able to adapt in a changing environment. Data distributions can change in different ways. The type of data drift should be taken into account in order to correctly design data streams and learning process [11]. [5] highlighted the fact that in current continual learning formulations, the data-stream is divided into separated tasks which arrive in a predefined order. They call this particularity *disjoint task formulation* and propose a new scenario that does not use this formulation and which is similar to a Task-agnostic IL scenario (see Formulation E in table I). [11] also provides an overview of data stream evolution possibilities. As explained in the paper, this characterization of data drift is important in order to have a better visualization of continual learning strengths and weaknesses. As one of the goals of this study is to define the different scenarios of continual learning, it is important to position it in relation to these definitions. Based on various surveys [17]–[19], the authors proposed to distinguish two types of context drift: *real concept drift* and *virtual drift*. *Real concept drift* consists in a change in the data label distribution i.e. for the same example the label will change over time. This type of drift will not be studied here because it is not common in classification problems where labels does not change over time. *Virtual drift* consists in a change in the data distributions that does not affect labels associated with each samples. It can be separated into two type of drift: the *label shift* which is a drift in samples that leads to new label emergence and the *domain drift* which consists in having new instances in the same label space. Label shift of data is equivalent to class-IL scenario and domain drift equivalent to domain-IL scenario.

Figure 1 summarizes the concepts described above in an atlas. It highlights the similarities and differences between scenarios and unifies literature terms. The atlas is structured around three questions:

- Definition of task boundaries: if task boundaries are not defined, the considered scenario is Task-agnostic IL, the algorithm does not have any information on when it has to train again the model, a process of novelty detection has to be deployed in this case.
- Task identity availability: if task identity are available during testing, we are in the case of a Task-IL scenario, if not, the scenario type can be Class-IL, Domain-IL or NIC depending on data in task 1 compared to task 0.
- Shift in data distribution: for the Domain-IL scenario, data for task 1 can belong to the same distribution as task 0 or a shift can exist in the data space.

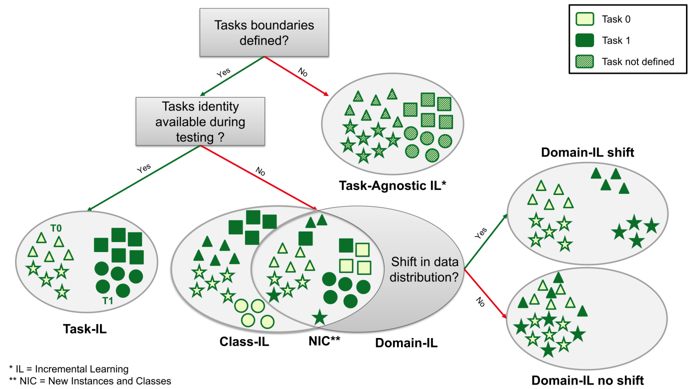


Fig. 1: Atlas of continual learning scenarios

This figure highlights the similarities and differences between scenarios and unifies literature terms.

2) Incremental batch learning vs Online continual learning vs Online streaming learning: This section gives an overview of existing streaming formulations. [20] and [21] introduced the term of **incremental batch learning**. As explained in both papers, incremental batch learning is a way to present data for training that is used in most existing continual learning methods. In this type of setting, samples of the current task comes by batches and can be seen several times by the algorithms. The training is possible only after having accumulated enough data in order to constitute a sufficiently large batch of examples of the current task. Most of the time, all the data of a task are presented at the same time and the algorithm can be trained over several epochs. This type of formulation was very useful to work on the catastrophic forgetting issue. In fact, the case where each task contains only one class is the most prone to forgetting problems as the distribution of data is non stationary by default in this type of setting. Nevertheless this formulation does not take into account certain real-life applications constrains where algorithms must constantly adapt to a changing environment with limited memory size.
[5] emphasize the difference between what they call online and **offline continual learning**. According to their definition, offline learning corresponds to incremental batch learning detailed by [21] as it consists of an unlimited access to all samples of a given task to update the model. At contrary, in online continual learning formulation, the model can store samples when they arrive but is only allowed to use them once for its update (i.e. the training only covers one epoch at each step). This formulation provides a better adaptability to on going data-flows as it requires less computing time and memory storage thanks to the limitation of training to one epoch.

[20] and [21] argue that online learning as defined by [5] does not provide sufficiently strong constraints to be a good streaming learning setting. They therefore propose a setting suitable to more restrictive real-life situations, and introduce the notion of **online streaming learning** which brings a very strong constraint for the data stream: each sample arrives one at a time and the model should adapt to it in a ”single-pass”. However, even if constraints imposed by online streaming learning are useful for some applications such as security, online learning for real-life scenario should not be limited to this setting. For example, even for an embedded device with very limited resources, a small amount of storage space may be available and the reduction of data-stream to one example at a time not mandatory.

B. Continual learning methods

Since the 90s, various algorithms have been proposed in order to deal with continual learning issues. [4], [22] divide them into three groups: regularization, parameter isolation, and replay.

Regularization methods such as Learning without Forgetting (LwF) [23] or Elastic Weight Consolidation (EWC) [24] are based on parameters update control during the model training phase in order to not loose past knowledge. The issue raised by this type of algorithm is their difficulty in remaining stable on past learned knowledge while being enough plastic to learn new knowledge [25], [26].

Parameter isolation methods, also called ”Dynamic architectures” [7], consist in allocating new neuronal resources in existing ANN when acquiring new knowledge. [27]–[30] propose models based on neuronal expansion during the learning process. As explained in [7] paper, the two main drawbacks of this kind of method are that they are difficult to use in an online manner. Indeed, they need a lot of information about the new task in order to train correctly new neuronal resources and those model need to be used in a task-incremental learning approach because the current task they are evaluating during the testing phase needs to be known.

Replay methods are shown in many papers and reviews to be the best state of the art methods that enable to overcome catastrophic forgetting in a continual learning setting [4], [9], [31], [32]. Replay methods can be divided into two categories: Rehearsal and Pseudo-rehearsal. In the rehearsal approach, a traditional way to alleviate catastrophic forgetting is to store a portion of the past knowledge and train it alongside new examples [5], [33]. Other methods also combine the use of a memory buffer and knowledge distillation which is a process that enable to transfer knowledge from a model to another [31], [34]–[36]. Instead of storing examples from previously learned knowledge, the pseudo-rehearsal approach consist in generating an auxiliary set of examples, named ”pseudo-examples”, that represent the original past input distribution. Pseudo-rehearsal methods usually use generative models in order to generate samples from the latent space of auto-encoders or generative adversarial networks that represent the previously learned knowledge [32], [37], [38]. Pseudo-rehearsal is also possible without a generative model, it consists in capturing model knowledge function with random noise using a re-injection procedure and a dual-network architecture [10], [39]–[41]

C. Summary and problem formulation

Table I summarizes different existing state of the art continual learning models in the frame of continual learning scenarios and streaming formulations proposed above. Most of the models presented here are replay-based methods. In fact, as explained before, replay-based approaches are the best state of the art methods for classical continual learning. Moreover, this type of methods is more compatible with online learning settings than parameter-isolation ones as they can be used in all type of existing continual learning scenarios. Another similarity is that all those models have been studied in a virtual drift configuration for data streams as defined by [11] which means that labels will not evolve across time. We describe here some of the models listed in the table. LwF, EWC, Icarl, BIC, UCIR and PoDNet are quite well-known state of the art models that overcome catastrophic forgeting in Task-IL or Class-IL scenarios. Icarl, BIC, UCIR and PODNet are particularly known for scaling up to database with a large number of classes like Imagenet-100 or Imagenet-1000. [15] proposed a model designed for Task-agnostic IL, BGD. [42] and [8] propose two memory-based models, GSS and ER-MIR which focus on finding the smarter memory update strategy. The first study takes place in a continual learning scenario where tasks boundaries are blurry while the second one is evaluated in a class-IL scenario. [5] propose a memory-based model, Gdumb, that stores samples in order to keep classes distribution in a memory and train a network from scratch with this memory. The authors also propose different formulations of continual learning which we sum up in table I with previously exposed terminology. Gdumb’s formulation E is directly inspired from [42] data stream proposal. Then, [7] propose a pseudo-generative model for online classification which is robust in a class-IL scenario with online streaming formulation. [9], [20], [48] and [21] propose four models, ExStream, SLDA, REMIND ans CIOSL, that are robust in an online streaming learning formulation. ExStream and REMIND are memory-based models while SLDA combines linear discriminant analysis and Deep learning and CIOSL use both regularization and replay approaches. In both studies, the authors define four types of streaming scenarios: (1) streaming i.i.d: the data stream contains randomly shuffled samples from all the dataset. (2) streaming class-i.i.d: the data stream contains samples from all the data set organized by classes. (3) streaming instance: the data stream is temporally organized with different instances of each classes (for example: 10 instances of boats, 15 instances of cats, 5 instances of boats etc...). (4) streaming class-instance: the data stream is organized by class and instances inside each classes are temporally organized (for example: all instances of boat, all instances of cats etc...). CWR* and AR1* from [47] are two

TABLE I: Classification of state-of-the-art continual learning models based on continual learning scenarios and streaming formulations on which they have been evaluated. I-Batch refers to Incremental-Batch learning and Online-Stream to Online-Streaming learning

CL scenario	Model	Streaming formulation		
		I-Batch	Online	Online-Stream
Task-Agnostic IL	GSS [42][1]	-	✓	-
	BGD [15]	-	✓	-
	Gdumb - Formulation E [5][1]	-	✓	-
Task-IL	LwF [23]	✓	-	-
	EWC [24]	✓	-	-
	Gdumb - Formulation C [5]	✓	-	-
	Gdumb - Formulation D [5]	-	✓	-
Class-IL	**Dream Net - Data Free** [10]	✓	-	-
	Dream Net - Combined replay [43]	✓	-	-
	Icarl [31]	✓	-	-
	BIC [44]	✓	-	-
	UCIR [45]	✓	-	-
	PoDNet [46]	✓	-	-
	Gdumb - Formulation B [5]	✓	-	-
	CWR* & AR1* - Scenario NC [47]	✓	-	-
	ER-MIR [8]	-	✓	-
	Gdumb - Formulation A [5]	-	✓	-
	Pseudo generative model [7]	-	✓	-
	ExStream - Class iid / Class instance [48]	-	-	✓
	SLDA - Class iid / Class instance [20]	-	-	✓
	Remind - Class iid / Class instance [9]	-	-	✓
	CIOSL - Class iid / Class instance [21]	-	-	✓
Domain-IL	CWR* & AR1* - Scenario NI [47]	✓	-	-
	ExStream - iid / instance [48]	-	-	✓
	SLDA - iid / instance [20]	-	-	✓
	Remind - iid / instance [9]	-	-	✓
	CIOSL - iid / instance [21]	-	-	✓
NIC	CWR* & AR1* - Scenario NIC [47]	✓	-	-

[1] we cannot totally consider that those models are evaluated in a Task-agnostic IL scenario as the boundaries are "blurry" and not non-existent

rehearsal-free methods that have been validated in Class-IL, Domain-IL and NI scenarios which is not common in the state of the art. Unlike most other models, those ones have been evaluated on Core50 database [49] which is well designed for NIC scenarios. Finally, Dream Net is a pseudo-rehearsal model that can have two different architectures: Dream Net - Combined replay [43] that uses a buffer to store some samples of past knowledge and Dream Net - Data Free [10], a privacy preserving model that do not store any example previously learnt. Both models have a dual-network architecture and use a process of re-injection sampling in order to generate pseudo-examples that represents learned function. Unlike Combined replay, Data Free has only been evaluated in a context of facial emotion recognition.

Based on this state of the art overview, this study will focus on exploring the possible sub-scenarios of Class-IL and Domain-IL and evaluate them on Dream Net - Data Free (in bold in the table) which is particularly promising for real-life application because of privacy concerns. We decided to not consider the Task-IL scenario in this study because knowing the current task during testing phase is not compatible with a streaming setting. We also leave aside the Task-agnostic IL scenario as some studies shows that this kind of scenario was not compatible with a replay-based algorithm alone [15], [16]. The idea of adding novelty detection to replay-based models in order to evaluate them on this type of scenario is in this study's perspectives. Models in italic in table I are those with which we will compare the results of Dream Net - Data Free in some scenarios in section IV.

III. Experiments

A. Scenarios

The review about different terms and concepts of continual learning and streaming learning presented in section II enables to define scenarios for real-life issues like a smart device learning in a changing environment. In near sensor settings, algorithms must be able to learn examples from a new class or from a new data distribution. Table II sums up the different scenarios we want to explore in this study. Following the problem formulation done before, two distinct categories of scenarios are presented: Class-IL scenarios and Domain-IL scenarios. The four sub-scenarios for each category are presented below.

Scenarios 1 - Class-IL

Scenario 1.A corresponds to a classical incremental batch learning scenario of continual learning without taking into account any streaming restrictions. It is important to test this scenario in order to position Dream Net with respect to standard continual learning state of the art. This scenario can

TABLE II: Description of eight continual learning scenarios

	Scenario 1 - Class-IL	Streaming formulation
A	Class(es) by class(es)	Incremental batch learning
B	Shuffled class(es) by class(es)	Online learning
C	Exclusively present one already learned class	Online learning
D	Present all already learned classes except one	Online learning

	Scenario 2 - Domain-IL	Streaming formulation
A	New samples of already known classes randomly shuffled	Online learning
B	New samples of already known classes with one predominant class	Online learning
C	New samples of already known classes with different predominant class	Online learning
D	2.A with batch-size 1	Online streaming learning

be trained with several classes in each task, for each database, a task is composed of a partition of 10% of data (i.e. one class per task for a dataset with 10 classes and 10 classes per task for a dataset with 100 classes). Then, with scenario 1.B, we propose to explore an online learning scenario where tasks are composed of a few samples. Each task contains examples from different classes (one or several). The training of each task is done only on one epoch. Scenario 1.C and 1.D purpose is to evaluate the ability of the model to generalize and avoid forgetting even if it does not see the examples of some classes anymore. For both scenarios an initial training is done with enough examples to reach the offline accuracy of the model for the concerned database. Then, for scenario 1.C, the data stream is composed of tasks with a little number of samples from one class exclusively, all other classes are not represented anymore in the 100 next tasks. Inversely, for scenario 1.D, the 100 next tasks are composed of all available classes excepted one.

Scenarios 2 - Domain-IL

In Scenario 2.A, each tasks are composed of only few samples like in scenario 1.B but all classes available in the dataset are represented and randomly shuffled. In this scenario there is therefore a drift of instances of classes over the data stream. Then, scenario 2.B is similar to 2.A but with one of the classes over-represented compared to the others in all tasks. In this scenario, the over-represented class is always the same while in scenario 2.C, it changes at each task. Scenarios 2.B and 2.C enable to show the effect of over-represented classes on the training. Those three scenarios are all trained in an online manner which means on one epoch only. Lastly, scenario 2.D is an online streaming learning scenario in which the data stream is composed of shuffled examples of all classes. It corresponds to the "iid" definition of ExStream and REMIND models and is equivalent to scenario 2.A with only one example per task.

For all *online learning* formulation, we consider a data stream composed of tasks containing each 50 samples. We choose this value as is the batch-size frequently used in literature [5], [7]. However, it would be interesting to do a study on this batch size in future work. For the *online streaming learning* formulation we consider by definition a data stream with tasks composed of 1 sample each. All scenarios are trained over only 1 epoch excepted the 1.A which is trained on 3 or more epoch depending of the considered database. An initial training is also provided in each case excepted 1.A as in real-life scenarios, intelligent systems are usually pre-trained in an offline manner and must then show their ability to deal with stream of new data different from the initial definition.

B. Models

Offline - Upper bound

This model is composed of an Auto-Hetero associative ANN with the same characteristics as *Learning Net* and *Memory Net*. For Domain-IL scenario (Scenario 2), we train this model with all examples at the same time in a non-streaming formulation. It gives us the maximum accuracy that can be achieved by the model. For Class-IL scenario (Scenario 1) we train this model when each new task appears with all previous tasks data. This enables to know the maximum accuracy we can obtain at each step of the training since classes can be different at each step in this scenario type.

Fine tune - Lower bound

This model is also composed of an Auto-Hetero associative ANN. We name it "fine-tune" as it only updates the model with incoming data of the current task without taking into consideration previous tasks nor being reinitialized between each task. This model is thus a kind of fine-tuning without any algorithm specifically designed for continual learning issues. It make possible to highlight when catastrophic forgetting appears in scenarios and thus considered as the lower-bound of the study.

Dream Net - Data Free

As explained in section II, Dream Net - Data Free will be evaluated on each scenario presented above. As presented in [10] paper, this model uses two Auto-Hetero associative ANNs (each auto-encoder and classifier at the same time): Learning Net and Memory Net. Memory net generates pseudo-examples that represent already learned knowledge using a re-injection sampling procedure with random noise. Learning Net learns conjointly new examples with pseudo-examples that it generates.

C. Datasets, Hyperparameters and Metrics

Datasets

To evaluate DreamNet, we use three well known datasets: Mnist, Cifar10 and Imagenet100.

- *Mnist* dataset contains 70000 gray-scale images of handwritten digits of size 28 x 28 pixel each [50]. This database is used in many deep learning studies and is considered as a proof-of-concept dataset. It is separated into 10 classes which corresponds to digits from 0 to 9. We consider 60000 images for the training phase and 10000 for the test phase. For this dataset, feature extraction is not mandatory as each image can be unfold into a vector of 784 feature. It is important for us to benchmark Dream Net - Data Free on this dataset as most of state of the art models take it as a reference.
- *Cifar10* contains 60000 32x32 images divided into 10 classes representing animals and means of transport [51]. We consider 50000 images for the training phase and 10000 for the test phase. As this dataset is more complex than Mnist, a feature extraction is done with a Resnet50 network pre-trained on Imagenet [52].
- *ImageNet* is a database of annotated images in which 1000 classes of objects appear. It presents about 1000 images per class for the training base and 50 for the test base. It is, since 2010, at the origin of an annual competition whose goal is to detect and classify as accurately as possible objects and scenes: ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [53]. It is notably within this competition that networks such as ResNet have distinguished themselves by their performances. In our study, we consider a reduction to 100 randomly selected ImageNet classes, named *ImageNet100*. We also use a Resnet50 network pre-trained on Imagenet for feature extraction of images.

For scenarios with initial pre-training, algorithms are trained on 2% of the database for Mnist and Cifar-10 and on 10% of the database for Imagenet-100. For the purpose of the experiment, scenarios 1.C and 1.D require a pre-training with enough data to be close to the the maximum offline accuracy on this database meaning 30000 examples for Mnist and Cifar-10, and 70000 examples for Imagenet-100.

Hyper-parameters

For a given database, we do not change hyper-parameters depending on continual learning scenarios because we want to demonstrate that the model is robust toward any kind of data arrival for the same set of hyper-parameters. Nevertheless, we adapt those hyper-parameters depending on the considered database and application.

Here, all experiments are performed with hyper-parameters presented Dream Net - Data Free model's paper [10]. Note that for Imagenet-100 database, Dream Net - Data Free and baseline models have two hidden layers of 1000 neurons and for Mnist database, the number of unit per hidden layer is 400 instead of 1000.

The random noise used for Dream Net - Data Free is an isotopic Gaussian noise centered at 0 with 1 as variance, N(0,1).

Metrics

We measure the performances of the model in each scenario with the average accuracy of the model after the training of each new task. We also look at confusion matrix over classes at the end of the training in order to evaluate the behavior of the model concerning data drift especially in Class-IL scenarios where classes change over the data stream. Each experiment is performed 5 times under the same conditions and our results are an average of those 5 runs, displayed with confidence intervals at 95%.

IV. Results

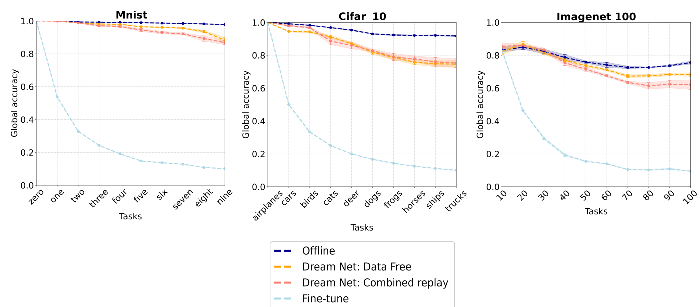
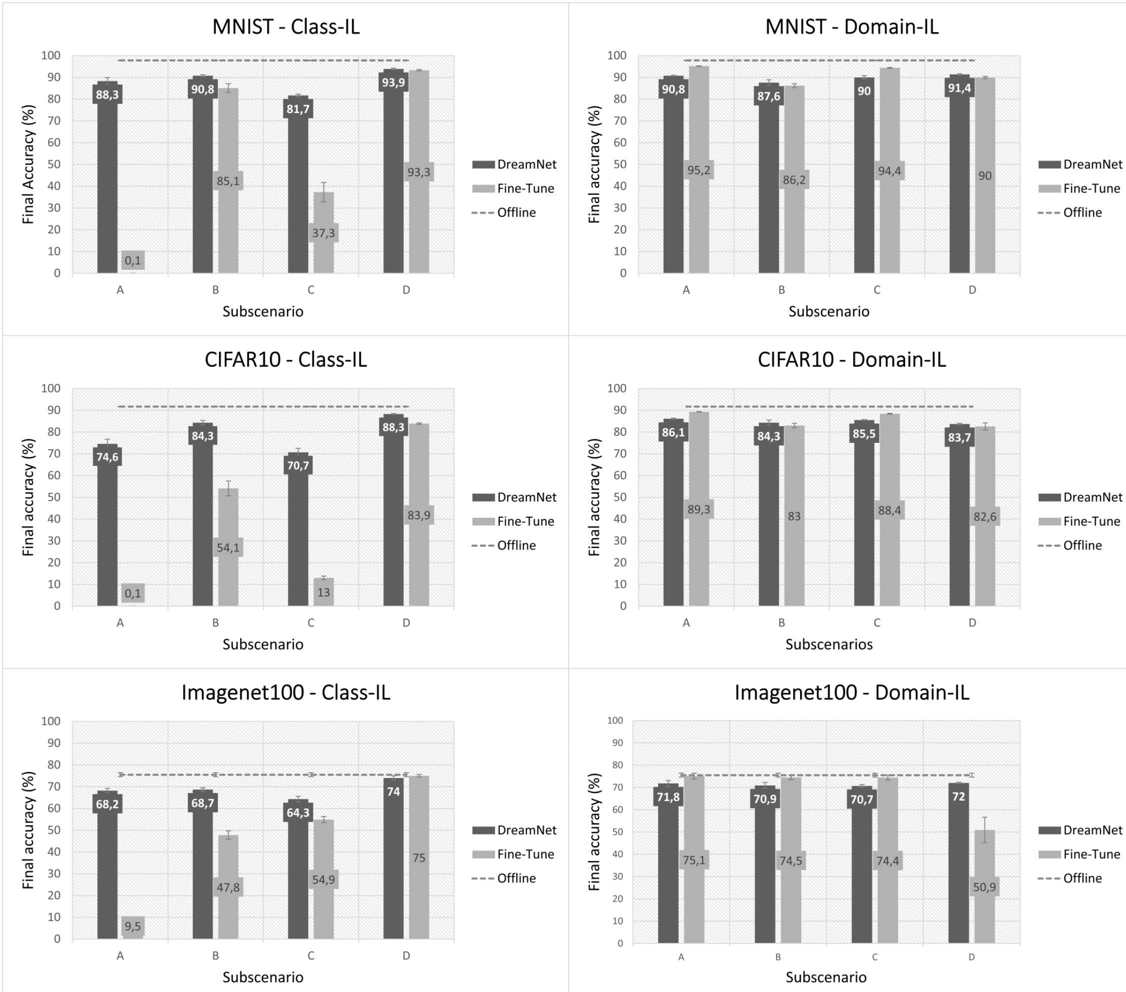
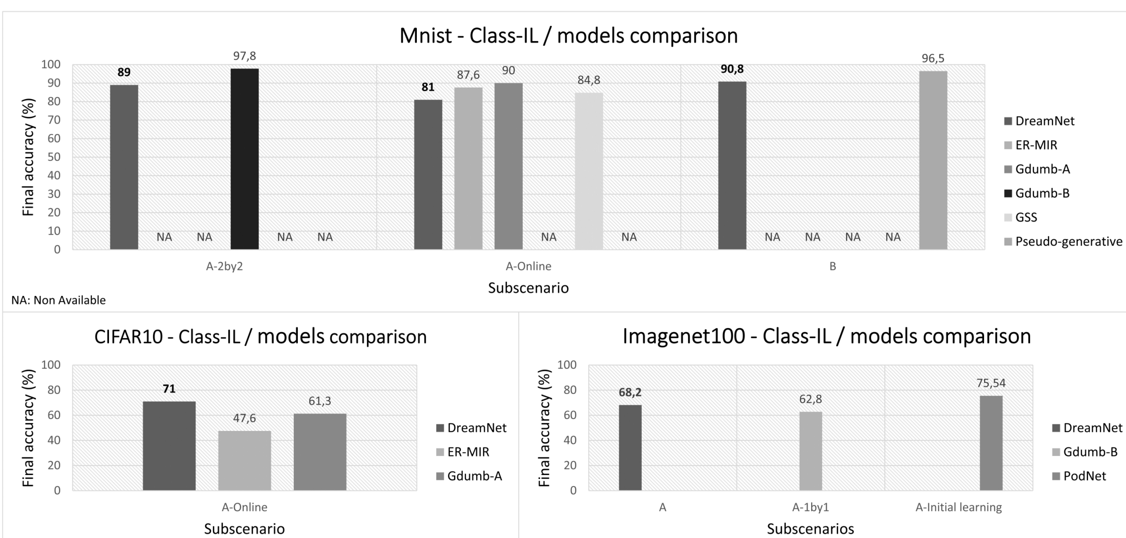


Fig. 2: Global accuracy across tasks for Class-IL incremental batch learning sub-scenario 1.A. Dream Net - Data Free is compared to baseline models and Dream Net - Combined replay model on Mnist, Cifar10 and Imagenet100

This section presents experimental results of this study over scenarios and databases and compare them over other state of the art models. Offline model accuracy for Mnist, Cifar10 and Imagenet100 are respectively: $97.8\% \pm 0.1\%$, $91.7\% \pm 0.4\%$ and $75.5\% \pm 0.9\%$. Due to the auto-hetero architecture of the model and to absence of hyper-parameters optimization, accuracy may be lower than for literature models which are simple classifiers. However, it is important to compare accuracy results of every scenarios to those of an offline model that has the same architecture. Figure 2 shows results accross databases for the first Class-IL sub-scenario. In this scenario, Mnist and Cifar-10 classes are learned 1 by 1 while Imagnet-100 classes are learned 10 by 10. For all databases, there is no initial training. Fine-tune baseline model highlights the catastrophic forgetting effect. For each database, Dream Net - Data Free overcomes the catastrophic forgetting issue with a final accuracy competitive with Dream Net - Combined replay that uses a buffer to store past knowledge. Figure 3a gives results of Dream Net - Data Free model over each scenario defined in figure II. Results are presented with histograms in order to easily compare final accuracy of Dream Net in each configuration with baseline models accuracy. This figure highlights that Dream Net - Data Free is robust in each sub-scenario and for the three databases. Fine-Tune accuracy show



(a) Final accuracy results of Dream Net - Data Free and Fine-Tune across eight scenarios described in table II and three dataset, Mnist, Cifar10 and Imagenet100. Final accuracy of online model is displayed in dotted line on each graphic.



(b) Comparison of Dream Net - Data Free to literature models (highlighted in *italic* in table I) on some specific scenarios

Fig. 3: Final accuracy of Dream Net - Data Free across scenarios and datasets compared to baseline and literature models.

that in the case of Class-IL scenarios catastrophic forgetting appears. But it also brings out the fact that for Mnist, Cifar10 and Imagenet100 databases, domain-IL scenarios do not lead to catastrophic forgetting. We can explain it by the fact that there is no shift in data distributions for those databases. Despite this last comment, those results are very promising regarding the ability of Dream Net - Data Free to deal with domain incremental learning scenarios with data distribution shift. In fact, its stability across scenarios with and without catastrophic forgetting gives an idea of its ability to deal with more difficult distribution shifts. Moreover, Dream Net - Data Free final accuracy is above fine-tune model for scenario 2.D, this confirms again the ability of this model to deal with complex and restrictive data streams (here data stream is restricted to one sample at time). Figure 3b provides a comparison with literature models in italic in table I. As most models were evaluated only on specific Class-IL scenarios, we cannot compare Dream Net - Data free in each scenarios. A-1by1, A-2by2, A-Online and A-initial learning are variants of sub-scenario 1.A which corresponds respectively to: training classes one by one, training classes two by two, training on one epoch only (online learning with large batches) and network initially trained with 50 classes before training classes one by one. Even if it does not store any example of the past knowledge in buffer unlike other models, figure 3b shows that Dream Net - Data Free is competitive with other state of the art model. We do not present literature comparison on Domain-IL scenarios here because models like ExStream, SLDA, Remind and CIOSL uses a different metric called normalized incremental learning performance which depends non-linearly on offline accuracy, it is thus difficult to compare our final accuracy with it. And CWR* and AR1* were evaluated only on Core50 database which has not been implemented in this study.

V. Conclusion and perspectives

In real-life applications, a smart device must be agnostic to data arrival scenarios and be able to learn new information from new classes or new instances of already known classes. For this reason, in this paper, we provided an overview of continual learning scenarios depending on data stream constitution. We also demonstrated that Dream Net - Data Free was a robust continual learning algorithm able to deal with various types of continual learning scenarios and data streams. This model presents stable results over the eight proposed scenarios. This robustness is a real asset for Dream Net - Data Free highlighting its ability to be integrated in real-life scenarios where the order of arrival of data is unpredictable. Moreover, this model is very interesting for privacy issues as it does not store any data learned before. It is for instance particularly interesting for human-centered applications. The efficiency of the model has already been proven for face emotion recognition in the Class-IL scenario [10]. We plan to extend this study to the other continual learning scenarios described in this work to explore the model's ability to generalize face emotion recognition across multiple ethnicities. We are also considering other real-life applications such as health or environment monitoring. One of the next steps of this work is to test Dream Net - Data Free on databases where a shift in the distribution of data takes place in order to explore Domain-IL shift and NIC scenarios. In this type of setting, we expect catastrophic forgetting because a drastic shift in data occurs. We especially plan to implement Core50 database in our evaluation framework. Future work will also move towards a more autonomous system and explore Task-agnostic IL scenarios to allow the system to incrementally detect novelty.

VI. Acknowledgments

The authors thank Julie GALLIERE and Miguel SOLINAS for their preliminary work on Imagenet100, and Katie MATTON, Matthew GROH and Adrien VIAULT for their feedbacks on early drafts of this paper.

References

[1] C. S. Lee and A. Y. Lee, "Clinical applications of continual learning machine learning," *The Lancet Digital Health*, vol. 2, no. 6, pp. e279–e281, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2589750020301023

[2] D. Kiyasseh, T. Zhu, and D. Clifton, "A clinical deep learning framework for continually learning from cardiac signals across diseases, time, modalities, and institutions," *Nature Communications*, vol. 12, no. 1, p. 4221, Dec. 2021. [Online]. Available: http://www.nature.com/articles/s41467-021-24483-0

[3] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of Learning and Motivation*, G. H. Bower, Ed. Academic Press, 1989, vol. 24, pp. 109–165. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0079742108605368

[4] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," 2020. [Online]. Available: http://arxiv.org/abs/1909.08383

[5] A. Prabhu, P. H. S. Torr, and P. K. Dokania, "GDumb: A simple approach that questions our progress in continual learning," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Springer International Publishing, 2020, vol. 12347, pp. 524–540, series Title: Lecture Notes in Computer Science. [Online]. Available: http://link.springer.com/10.1007/978-3-030-58536-5_31

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

[7] A. Besedin, P. Blanchart, M. Crucianu, and M. Ferecatu, "Deep online classification using pseudo-generative models," vol. 201, p. 103048, 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1077314220300941

[8] R. Aljundi, L. Caccia, E. Belilovsky, M. Caccia, M. Lin, L. Charlin, and T. Tuytelaars, "Online continual learning with maximally interfered retrieval," 2019. [Online]. Available: http://arxiv.org/abs/1908.04742

[9] T. L. Hayes, K. Kafle, R. Shrestha, M. Acharya, and C. Kanan, "REMIND your neural network to prevent catastrophic forgetting," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Springer International Publishing, 2020, vol. 12353, pp. 466–483, series Title: Lecture Notes in Computer Science. [Online]. Available: https://link.springer.com/10.1007/978-3-030-58598-3_28

[10] M. Mainsant, M. Solinas, M. Reyboz, C. Godin, and M. Mermillod, "Dream net: a privacy preserving continual leaming model for face emotion recognition," in *2021 9th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, 2021, pp. 01–08.

[11] T. Lesort, M. Caccia, and I. Rish, "Understanding continual learning settings with data distribution drift analysis," 2021. [Online]. Available: http://arxiv.org/abs/2104.01678

[12] Y.-C. Hsu, Y.-C. Liu, A. Ramasamy, and Z. Kira, "Re-evaluating continual learning scenarios: A categorization and case for strong baselines," 2018, version: 1. [Online]. Available: http://arxiv.org/abs/1810.12488

[13] G. M. van de Ven and A. S. Tolias, "Three scenarios for continual learning," 2019, version: 1. [Online]. Available: http://arxiv.org/abs/1904.07734

[14] V. Lomonaco, D. Maltoni, and L. Pellegrini, "Rehearsal-free continual learning over small non-i.i.d. batches," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Los Alamitos, CA, USA: IEEE Computer Society, jun 2020, pp. 989–998. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPRW50498.2020.00131

[15] C. Zeno, I. Golan, E. Hoffer, and D. Soudry, "Task agnostic continual learning using online variational bayes," 2019. [Online]. Available: http://arxiv.org/abs/1803.10123

[16] R. Aljundi, K. Kelchtermans, and T. Tuytelaars, "Task-free continual learning," pp. 11 246–11 255, 2019, ISSN: 2575-7075.

[17] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," vol. 46, no. 4, pp. 1–37, 2014. [Online]. Available: https://dl.acm.org/doi/10.1145/2523813

[18] A. Gepperth and B. Hammer, "Incremental learning algorithms and applications," p. 13, 2016.

[19] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," vol. 31, no. 12, pp. 2346–2363, 2019, conference Name: IEEE Transactions on Knowledge and Data Engineering.

[20] T. L. Hayes and C. Kanan, "Lifelong machine learning with deep streaming linear discriminant analysis," 2020. [Online]. Available: http://arxiv.org/abs/1909.01520

[21] S. Banerjee, V. K. Verma, T. Parag, M. Singh, and V. P. Namboodiri, "Class incremental online streaming learning," 2021. [Online]. Available: http://arxiv.org/abs/2110.10741

[22] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," vol. 113, pp. 54–71, 2019. [Online]. Available: http://arxiv.org/abs/1802.07569

[23] Z. Li and D. Hoiem, "Learning without forgetting," 2017. [Online]. Available: http://arxiv.org/abs/1606.09282

[24] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," vol. 114, no. 13, pp. 3521–3526, 2017, number: 13. [Online]. Available: http://www.pnas.org/lookup/doi/10.1073/pnas.1611835114

[25] W. C. Abraham and A. Robins, "Memory retention – the synaptic stability versus plasticity dilemma," vol. 28, no. 2, pp. 73–78, 2005, number: 2. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166223604003704

[26] M. Mermillod, A. Bugaiska, and P. Bonin, "The stability-plasticity dilemma: investigating the continuum from catastrophic forgetting to age-limited learning effects," vol. 4, 2013. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fpsyg.2013.00504/abstract

[27] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," 2016. [Online]. Available: http://arxiv.org/abs/1606.04671

[28] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A. A. Rusu, A. Pritzel, and D. Wierstra, "PathNet: Evolution channels gradient descent in super neural networks," 2017. [Online]. Available: http://arxiv.org/abs/1701.08734

[29] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," 2018. [Online]. Available: http://arxiv.org/abs/1708.01547

[30] G. Hocquet, O. Bichler, and D. Querlioz, "OvA-INN: Continual learning with invertible neural networks," 2020. [Online]. Available: http://arxiv.org/abs/2006.13772

[31] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "iCaRL: Incremental classifier and representation learning," 2017. [Online]. Available: http://arxiv.org/abs/1611.07725

[32] R. Kemker and C. Kanan, "FearNet: Brain-inspired model for incremental learning," 2018. [Online]. Available: http://arxiv.org/abs/1711.10563

[33] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with a-GEM," 2019. [Online]. Available: http://arxiv.org/abs/1812.00420

[34] F. M. Castro, M. J. Marín-Jiménez, N. Guil, C. Schmid, and K. Alahari, "End-to-end incremental learning," 2018. [Online]. Available: http://arxiv.org/abs/1807.09536

[35] B. Zhao, X. Xiao, G. Gan, B. Zhang, and S.-T. Xia, "Maintaining discrimination and fairness in class incremental learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020, pp. 13 205–13 214. [Online]. Available: https://ieeexplore.ieee.org/document/9156766/

[36] X. Hu, K. Tang, C. Miao, X.-S. Hua, and H. Zhang, "Distilling causal effect of data in class-incremental learning," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021, pp. 3956–3965. [Online]. Available: https://ieeexplore.ieee.org/document/9578597/

[37] T. Lesort, A. Gepperth, A. Stoian, and D. Filliat, "Marginal replay vs conditional replay for continual learning," 2019. [Online]. Available: http://arxiv.org/abs/1810.12069

[38] G. M. van de Ven, H. T. Siegelmann, and A. S. Tolias, "Brain-inspired replay for continual learning with artificial neural networks," vol. 11, no. 1, p. 4069, 2020. [Online]. Available: http://www.nature.com/articles/s41467-020-17866-2

[39] R. M. French, "Pseudo-recurrent connectionist networks: An approach to the 'sensitivity-stability' dilemma," vol. 9, no. 4, pp. 353–380, 1997, number: 4. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/095400997116595

[40] B. Ans and S. Rousset, "Avoiding catastrophic forgetting by coupling two reverberating neural networks," vol. 320, no. 12, pp. 989–997, 1997, number: 12. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0764446997824729

[41] M. Solinas, C. Galiez, R. Cohendet, S. Rousset, M. Reyboz, and M. Mermillod, "Generalization of iterative sampling in autoencoders," p. 9, 2020.

[42] R. Aljundi, M. Lin, B. Goujaud, and Y. Bengio, "Gradient based sample selection for online continual learning," 2019. [Online]. Available: http://arxiv.org/abs/1903.08671

[43] M. Solinas., S. Rousset., R. Cohendet., Y. Bourrier., M. Mainsant., A. Molnos., M. Reyboz., and M. Mermillod., "Beneficial effect of combined replay for continual learning," in *Proceedings of the 13th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART,*, INSTICC. SciTePress, 2021, pp. 205–217.

[44] Y. Wu, Y. Chen, L. Wang, Y. Ye, Z. Liu, Y. Guo, and Y. Fu, "Large scale incremental learning," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 374–382. [Online]. Available: https://ieeexplore.ieee.org/document/8954008/

[45] S. Hou, X. Pan, C. C. Loy, Z. Wang, and D. Lin, "Learning a unified classifier incrementally via rebalancing," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 831–839.

[46] A. Douillard, M. Cord, C. Ollion, T. Robert, and E. Valle, "Podnet: Pooled outputs distillation for small-tasks incremental learning," in *European Conference on Computer Vision*. Springer, 2020, pp. 86–102.

[47] V. Lomonaco, D. Maltoni, and L. Pellegrini, "Rehearsal-free continual learning over small non-i.i.d. batches," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 989–998.

[48] T. L. Hayes, N. D. Cahill, and C. Kanan, "Memory efficient experience replay for streaming learning," 2018, version: 1. [Online]. Available: http://arxiv.org/abs/1809.05922

[49] V. Lomonaco and D. Maltoni, "CORe50: a new dataset and benchmark for continuous object recognition," p. 10, 2017.

[50] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[51] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)," 2010. [Online]. Available: http://www.cs.toronto.edu/ kriz/cifar.html

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[53] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.