Brain-Inspired Continual Learning: Rethinking the Role of Features in the Stability-Plasticity Dilemma

Ph.D. Dissertation Defense

Hikmat Khan

April 15, 2024

Committee:

Dr. Nidhal C. Bouaynaya, Ph.D.

Dr. Ghulam Rasool, Ph.D.

Dr. Ravi P. Ramachandran, Ph.D.

Dr. Robi Polikar, Ph.D.

Dr. Shlomo Engelberg, Ph.D.

Outline

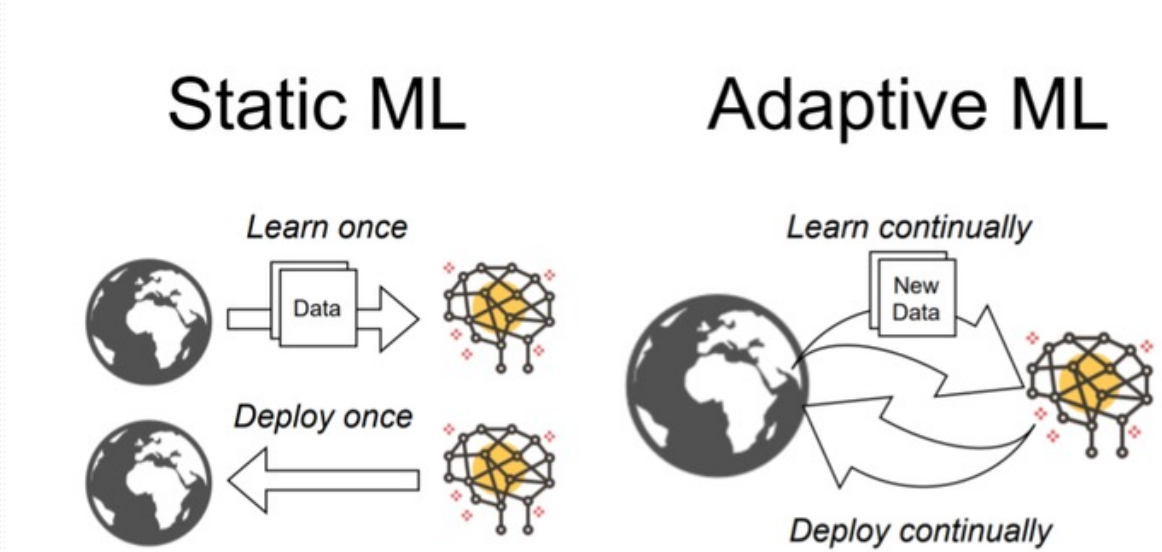
I. Introduction
   1. Continual Learning
II. Related Work
III. Research Questions
IV. Our Work: A Multidisciplinary Approach
V. Primary Contributions
   1. Adversarially Robust Continual Learning
   2. The Importance of Robust Features in Mitigating Catastrophic Forgetting
   3. Brain-Inspired Continual Learning Robust Feature Distillation and Re-consolidation for Class Incremental Learning
   4. Adversarially Diversified Rehearsal Memory (ADRM): Mitigating Memory Overfitting Challenge in Continual Learning
VI. Conclusions
VII. Future Work

Outline

Deep Learning – The Promise

- Deep Learning (DL)/Artificial Intelligence (AI) models have achieved or even surpassed human-level accuracy in several areas, including computer vision and pattern recognition.

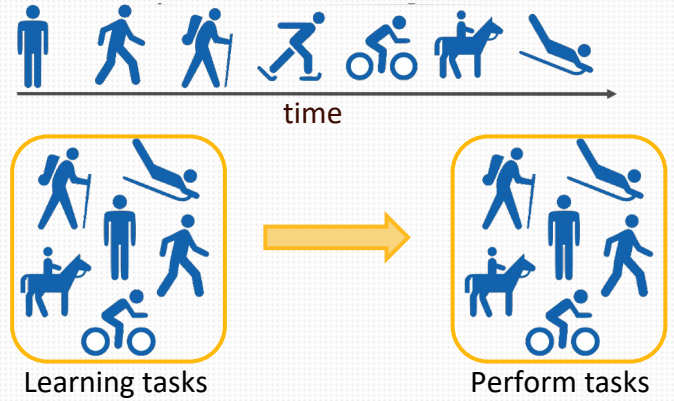
The World is Non-Stationary [5]

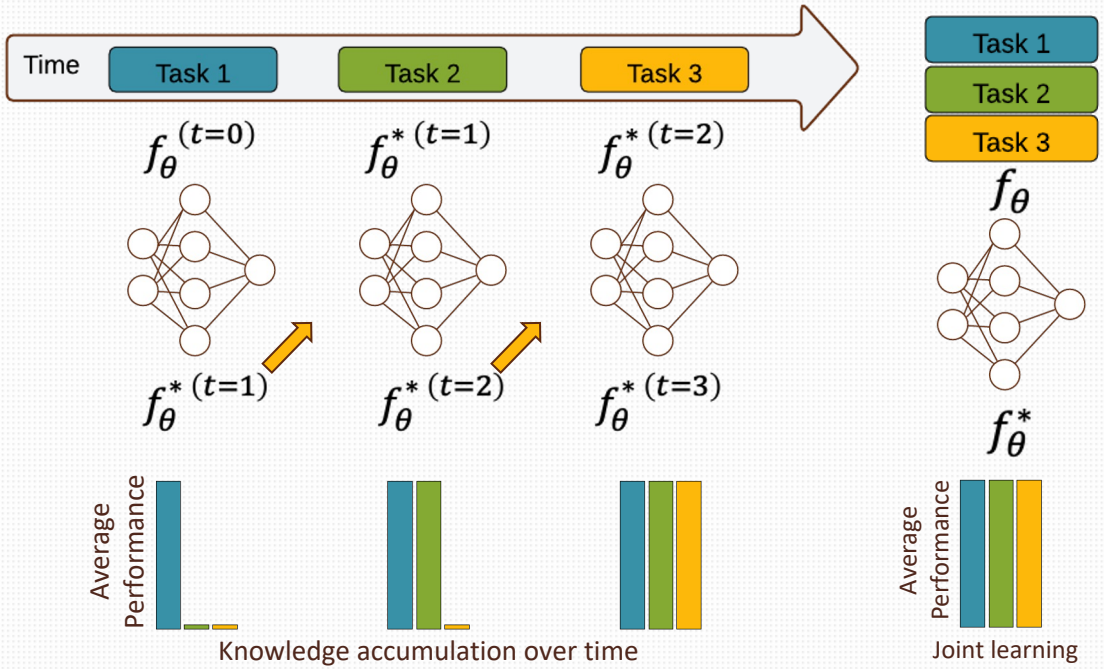


https://medium.com/gooroo-official/if-youre-not-learning-you-re-dying-b2b4d54eace4 | https://senicessm.live/product_details/47625942.html

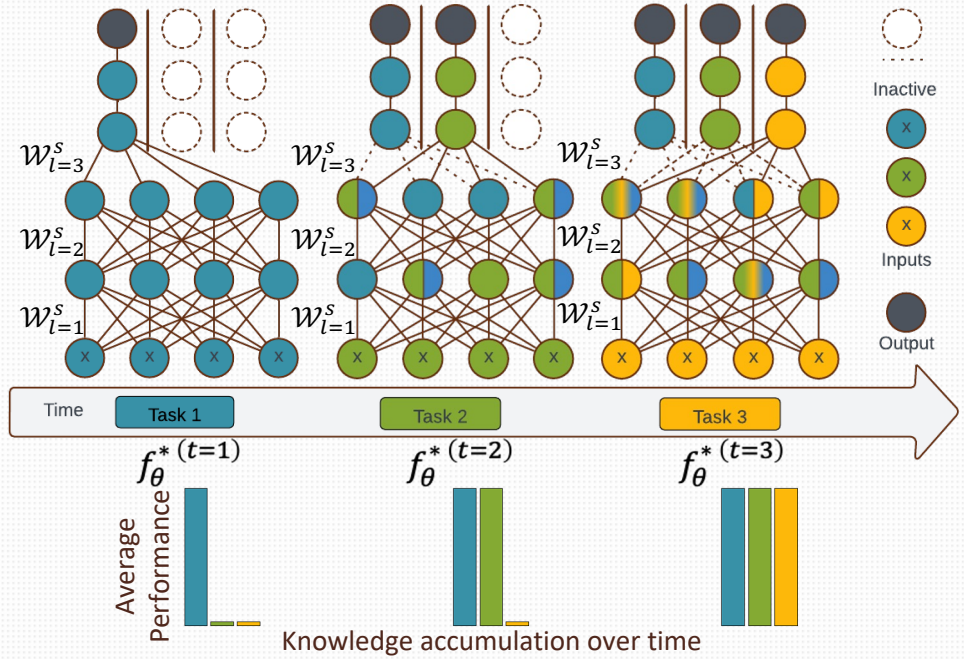
Continual Learning (CL)

- Learning tasks over time[1].
  - The tasks can be any classification or segmentation.
- Offline learning [1, 2].
  - Learn all tasks simultaneously.



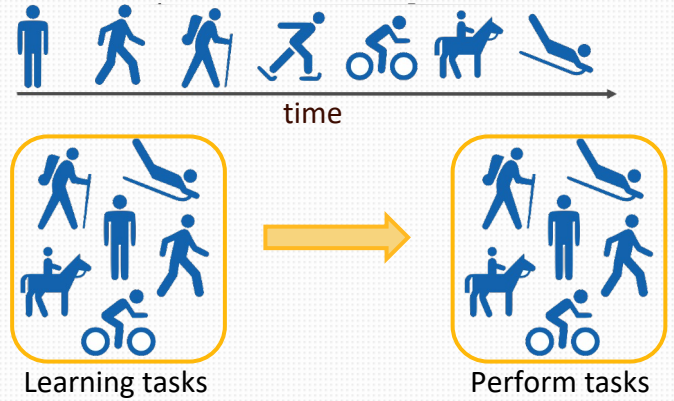


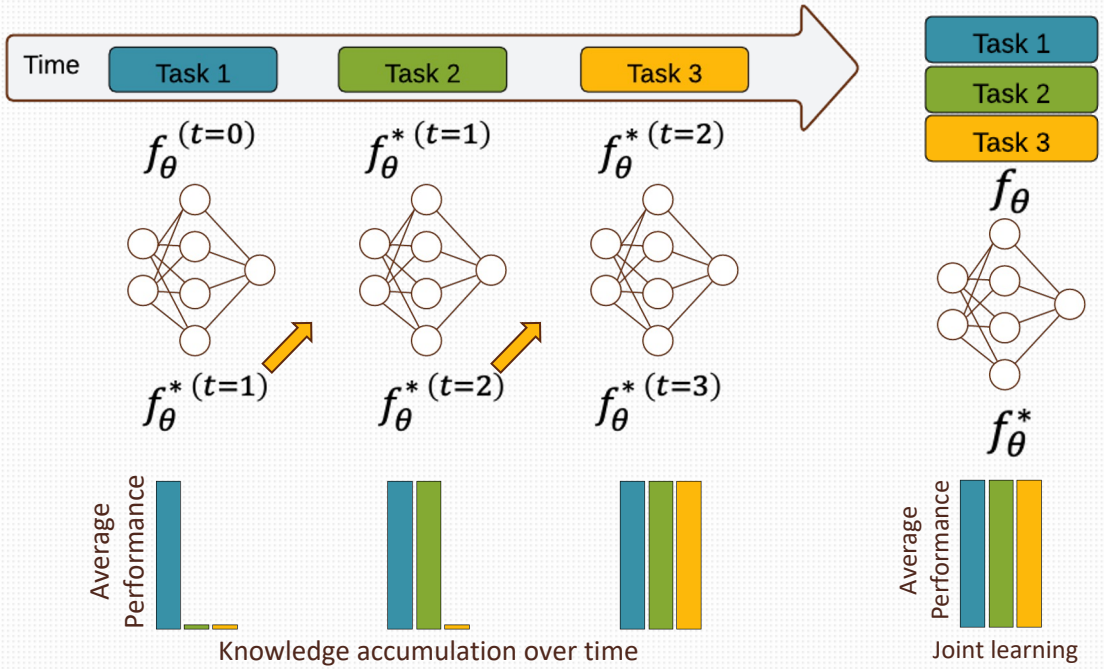
Task Incremental Learning [2]



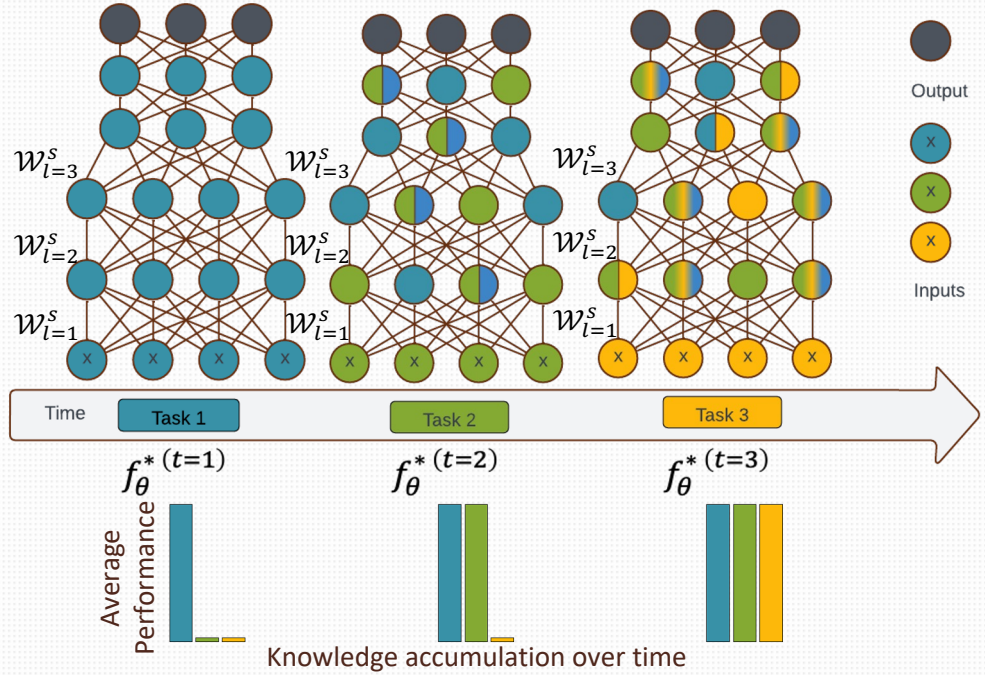
Continual Learning (CL) (Continued)

- Learning tasks over time[1].
  - The tasks can be any classification or segmentation.
- Offline learning [1, 2].
  - Learn all tasks simultaneously.



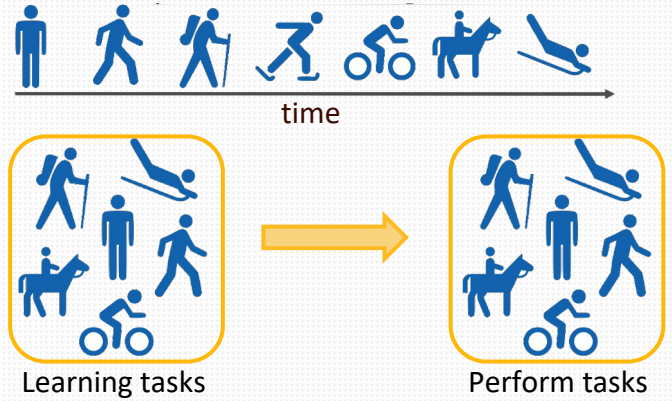


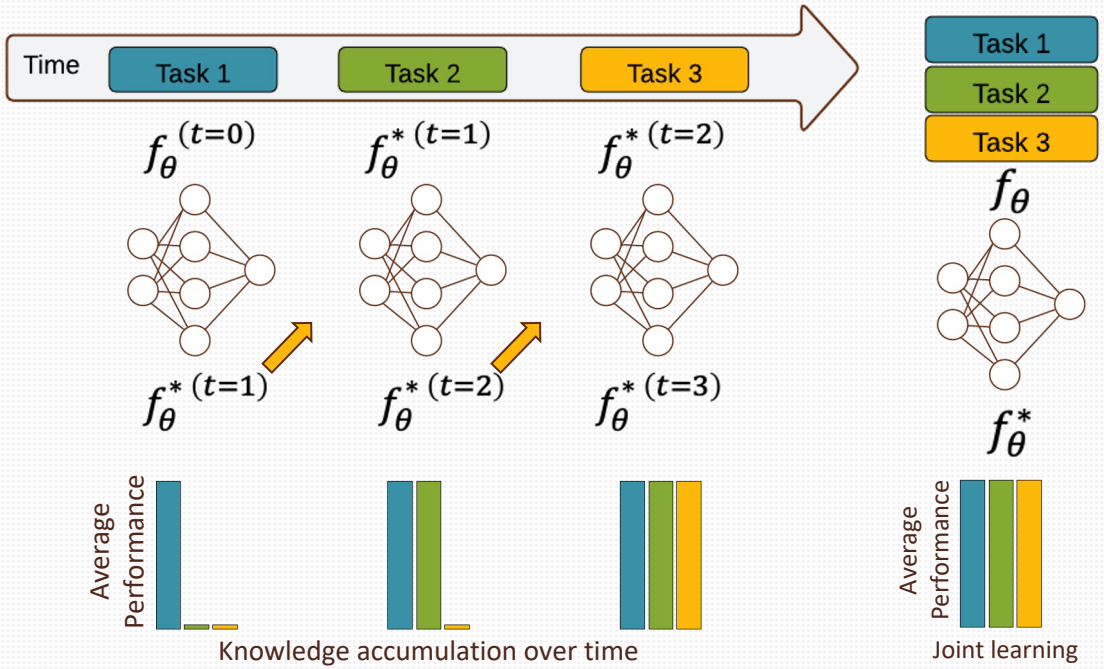
Domain Incremental Learning [2]



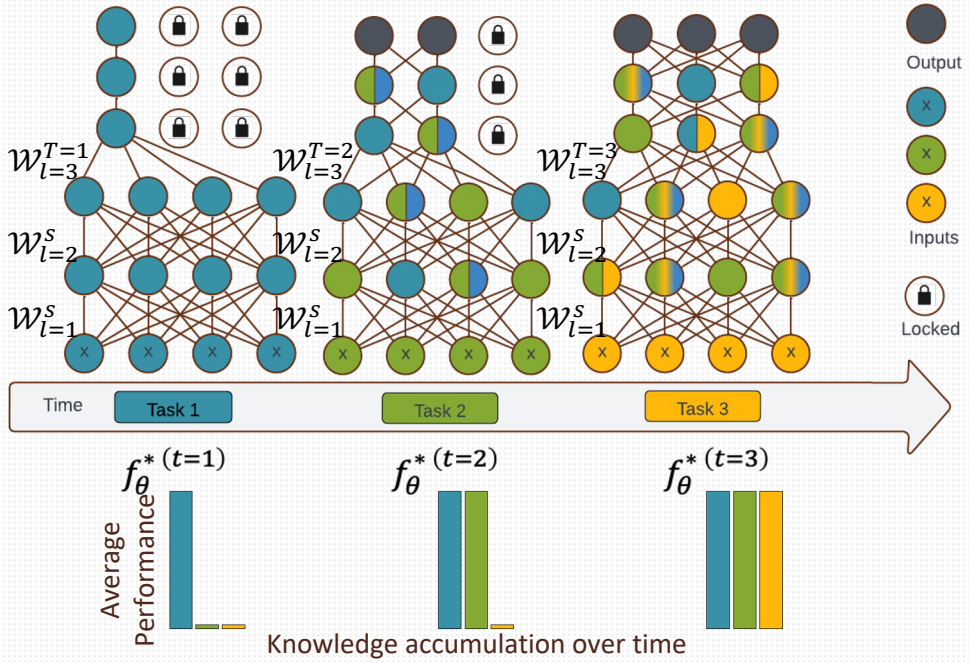
Continual Learning (CL) (Continued)

- Learning tasks over time[1].
  - The tasks can be any classification or segmentation.
- Offline learning [1, 2].
  - Learn all tasks simultaneously.



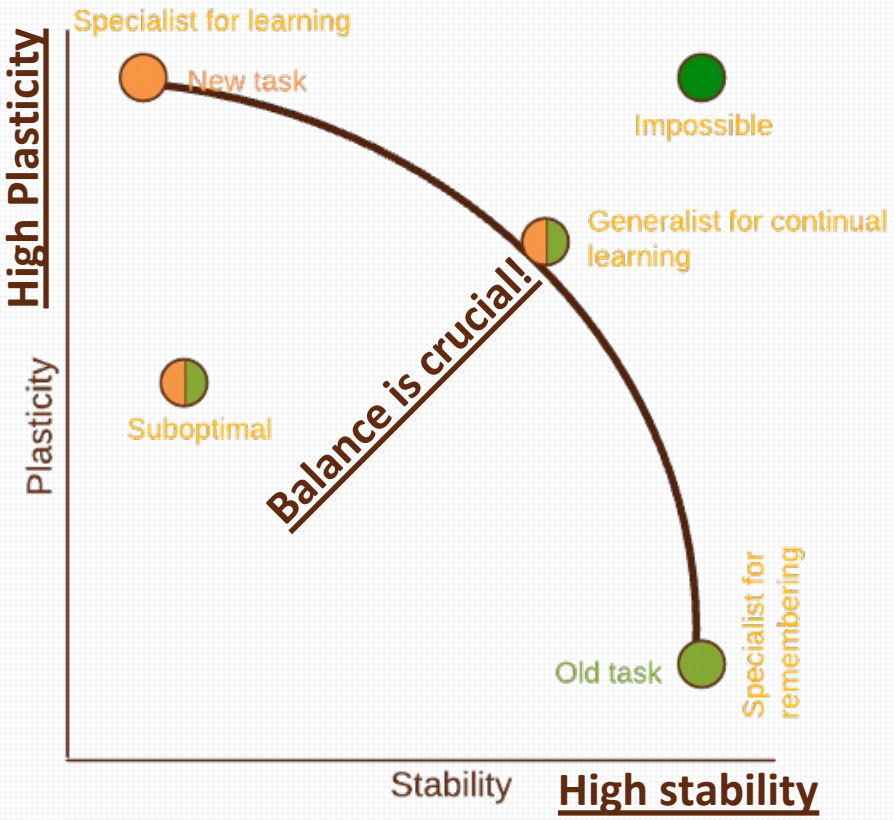


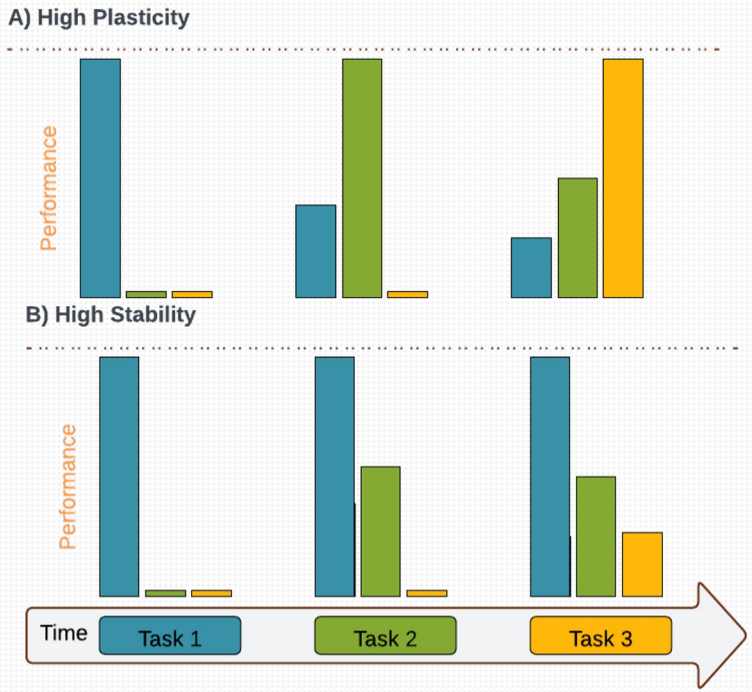
Class Incremental Learning [2]



Challenges: What is Stopping CL

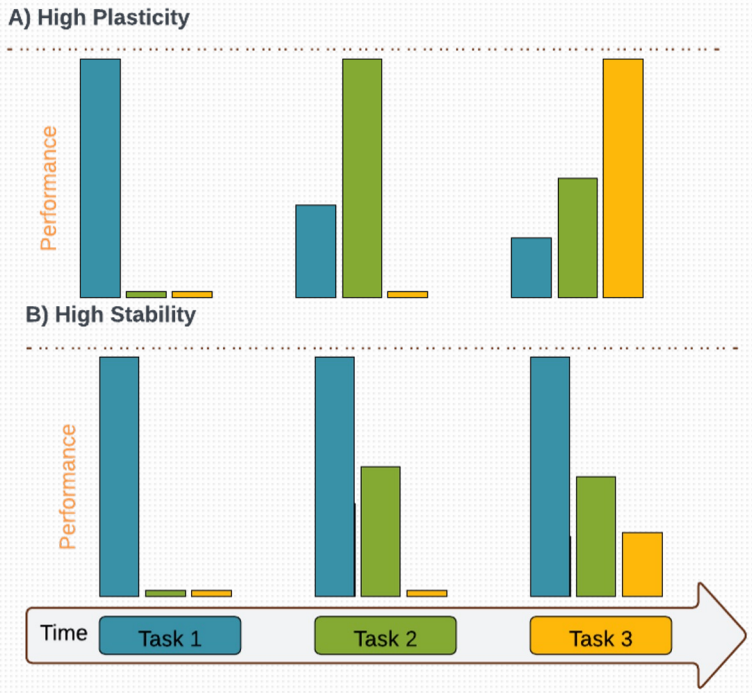
- The plasticity and stability dilemma [3].
  - The plasticity-stability dilemma is a fundamental challenge in both neuroscience and deep learning design.





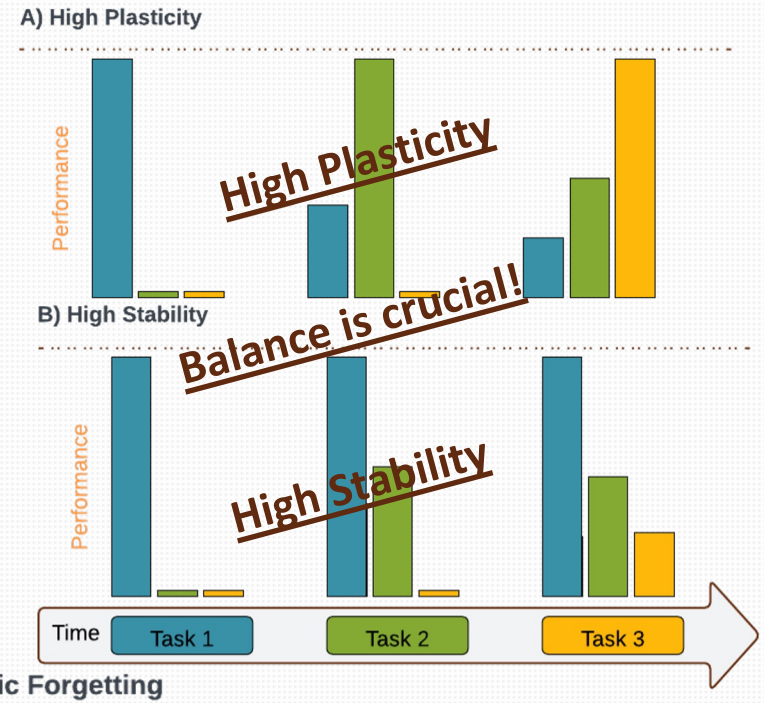
Challenges: What is Stopping CL

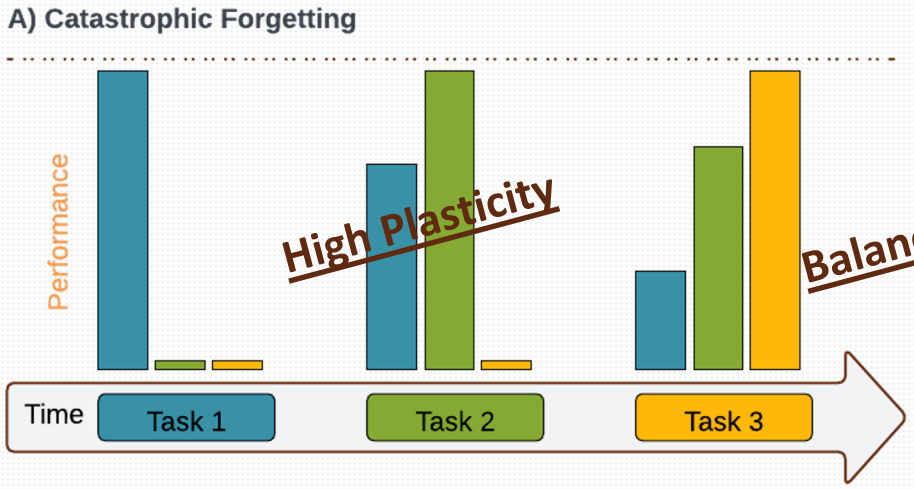
- The plasticity and stability dilemma [3].
  - Plasticity:
    - Easily learn and adapt to new experiences.
    - Enables good adaptation.
  - Stability:
    - Ability to retain previously learned experiences.
    - Ensures proficiency in solving old tasks.
- Catastrophic forgetting [4].
  - If a system is too plastic, it risks overriding old knowledge (aka catastrophic forgetting)
  - A phenomenon where a model forgets previously learned knowledge after acquiring new knowledge.

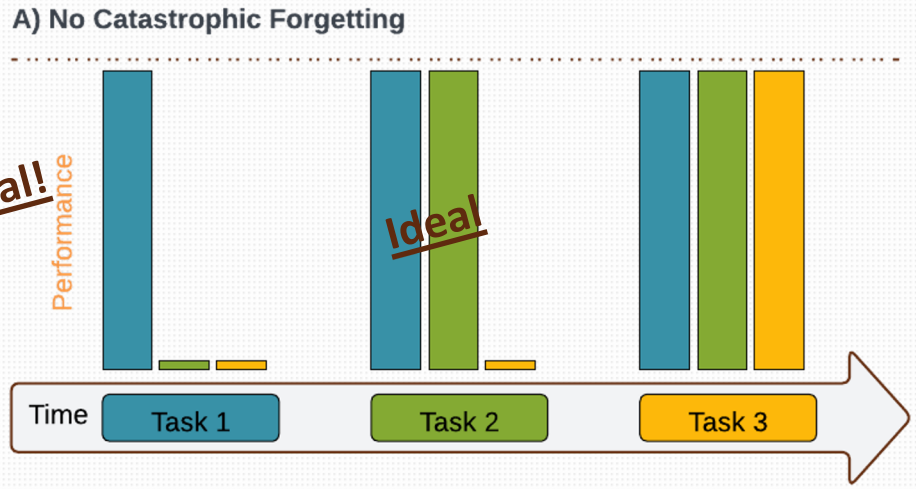


Challenges Stopping CL

- The plasticity and stability dilemma [3].
  - Plasticity:
    - Easily learn and adapt to new experiences.
    - Enables good adaptation.
  - Stability:
    - Ability to retain previously learned experiences.
    - Ensures proficiency in solving old tasks.
- Catastrophic forgetting [4].



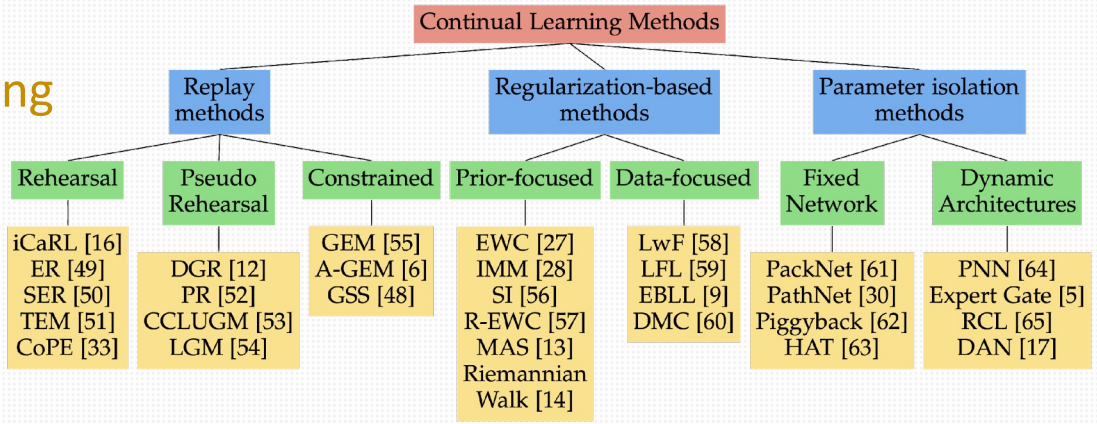




Outline

Related Work

1. Rehearsal-based Approaches [1, 5]
   - Store a subset of the old data and rehearse it during the learning of the subsequent tasks.
2. Regularization-based Approaches [1, 5]
   - Penalize (important) parameter updates.
3. Architectural-based Approaches [1, 5]
   - Increase the model capacity for every new task.
   - Explicitly identify important parameters for each task.
4. Knowledge Distillation-based Approaches [1, 5]
   - Take inspiration from the Knowledge Distillation.
   - Use the previous task model as a teacher.
5. Dataset Distillation-based Approaches [1, 5]
   - Create representative subsets that can be rehearsed during the learning of the subsequent tasks.
6. Brain-inspired Approaches [1, 5]
   - Incorporate inspiration from neuroscience in designing the CL
7. Hybrid Approaches [1, 5]
   - Combine multiple approaches to create more robust solutions against catastrophic forgetting.



Outline

Research Questions

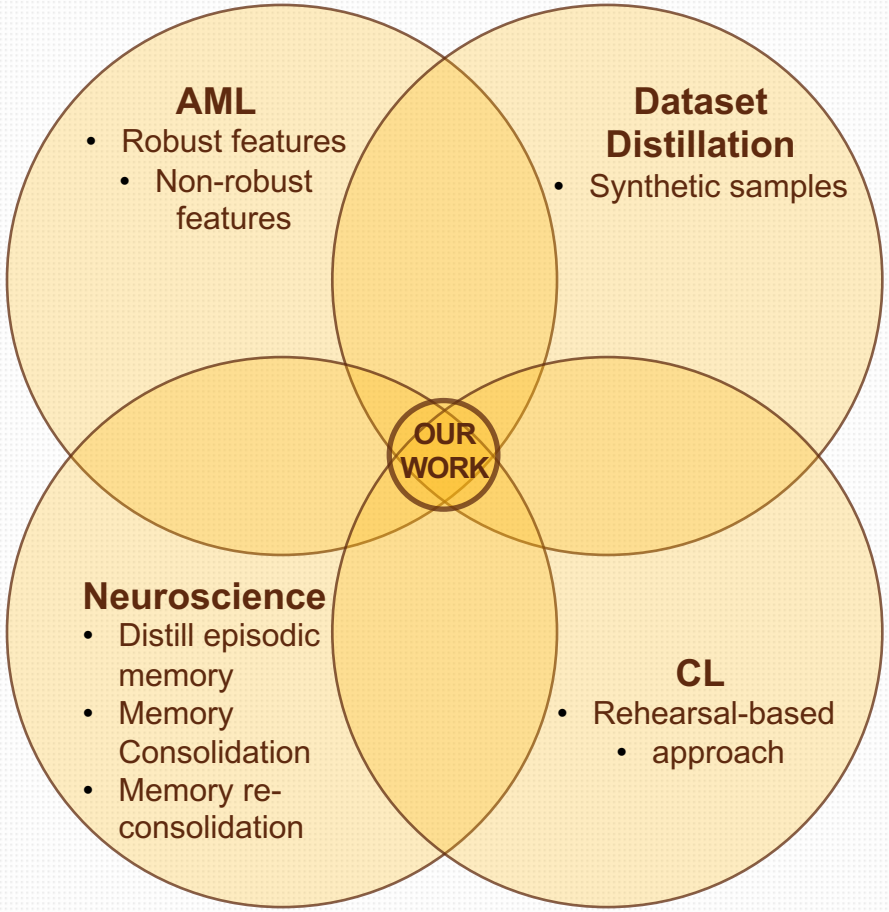
- What is the role of features (in the input dataset) in the robustness and mitigation of catastrophic forgetting in the CL? [6]
- Can disentangled CL-robust features be used to train the CL model to mitigate catastrophic forgetting? [7]
- Can neuroscience insights on memory consolidation, reconsolidation, and dataset distillation be integrated into CL approach design? [8]
- Can rehearsal memory be diversified to prevent rehearsal memory overfitting in CL models? [9]

Outline

I. Introduction
   1. Continual Learning
II. Related Work
III. Research Questions
IV. Our Work: A Multidisciplinary Approach
V. Primary Contributions
   1. Adversarially Robust Continual Learning
   2. The Importance of Robust Features in Mitigating Catastrophic Forgetting
   3. Brain-Inspired Continual Learning Robust Feature Distillation and Re-consolidation for Class Incremental Learning
   4. Adversarially Diversified Rehearsal Memory (ADRM): Mitigating Memory Overfitting Challenge in Continual Learning
VI. Conclusions
VII. Future Work

Our Work: A Multidisciplinary Approach

- Our work is the first to integrate insights from Adversarial Machine Learning (AML), Dataset Distillation (or Condensation), Neuroscience, and CL to mitigate catastrophic forgetting.



Outline

1.

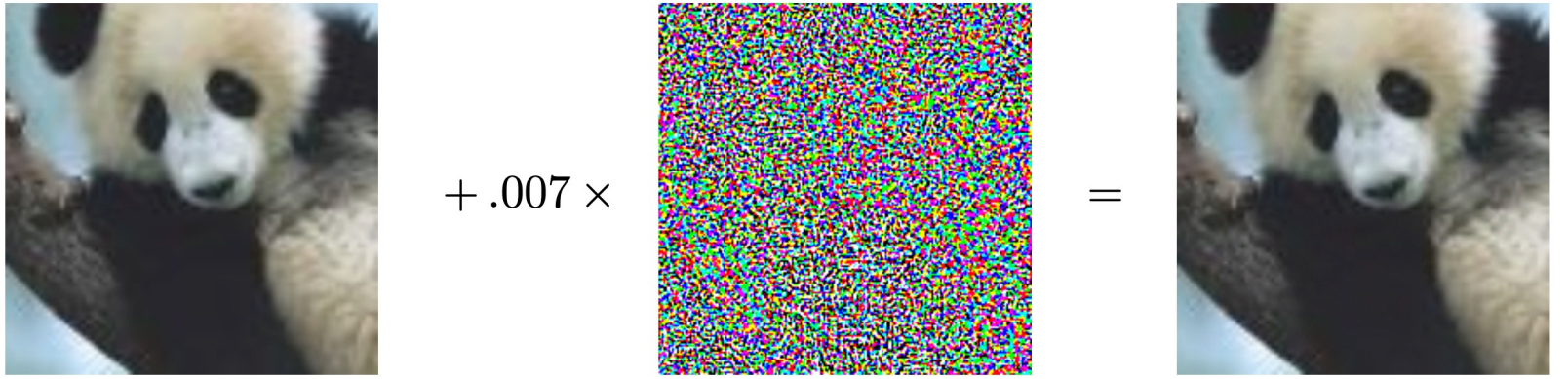
What is the role of features (in the input dataset) in the robustness and mitigation of catastrophic forgetting in the CL?

Hikmat. K, et all, "Adversarially Robust Continual Learning," *2022 International Joint Conference on Neural Networks (IJCNN)*, Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892970.

Adversarial Examples are not Bugs, They are Features

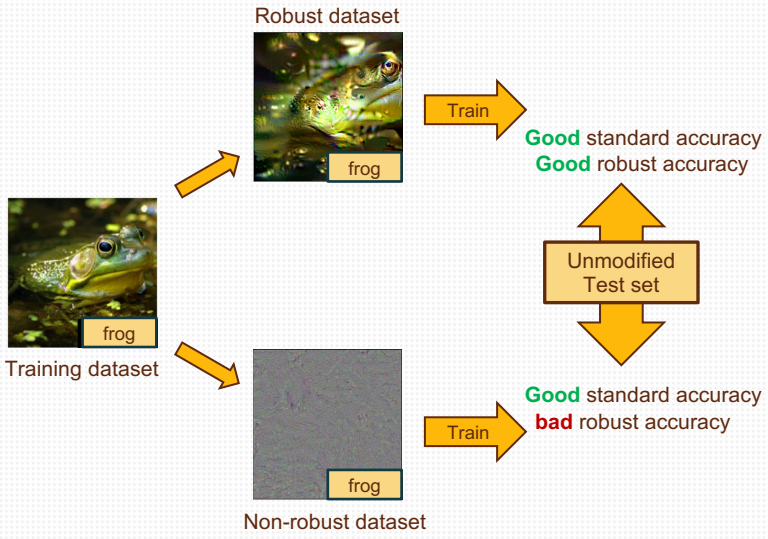
- Fast Gradient Sign Method (FGSM) [10]

$$x_{adv} = x + \epsilon * sign(\Delta_x \, J(\theta, x, y))$$

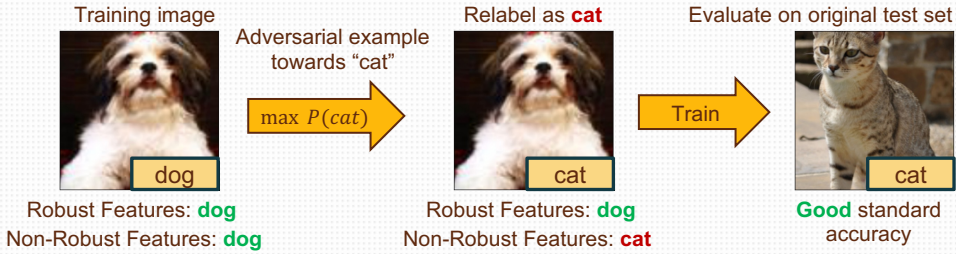


Adversarial Examples are not Bugs, They are Features (Continued)

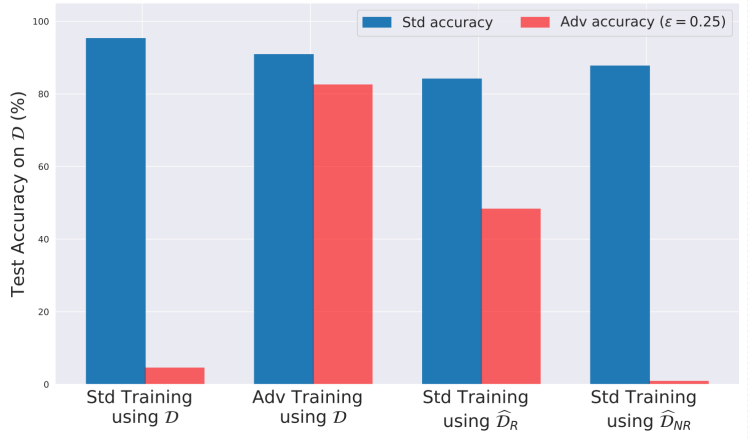
- Adversarial vulnerability is a direct result of models' sensitivity to well-generalizing features in the dataset [11].
- It is possible to disentangle robust from non-robust features in standard datasets [11].



Disentangle features into robust and non-robust



Constructed a dataset that appears mislabeled to humans (via adversarial examples) but results in good accuracy on the original test set



Standard and robust accuracy on the CIFAR-10 test set ($\mathcal{D}$) for models trained with: (i) standard training (on$\mathcal{D}$) ; (ii) standard training on $\widehat{\mathcal{D}_{NR}}$; (iii) adversarial training (on $\mathcal{D}$); and (iv) standard training on $\widehat{\mathcal{D}_R}$.

Adversarial Examples are not Bugs, They are Features (Continued)

- Dataset:
  - Let $\mathcal{D}$ be the dataset consisting of pairs $\{(x_i, y_i)\}$ for i ranging from 1 to N:

$$\{(x_i, y_i)\}_{i=1}^{N} \in \mathcal{D}$$

- Binary classification:
  - The pairs $(x, y) \in \mathcal{X} \times \{\pm 1\}$ are sampled from $\mathcal{D}$ and the label y is taking values in the set $\{\pm 1\}$.
- Goal:
  - The goal is to learn a classifier C that accurately assigns each x the corresponding labels of y:

$$C: \mathcal{X} \to \{\pm 1\}$$

- Feature:
  - Let's define a feature to be a function mapping each $x \in \mathcal{X}$ to the real numbers $\mathbb{R}$.

$$f : \mathcal{X} \to \mathbb{R}$$

- Set of all features:
  - Let $\mathcal{F}$ be a set of all the features, with each $f \in \mathcal{F}$ mapping $x \in \mathcal{X}$ to the real numbers $\mathbb{R}$:

$$\mathcal{F} = \{f : \mathcal{X} \to \mathbb{R}\}$$

Adversarial Examples are not Bugs, They are Features (Continued)

- Assumptions:
  - Let's assume the features in $\mathcal{F}$ are shifted/scaled to be mean-zero and unit-variance.

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[f(x)] = 0$$
$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[f(x)^2] = 1$$

- ρ −useful feature:
  - Feature f is ρ-useful $(\rho > 0)$ if it is correlated with the true label in expectation:

$$\mathbb{E}_{(x,y)\sim D}[y.f(x)] \geq \rho$$

- γ −robust feature:
  - Feature f is γ-robust $(\gamma > 0)$ feature, if under perturbation, f remains γ −useful:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\inf_{\sigma\in\Delta(x)} y.f(x+\sigma)\ \right] \geq \gamma$$

Adversarial Examples are not Bugs, They are Features (Continued)

- **Assumption:**
  - Given the robust classifier C with a robust (learned) set of features $F_C \in \mathcal{F}$.

- Classifier:
  - The classifier $C = (F, w, b)$, with features $F \in \mathcal{F}$, weight vector w, and a scalar bias b. For a given input $x \in \mathcal{X}$:

$$C(x) = \text{sgn}\left(b + \sum_{f \in \text{F}} w_f . f(x)\right)$$

- Standard training:
  - The standard training of the classifier is performed by minimizing a loss function:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{L}_\theta(x, y)] = -\mathbb{E}_{(x,y)\sim\mathcal{D}}[y.\left(b + \sum_{f \in \text{F}} w_f . f(x)\right)$$

Adversarial Examples are not Bugs, They are Features (Continued)

- Adversarial training:
  - Robust training of the classifier is performed by minimizing a loss function ($\mathcal{L}$):

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[\max_{\sigma\in\Delta(x)} \mathcal{L}_\theta(x+\delta, y)]$$

- Robustified dataset ($\widehat{\mathcal{D}_R}$):
  - Aim is to construct $\widehat{\mathcal{D}_R}$ via a one-to-one mapping $x \rightarrow x_r$ from the original training set $\mathcal{D}$:

$$\mathbb{E}_{(x_r,\, y)\sim\widehat{\mathcal{D}_R}}[y.f(x)] = \begin{cases} \mathbb{E}_{(x,y)\sim\mathcal{D}}[y.f(x)] & \text{if } f \in \mathcal{F}_C \\ 0 & \text{otherwise} \end{cases}$$

- Optimization objective:
  - x_r is obtained by minimizing the below:

$$\min_{x_r} \| g(x_r) - g(x) \|_2^2$$

- Starting $x_0 \sim \mathcal{D}$ is randomly sampled.
- g is the mapping from $x \in \mathcal{X}$ to the representation layer.
- Optimized via GD in the input space.
- We used the logit layer (pre-softmax layer).

Adversarial Examples are not Bugs, They are Features (Continued)

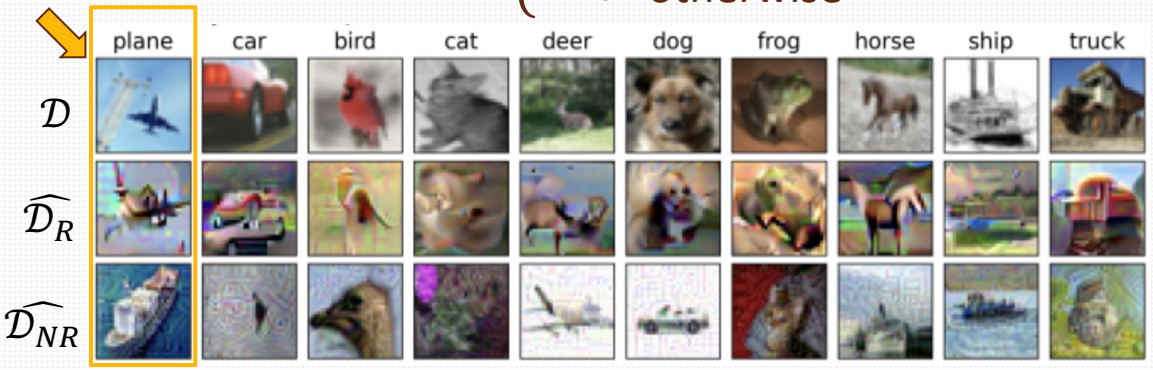
- Non-robustified dataset ($\widehat{\mathcal{D}_{NR}}$):
  - Construct $\widehat{\mathcal{D}_{NR}}$ by adding a small perturbation to x to be classified as t by a standard model:

$$x_{adv} = \min_{\|x'-x\| \leq \varepsilon} L_C(x', t)$$

where $\mathcal{L}_C$ is the loss under a standard (non-robust) classifier C, ϵ is strength of perturbation.

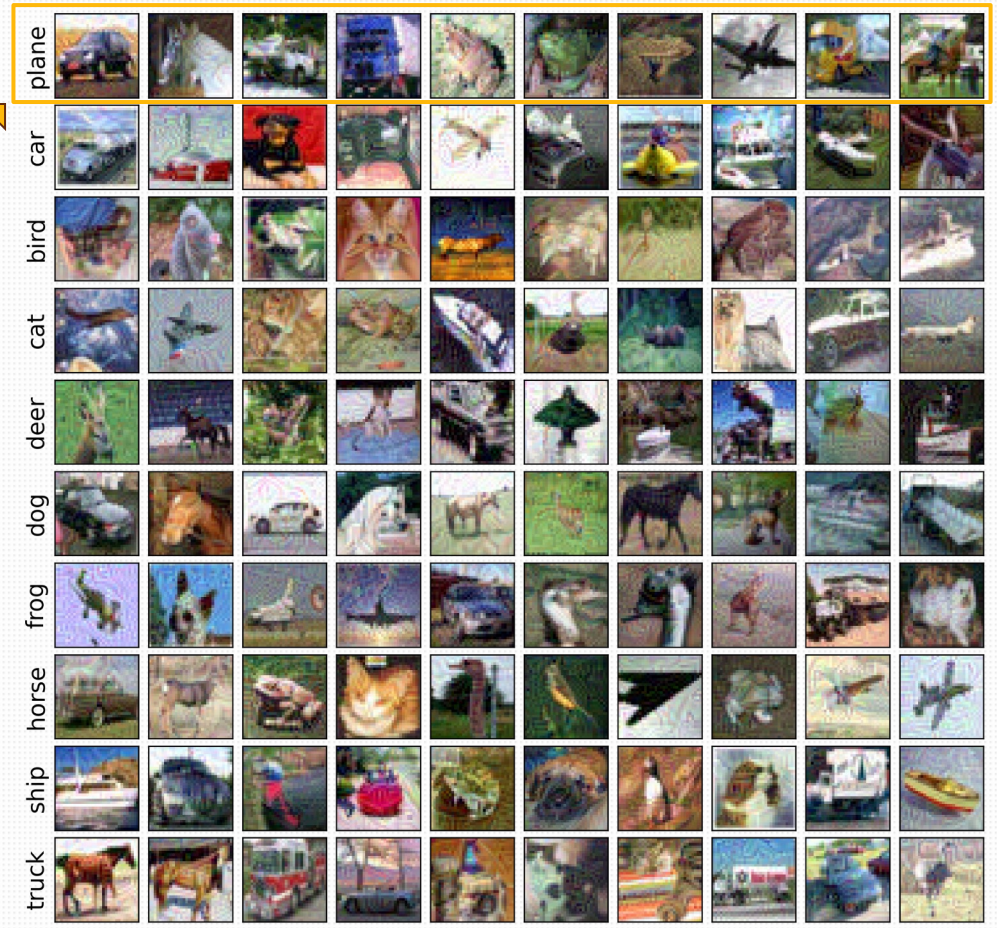
- Non-robustified pairs:
  - (x_{adv}, t) make up the new training set.

$$\mathbb{E}_{(x,y)\sim\widehat{\mathcal{D}}_{NR}}[y.f(x)] = \begin{cases} > 0 & \text{if } f \text{ non-robustly useful under } \mathcal{D}, \\ \approx 0 & \text{otherwise} \end{cases}$$



More Examples For Robust ($\mathcal{D}_R$) and Non-Robust ($\mathcal{D}_{NR}$) Features

Adversarially robust features
(Samples from robustified CIFAR10 ($\mathcal{D}_R$) dataset)



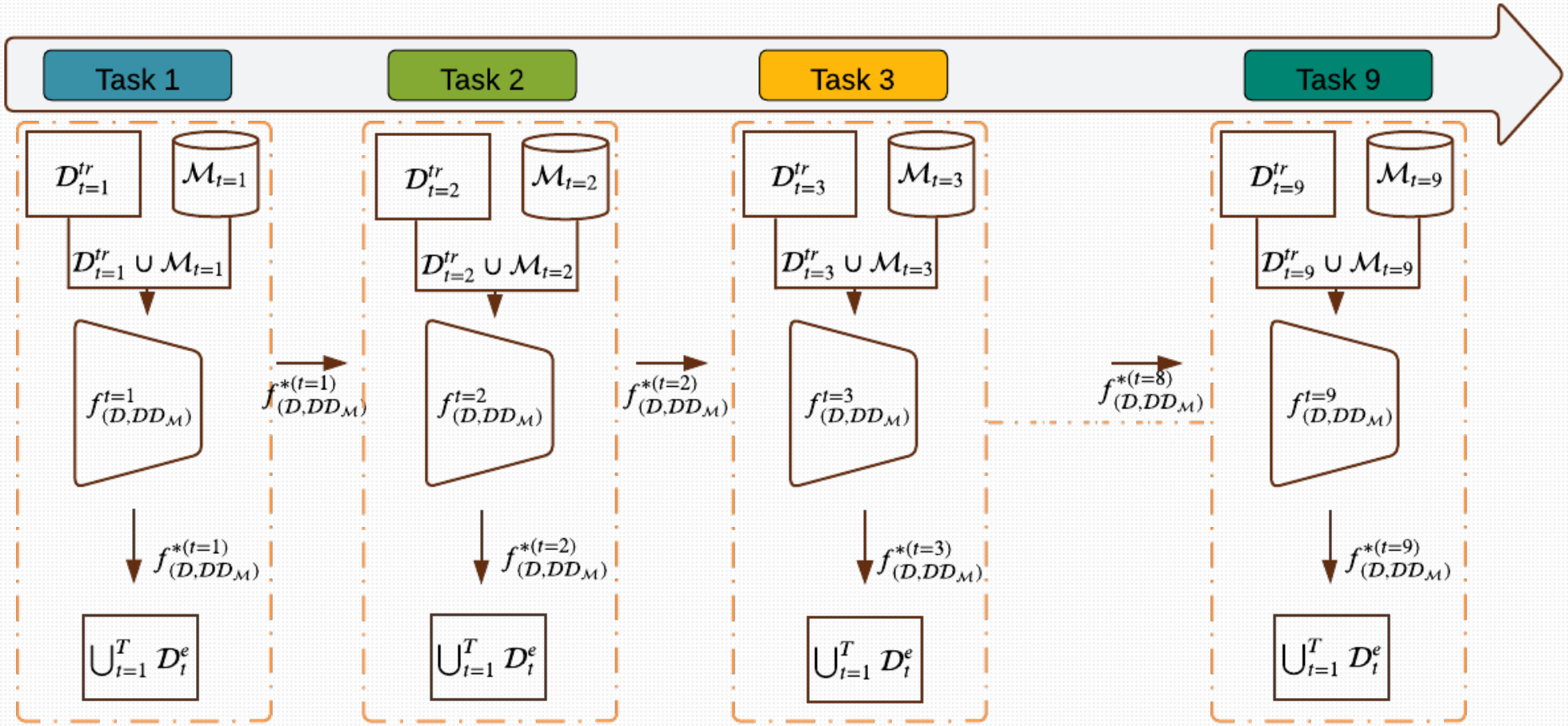
Adversarial non-robust features
(Samples from non-robustified CIFAR10 ($\mathcal{D}_{NR}$) dataset)

Training: Five Baseline CL Models Using Rehearsal Strategy

- Standard model ($f_{(\mathcal{D},\mathcal{DD})}$; Training dataset, Rehearsal Memory)
  - Trained on standard CIFAR10
  - Replay buffer consisted of samples from standard CIFAR10
- Standard model with robust replay ($f_{(\mathcal{D},\mathcal{DR})}$)
  - Trained on standard CIFAR10 dataset
  - Replay buffer consisted of samples from standard CIFAR10 and robustified CIFAR10
- Standard model with non-robust replay ($f_{(\mathcal{D},\mathcal{DN})}$)
  - Trained on standard CIFAR10 dataset
  - Replay buffer consisted of samples from standard and non-robustified CIFAR10
- Robust model with standard replay ($f_{(\mathcal{R},\mathcal{RD})}$)
  - Trained on robustified CIFAR10
  - Replay buffer consisted of samples from standard and robustified CIFAR10
- Non-robust model with standard replay ($f_{(\mathcal{N},\mathcal{ND})}$)
  - Trained on non-robustified CIFAR10
  - Replay buffer consisted of samples from non-robustified and standard CIFAR10

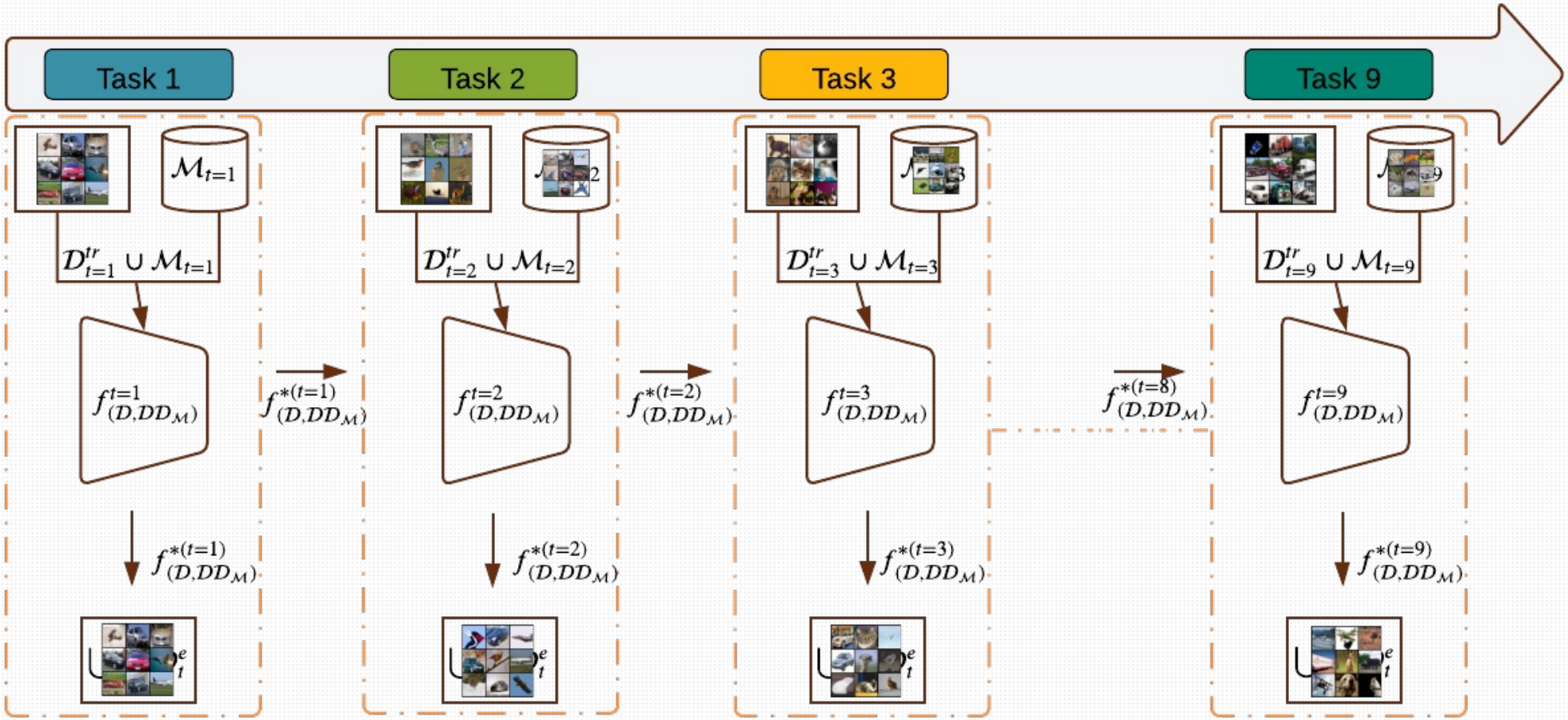
Training Baselines

- The standard baseline $f_{(\mathcal{D},\mathcal{DD})}$ has been trained as follow:



Training Baselines

- The standard baseline $f_{(\mathcal{D},\mathcal{DD})}$ has been trained as follow:



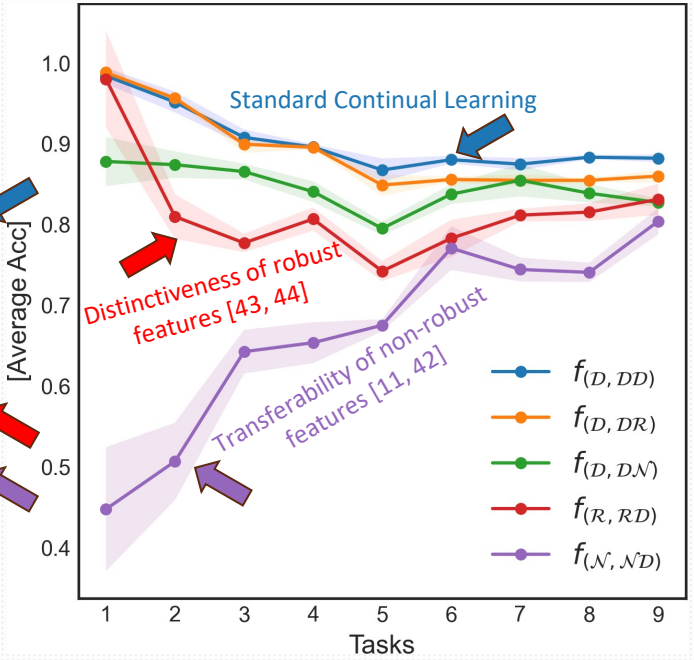
Results: Standard Continual Learning

- We presented an empirical study demonstrating the importance of input features in CL.
- The average accuracy is computed as

$$Avarege\ Acc = \frac{1}{T}\sum_{i=1}^{T} R_{T,i}$$

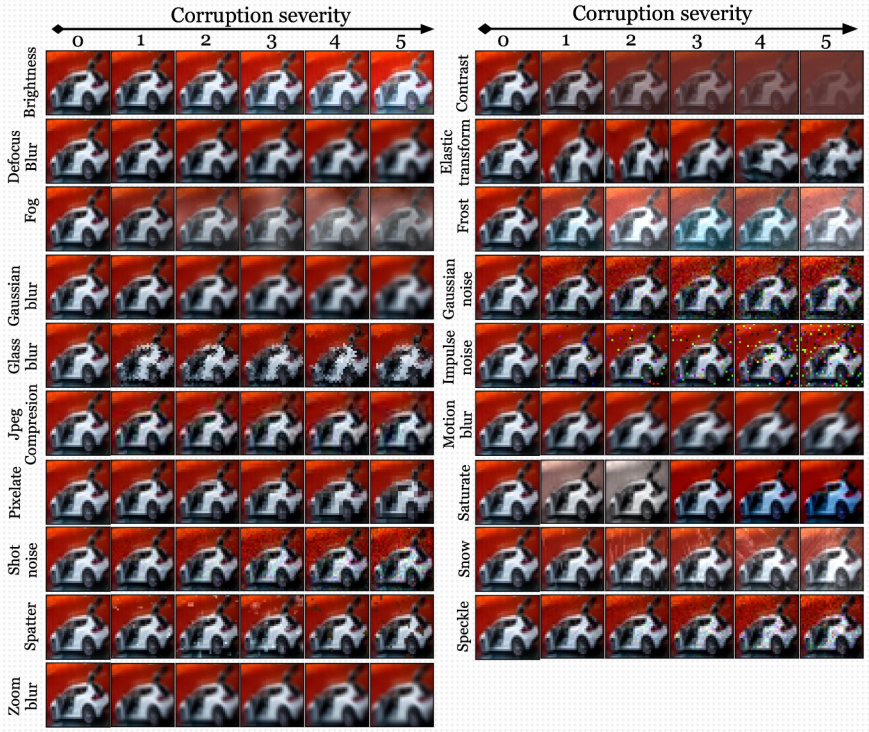
Model	**Training set**	**Replay buffer** $(size = 16000)$	**Average accuracy**
$f_{(\mathcal{D},\mathcal{DD})}$	CIFAR10 ($\mathcal{D}$)	CIFAR10 ($\mathcal{D}$) + CIFAR10 ($\mathcal{D}$)	$88.20\pm_{0.48}$
$f_{(\mathcal{D},\mathcal{DR})}$	CIFAR10 ($\mathcal{D}$)	CIFAR10 ($\mathcal{D}$) + Robustified CIFAR10 ($\mathcal{R}$)	$85.99\pm_{0.77}$
$f_{(\mathcal{D},\mathcal{DN})}$	CIFAR10 ($\mathcal{D}$)	CIFAR10 ($\mathcal{D}$) + Non-Robustified CIFAR10 ($\mathcal{N}$)	$82.72\pm_{0.49}$
$f_{(\mathcal{R},\mathcal{RD})}$	Robustified CIFAR10 ($\mathcal{R}$)	Robustified CIFAR10 ($\mathcal{R}$) + CIFAR10 ($\mathcal{D}$)	$83.12\pm_{1.9}$
$f_{(\mathcal{N},\mathcal{ND})}$	Non-Robustified CIFAR10 ($\mathcal{N}$)	Non-Robustified CIFAR10 ($\mathcal{N}$) + CIFAR10 ($\mathcal{D}$)	$80.40\pm_{1.6}$

The average accuracies of all 5 models. $\mathcal{D}$ =Standard CIFAR10, $\mathcal{D}_R$ = Robustified CIFAR10 and $\mathcal{D}_{NR}$ =Non-robustified CIFAR10. In model $f(X, YZ)$: the first entry represents the training data set, the second set of letters denotes the replay buffer datasets sampled equally.



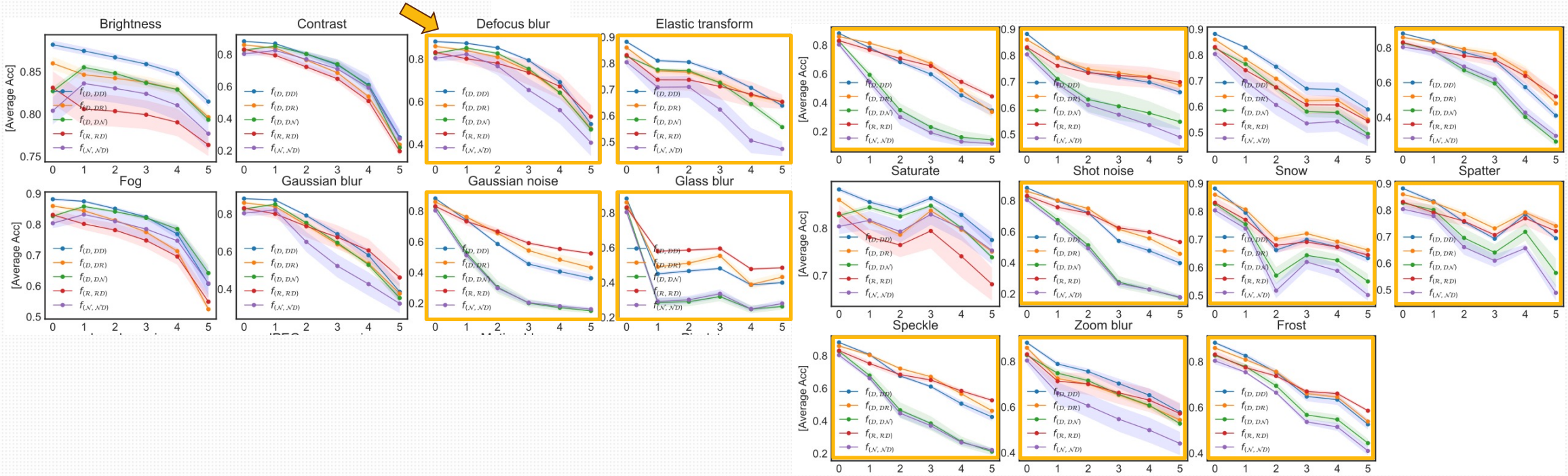
CIFAR10-C (Corrupted) Dataset

- The CIFAR10-C dataset is a variant of the original CIFAR10 dataset, characterized by the inclusion of corrupted images designed to challenge and evaluate the robustness of models [21].
  - Corrupted images are generated by applying nineteen different types of corruption, such as brightness, fog, and others.
  - We used the CIFAR-10-C dataset to evaluate the robustness and catastrophic forgetting of five benchmark CL models.



Evaluating Against Common Corruptions

- The model trained and rehearsed with robust features performed relatively best (outperforming others in 12 out of 19) in various noisy conditions.
- Conversely, the model utilizing non-robust features performed the worst in noisy conditions, underscoring the important role of the features.
- Emphasizes the critical role of features in mitigating the catastrophic forgetting and robustness of CL models, especially in noisy conditions.



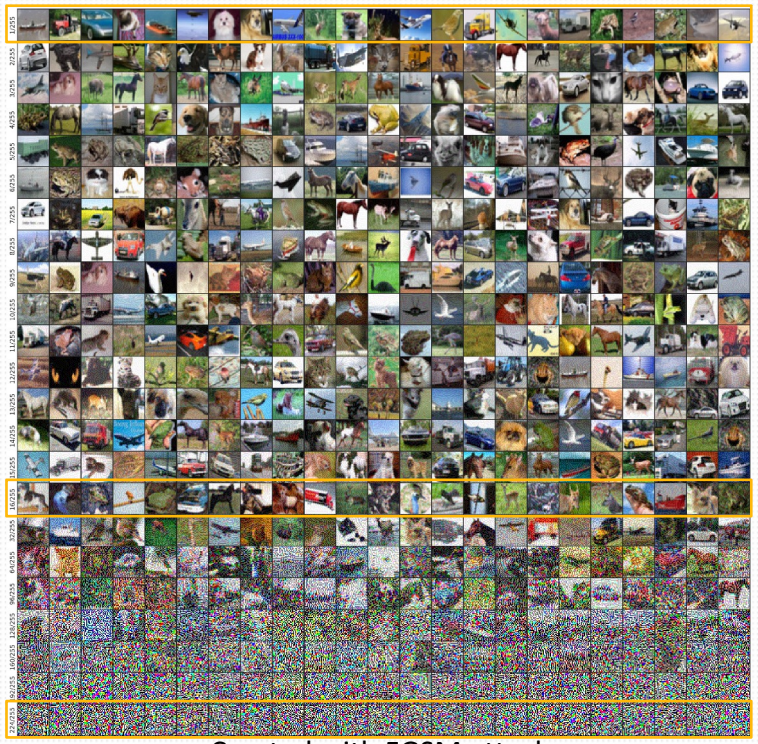
Adversarially Perturbed CIFAR10 Datasets

- We created 32 adversarially perturbed CIFAR-10 (eval-sets) using FGSM [10] and PGD to access the role of the features in catastrophic forgetting of the CL models in adversarial conditions [46].
  - with varying epsilon values i.e., $\epsilon = \{\frac{1}{255}, \frac{2}{255}, \frac{3}{255}, \dots, \frac{192}{255}, \frac{224}{255}\}$

$\epsilon = \frac{1}{255}$

$\epsilon = \frac{16}{255}$

$\epsilon = \frac{244}{255}$



Created with FGSM attack

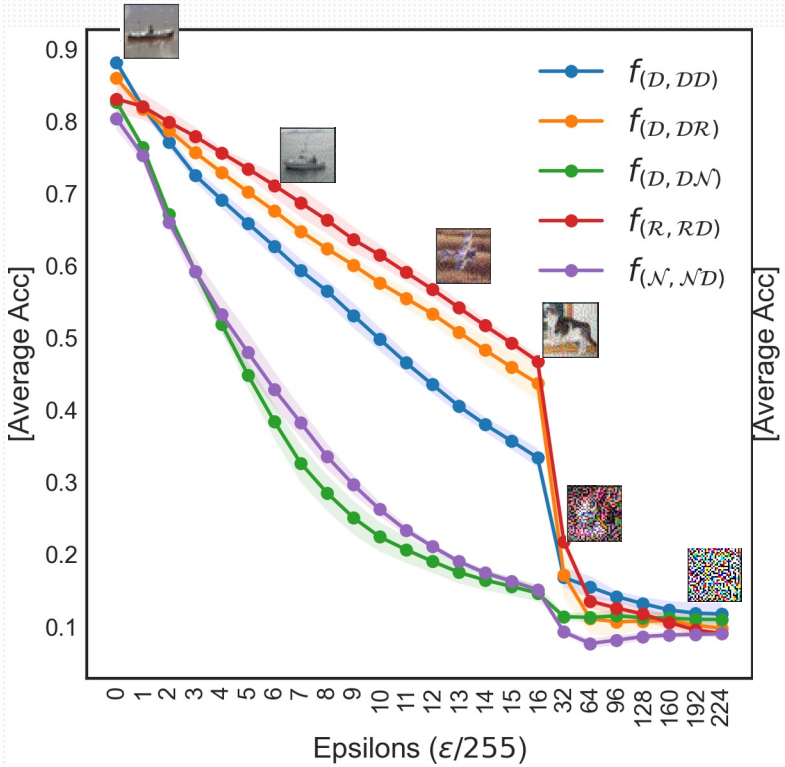
Created with PGD−L_2 attack

Low adversarial noise

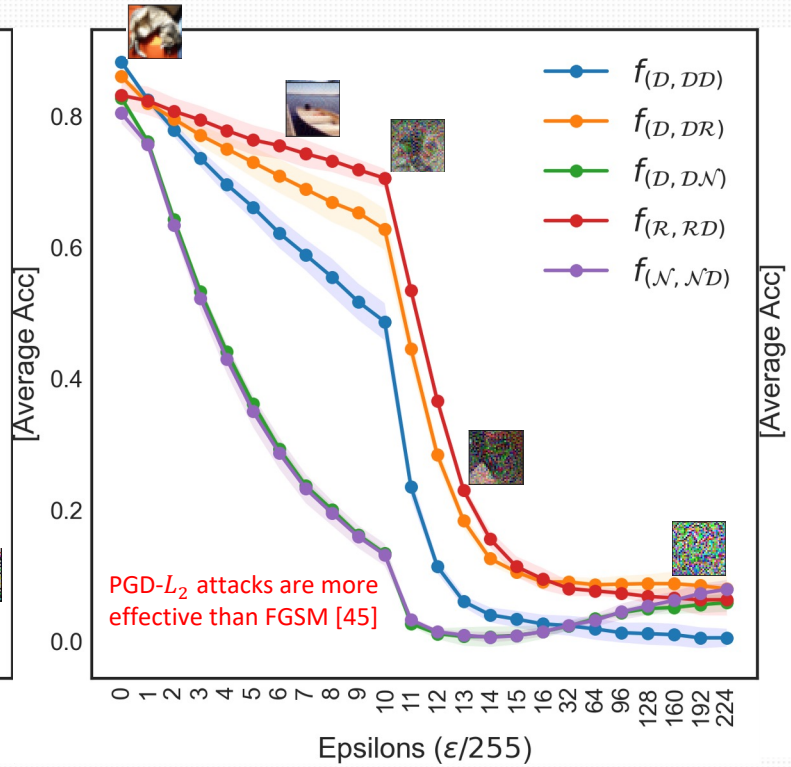
High adversarial noise

Evaluating Against Adversarial Attacks

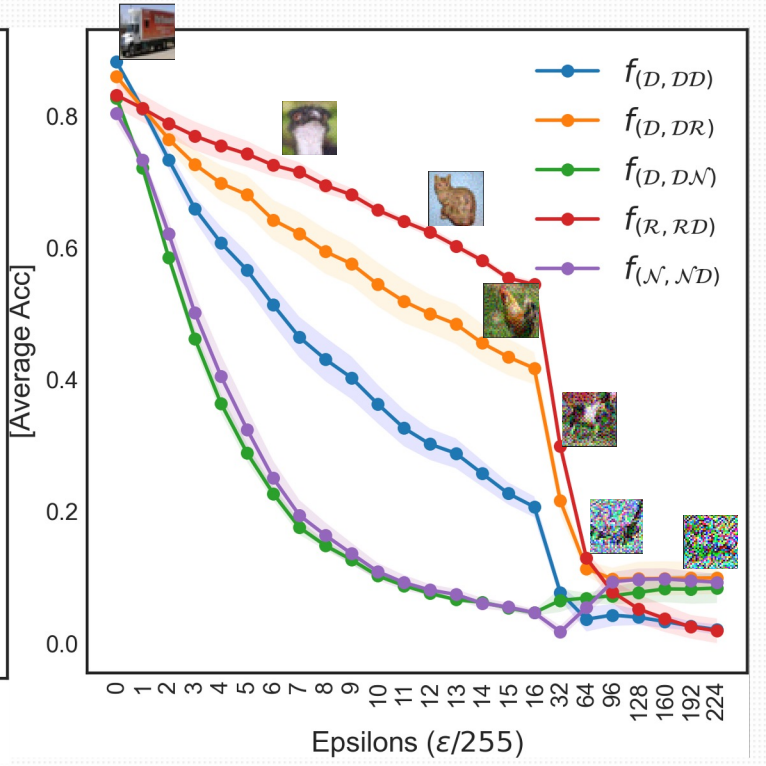
- The robust models $f_{(\mathcal{R},\mathcal{RD})}$ and $f_{(\mathcal{D},\mathcal{DR})}$ achived the highest level of robustness and less catastrophic forgetting.



Catastrophic forgetting under FGSM



Catastrophic forgetting under PGD-L_2

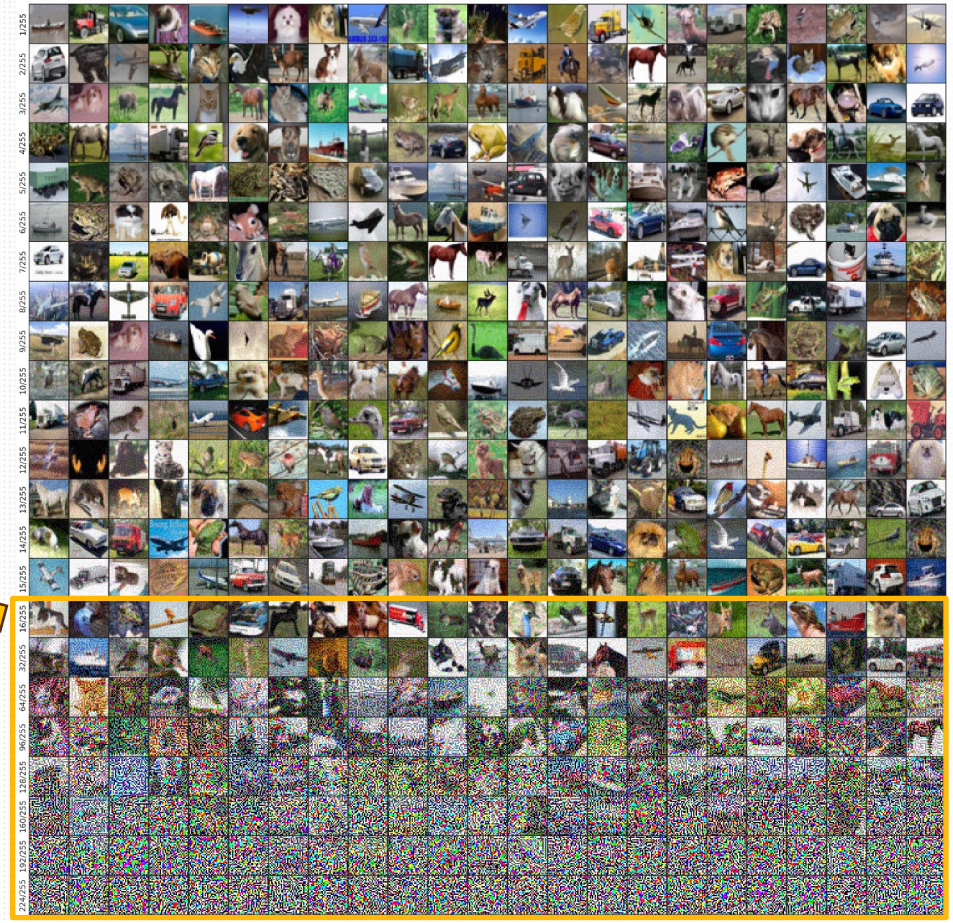


Catastrophic forgetting under PGD-L_{inf}

Adversarially Perturbed CIFAR10 Datasets

- A drastic decrease in accuracy has been noted between $\frac{16}{255} \leq \epsilon \leq \frac{244}{255}$ for FGSM and $\frac{10}{255} \leq \epsilon \leq \frac{244}{255}$ for PGD$-L_2$.

FGSM: $\epsilon \geq \frac{16}{255}$



Created with FGSM attack

Low adversarial noise

PGD: $\epsilon \geq \frac{10}{255}$

High adversarial noise

Created with PGD$-L_2$ attack

Conclusions

- Our experiments reveal that models trained on standard features achieved higher accuracy on clean data compared to those trained with robust or non-robust features.
- However, in noisy and adversarial conditions, the models trained using standard and non-robust features performed poorly compared to the model trained using robust features.
- The model trained using non- robust features performed the worst in noisy conditions and under adversarial attacks.
- Our conclusion highlights the importance of features in preventing catastrophic forgetting and defining the robustness of CL models.

Outline

2.

Can disentangled CL-robust features be used to train the CL model to mitigate catastrophic forgetting?

Hikmat. K, et all, "Adversarially Robust Continual Learning," *2022 International Joint Conference on Neural Networks (IJCNN)*, Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892970.

Role of CL Robust Feature in Mitigating Catastrophic Forgetting

- We hypothesize that the features in the input dataset contribute to catastrophic forgetting.
  - However, if the model is provided with CL robust features, it can mitigate catastrophic forgetting.
  - To extract CL robust features, we presuppose access to a pre-trained oracle CL model.
  - Utilizing this pre-trained oracle CL model, we disentangle CL robust features to create CL robust dataset ($\mathcal{D}_R$)
  - The baseline CL approaches are then trained on both the $\mathcal{D}_R$ and standard dataset ($\mathcal{D}$).
  - We observed that when the model is provided with CL robust features (i.e., $\mathcal{D}_R$), it achieved higher performance compared to using standard features (i.e., $\mathcal{D}$).

CL Robust Dataset ($\mathcal{D}_R$)

- ρ −useful features:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[y.\, f(x)] \geq \rho$$

- ρ −useful features in CL:
  - Let f_t be a CL model at task t; then ρ-useful features in the context of CL can be defined as follows:

$$\mathbb{E}_{(x,y)\sim(\mathcal{D}_{t-\tau})}[y.\, f_t(x)] \geq \rho_t \qquad \tau = 1, \dots, t$$

- **Assumption:**
  - Let f_t^o be oracle CL model, we construct CL-robust ($\mathcal{D}_R$) dataset such that:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}_R}[f_t(x_{t-\tau}).\, \mathrm{y_{t-\tau}}] = \begin{cases} \mathbb{E}_{(x,y)\sim\mathcal{D}}[f_t^o(x_{t-\tau}).\, y_{t-\tau}] & \text{if } f \in \mathcal{F}^o, \tau = 1, \dots, t \\ 0 & \text{otherwise} \end{cases}$$

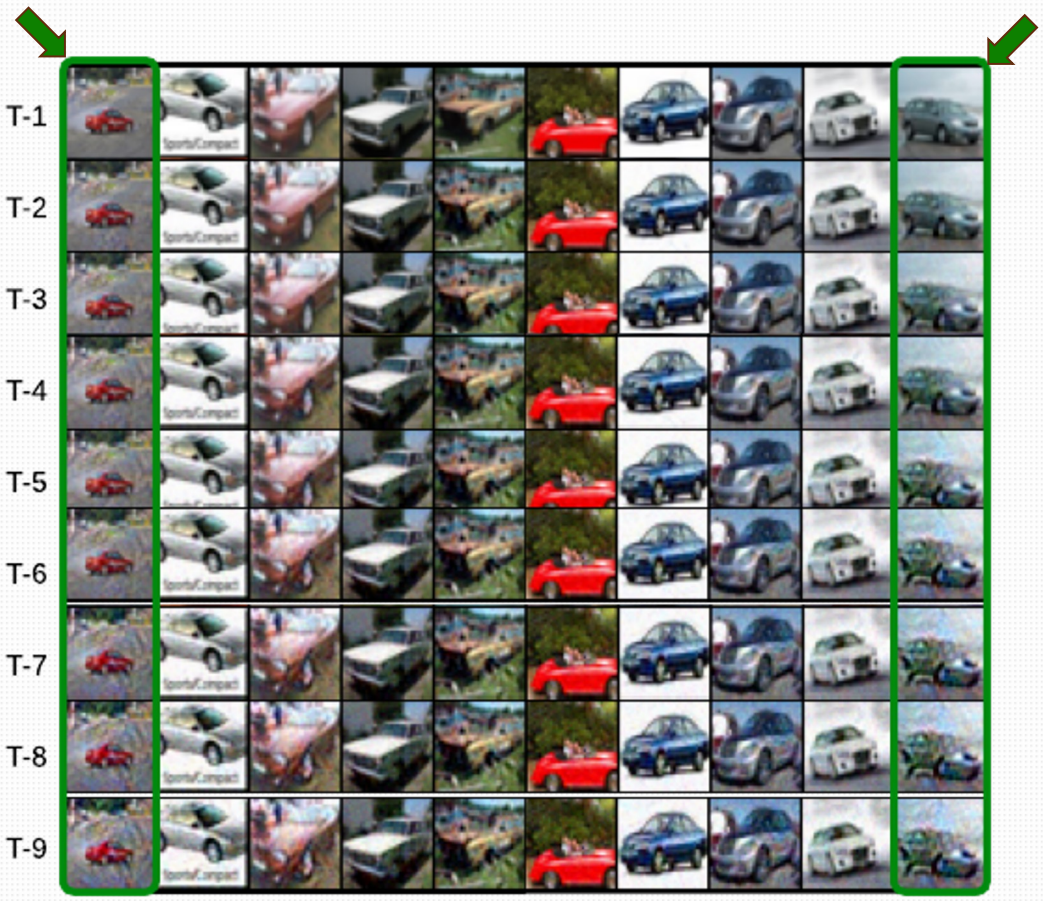
- Optimization objective:
  - f_t^o (pretrained oracle model at task t) is used to create the $\mathcal{D}_R$ by minimizing the below

$$x_{cl} = \min_{x_z \in [0,1]^d} \| f_t^o(x_z) - f_t^o(x_{t-\tau}) \|_2^2, \qquad \tau = 1, \dots, t., \qquad x_z \sim \mathcal{D}$$

Examples from CL-Robust Dataset ($\mathcal{D}_R$)

- Task-wise CL robust features for the airplane and car classes, respectively.

a) CL robust features of airplane class



a) CL robust features of car class

Training Baselines and Results

- We trained the CL five baseline on both the standard $\mathcal{D}$ and $\mathcal{D}_R$ datasets.
  - Multi-task or Joint learning [1]
  - Finetune [1]
  - Elastic Weight Consolidation (EWC) [16]
  - Rehearsal-based approach (Replay) [15]
  - Dynamically Expanding Representation (DER) [17]
  - PODNet [18]

CL Method	Split-CIFAR10		
	Nine Tasks	Five Tasks	Two Tasks
Multi-Task	94.8±0.61		
Finetune	11.11±0.68	19.6±0.81	45.31±0.54
Finetune-CLR	**13.0**±0.50	**21.03**±1.01	**51.48**±0.63
EWC	19.39 ±1.1	20.05 ±0.65	46.73 ±1.16
EWC-CLR	**25.71** ±1.0	**28.33** ±0.70	**54.07** ±1.95
Replay	55.06 ±1.9	62.18 ±1.30	63.9 ±0.97
Replay-CLR	**75.86** ±1.2	**78.29** ±0.93	**80.77** ±1.09
DER	52.42 ±1.5	65.65 ±0.72	71.6 ±0.94
DER-CLR	**70.13** ±1.1	**81.74** ±0.65	**87.32** ±0.90
PODNet	55.14 ±1.2	59.62 ±0.74	80.09 ±1.16
PODNet-CLR	**72.71** ±1.87	**76.94** ±0.63	**85.82** ±1.23

- CL baselines (suffixed with "-CLR") are trained on $\mathcal{D}_R$ achieved higher average accuracy compared to the same CL baselines trained on standard $\mathcal{D}$ datasets.

Conclusions and Limitations

- We demonstrated the CL-robust features played a crucial role in mitigating catastrophic forgetting.

- Observations indicated that CL baselines trained on the CL robust dataset ($\mathcal{D}_R$) achieved higher average accuracy compared to those trained on a standard dataset ($\mathcal{D}$), emphasizing the importance of CL robust features.

- **Limitation**: A key assumption in our study is the availability of a pre-trained CL Oracle model for creating the CL-robust dataset.

Outline

3.

Can neuroscience insights on memory consolidation, reconsolidation, and dataset distillation be integrated into CL approach design?

Hikmat. K, et all, "Adversarially Robust Continual Learning," *2022 International Joint Conference on Neural Networks (IJCNN)*, Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892970.

Neuroscience: Memory Consolidation and Re-Consolidation

- The mammalian brain replays waking experiences in a distilled and compressed form, not in the original form, to facilitate memory consolidation [28, 29]. [Rehearsing the Distilled Memory Samples]

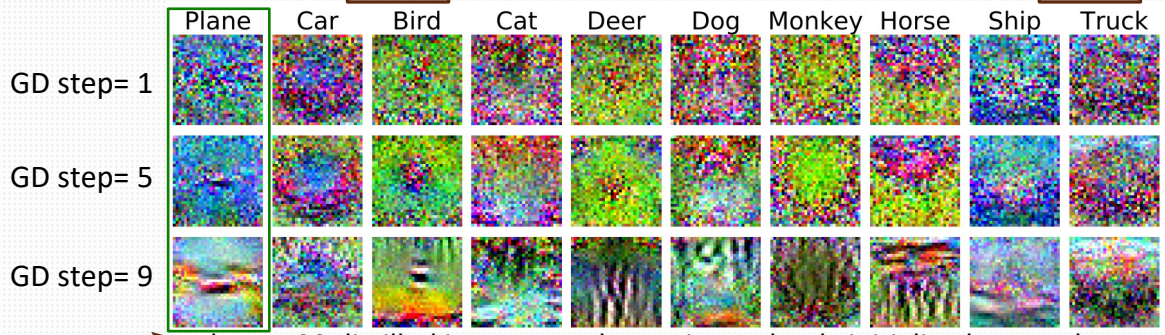
- Novel waking experiences have a pronounced influence on previous experiences [29, 30]. [Re-Distillation of the Distilled Memory Samples]

- The mammalian brain does not reenact waking experiences in a significantly accelerated time scale (not at the original time scale or in form) to enable memory consolidation during dreams [30, 31].

Dataset Distillation or Condensation

- Dataset Distillation or Condensation [12]
  - Distill the features of the large dataset into a smaller synthetic dataset so that, when the same model is trained on both the original large dataset and the synthetic dataset, it achieves comparable performance.
  - Large dataset $\mathcal{D}_{tr} = \{(x_i, y_i)\}_{i=1}^{N}$
  - Small dataset $\mathcal{D}_s = \{(x'_j, y'_j)\}_{j=1}^{N'}, \quad N' \ll N$
  - Such that: $\mathbb{E}_{(x,y)\sim\mathcal{D}_{te}}[m(f(x;\theta^{(*)}), y)] \cong \mathbb{E}_{(x,y)\sim\mathcal{D}_{te}}[m(f(x;\theta'^{(*)}), y)]$

$\theta^{(*)}$ optimized on $\mathcal{D}_{tr}$
$\theta'^{(*)}$ optimized on $\mathcal{D}_s$



These 100 distilled images used to train randomly initialized networks to 36.79% average test accuracy on CIFAR10.

- Neural Architecture Search [13, 14, 34]
  - Synthesized samples are used to accelerate the neural architecture search.
- Continual Learning:
  - Synthetic samples are suitable for creating efficient memory in CL.

Robust Rehearsal: CL Formulation

- ρ-useful features:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[y.f(x)] \geq \rho$$

- ρ-useful features in CL:
  - **Assumption**: We assumed f_t^o is oracle CL pre-trained model at each task t; then ρ-useful features were defined as follows.

$$\mathbb{E}_{(x,y)\sim(\mathcal{D}_{t-\tau})}[y.f_t^o(x)] \geq \rho_t \qquad \tau = 1,\dots,t$$

- **Eliminate the Assumption**: Use f_t (current CL model) instead of f_t^o (oracle CL model):
  - Let f_t be a CL model at task t; then ρ-useful can be defined as follows.

$$\mathbb{E}_{(x_t,y_t)\sim\mathcal{D}_t}[y_t, f_t(x_t)] \geq \rho_t$$

- γ −robust features in CL:
  - Feature f_t is CL-robust feature if it remains a γ_t −robustly useful for task $(t-\tau)$, with $(0 \leq \tau \leq t)$.

$$\mathbb{E}_{(x_{t-\tau},y_{t-\tau})\sim\mathcal{D}_{t-\tau}}[y_{t-\tau}, f_t(x_{t-\tau})] \geq \gamma_t, \quad \forall\tau = 0,\dots,\mathrm{t}-1$$

Robust Rehearsal: CL Formulation (Continued)

- Distillation of CL robust features ($\mathrm{x}_{\mathrm{clr},\ \mathrm{t}}^{[\mathrm{c}]}$) for c-class using CL model f_{θ_t} at task t.

$$\mathrm{x}_{\mathrm{clr},\ \mathrm{t}}^{[\mathrm{c}]} = \underset{\mathrm{x}}{\operatorname{argmin}}\, \mathcal{L}(\mathrm{x}; \mathrm{x}_{\mathrm{t}}^{[\mathrm{c}]}, \mathrm{f}_{\theta_{\mathrm{t}}})$$

- The loss ($\mathcal{L}$) is consisted of three terms.

$$\mathcal{L} = \mathcal{L}_i + \mathcal{L}_f + \mathcal{L}_p$$

- $\mathcal{L}_i$ is a loss in input space, can be defined as below.

$$\mathcal{L}_i\left(x; x_t^{[c]}\right) = \alpha \left\| x - x_t^{[c]} \right\|_2^2$$

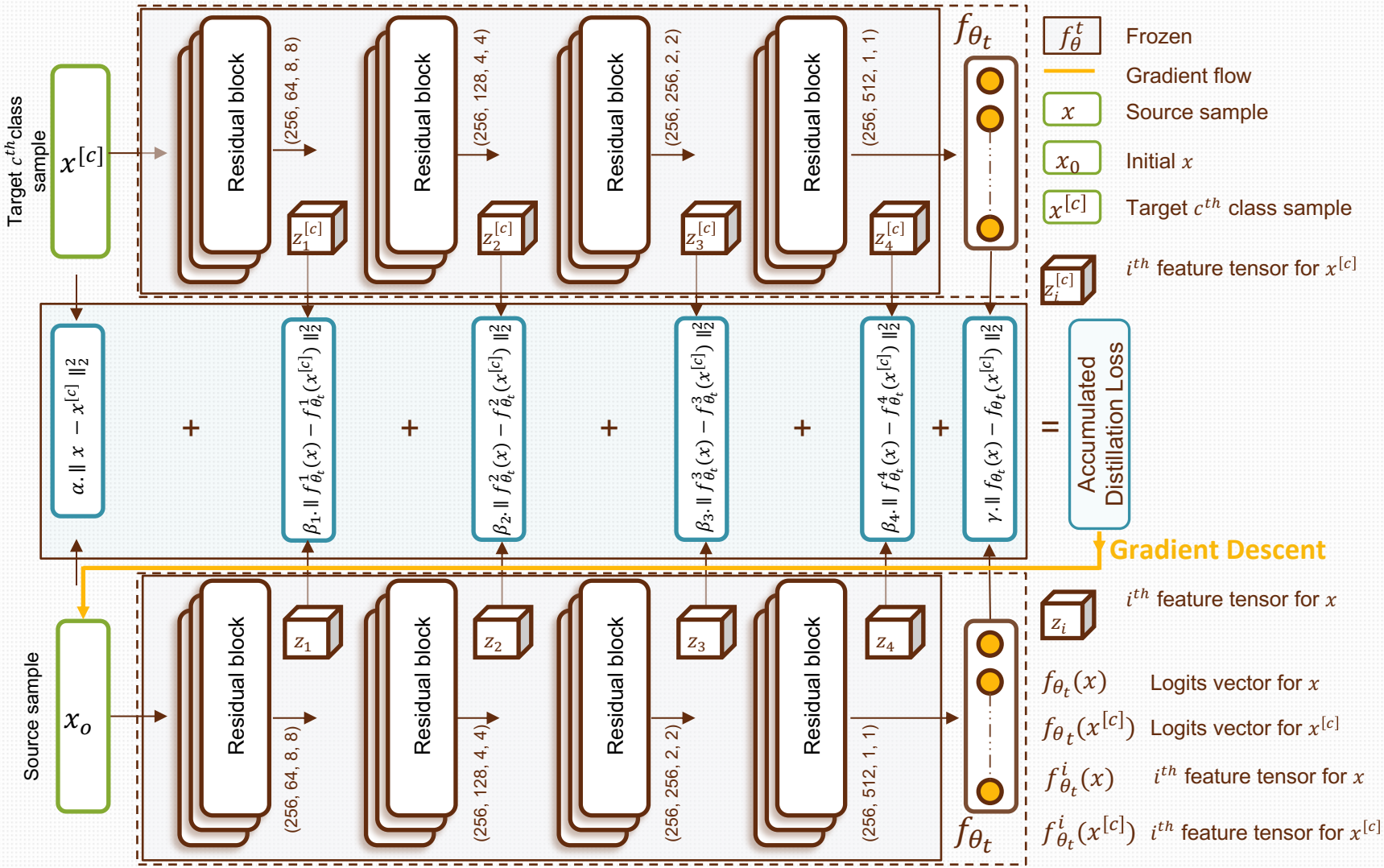
- $\mathcal{L}_f$ is a loss in features space, can be defined as below.

$$\mathcal{L}_f\left(x; x_t^{[c]}\right) = \sum_{i=i}^{n} \beta_i \left\| f_{\theta_t}^i(x) - f_{\theta_t}^i(x_t^{[c]}) \right\|_2^2$$

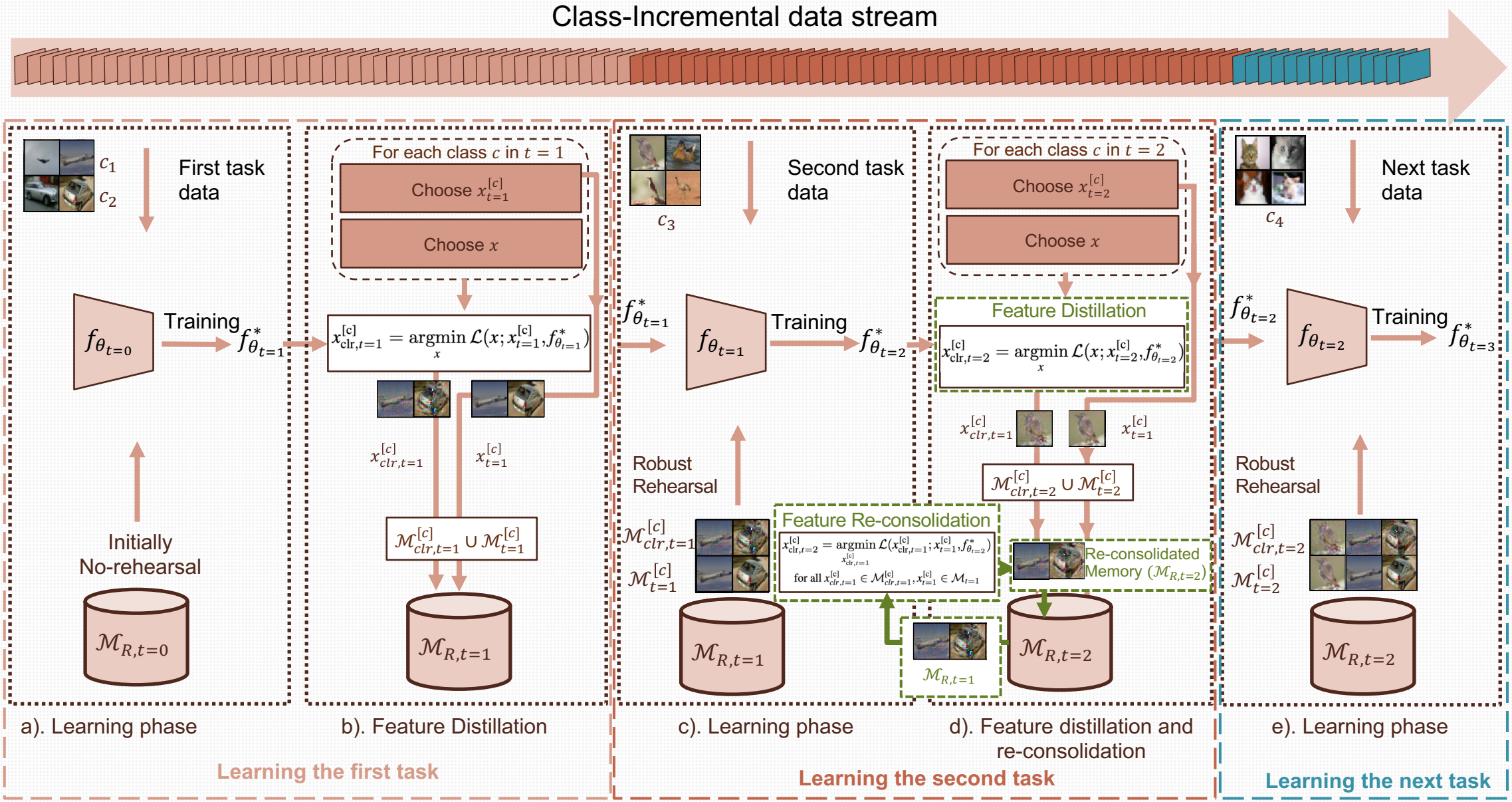
- $\mathcal{L}_p$ is a loss in prediction space, can be defined as below.

$$\mathcal{L}_p\left(x; x_t^{[c]}\right) = \gamma \left\| f_{\theta_t}(x) - f_{\theta_t}(x_t^{[c]}) \right\|_2^2$$

Robust Rehearsal: Three-Level Distillations



Robust Rehearsal: A Proposed Framework

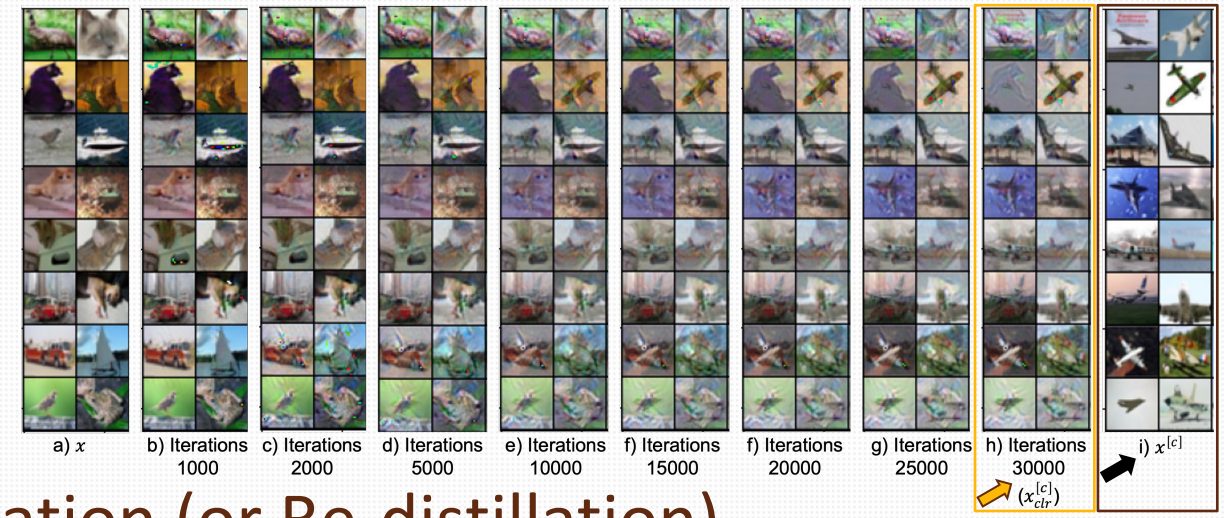


CL-Robust Samples: A Dream-like Features

- Feature Distillation

$$x^{[c]}_{clr,\,t} = \underset{x}{\operatorname{argmin}}\, \mathcal{L}(x, x^{[c]}_t, f_{\theta_t})$$

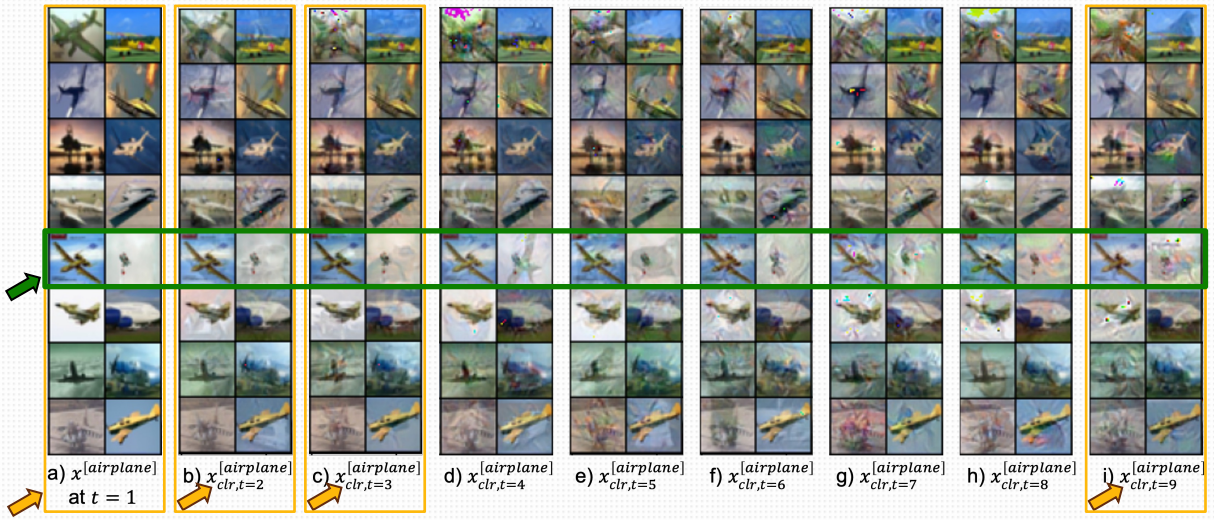
$$x \xrightarrow{CL-Robustified} x^{[airplane]}_{clr,t}$$



- Feature Re-consolidation (or Re-distillation)

$$x^{[c]}_{clr,\,t} = \underset{x^{[c]}_{clr,t-1}}{\operatorname{argmin}}\, \mathcal{L}(x^{[c]}_{clr,t-1}, x^{[c]}_t, f_{\theta_t})$$

$$x^{[airplane]}_{clr,t-1} \xrightarrow{Re-Consolidated} x^{[airplane]}_{clr,t}$$



Results on Benchmark Datasets

- Average accuracy for the CIFAR10, CIFAR100 datasets in class incremental setting.
- The average accuracy is computed as $Avarege\ Acc = \frac{1}{T}\sum_{i=1}^{T} R_{T,i}$

CL Method	Split-CIFAR10		
	9 steps	5 steps	2 steps
Joint		94.8±0.61	
Fine-tune	11.11±0.68	18.54±0.71	45.31±1.6
Replay [20]	55.06 ±1.90	62.18 ±1.30	63.9 ±0.97
Replay-CLR [39]	75.86 ±1.20	78.29 ±0.93	80.77 ±1.09
Replay-RR	**77.34** ±1.43	**80.12** ±1.39	**84.11** ±1.66
DER [30]	52.42 ±1.50	65.65 ±0.72	71.60 ±0.94
DER-CLR [39]	70.13 ±1.10	81.74 ±0.65	87.32 ±0.90
DER-RR	**72.39** ±1.78	**74.16** ±0.97	**83.92** ±0.84
PODNet [29]	55.14 ±1.20	59.62 ±0.74	80.09 ±1.16
PODNet-CLR [39]	72.71 ±1.52	76.94 ±0.63	**85.82** ±1.23
PODNet-RR	**74.23** ±0.76	**77.88** ±0.89	84.54 ±1.25

Split-CIFAR10 Dataset

CL Method	Split-CIFAR100 B0			Split-CIFAR100 B50	
	5 steps	10 steps	20 steps	5 steps	10 steps
Joint			84.45 ±2.36		
Fine-tune	27.8 ±2.29	22.4 ±1.91	19.5 ±2.13	13.49 ±1.98	11.34 ±2.71
iCaRL [19]	68.12 ±2.25	66.74 ±2.79	64.01±1.92	59.29 ±2.32	50.82 ±1.82
iCaRL-RR	**72.87** ±1.34	**73.66** ±1.24	**70.91** ±1.53	**66.24** ±1.09	**59.31** ±1.37
PodNET [29]	69.54 ±1.01	62.22 ±1.27	57.65 ±1.87	67.39 ±2.74	66.23 ±1.59
PodNET-RR	**77.51** ±1.56	**72.04** ±1.75	**66.65** ±2.22	**75.87** ±2.05	**74.89** ±1.85
DER [30]	72.56 ±2.68	71.44 ±1.89	70.22 ±2.34	68.61 ±2.11	69.02 ±1.88
DER-RR	**76.39** ±2.25	**79.09** ±2.31	**81.74** ±1.96	**77.63** ±1.94	**82.89** ±2.15
WA [108]	70.23 ±1.81	67.34 ±2.22	63.77 ±2.72	65.33 ±1.82	60.11 ±1.84
WA-RR	**74.65** ±1.93	**72.25** ±1.95	**68.56** ±1.87	**73.92** ±1.74	**67.17** ±2.15
FOSTER [109]	70.97 ±1.83	69.19 ±2.01	64.37 ±2.51	68.58 ±1.91	67.17 ±1.71
FOSTER-RR	**75.10** ±1.66	**74.09** ±1.85	**71.39** ±2.47	**74.61** ±1.78	**71.49** ±1.91
BiC [83]	67.35 ±1.75	65.78 ±1.99	62.41 ±2.14	61.20 ±2.21	55.98 ±2.84
BiC-RR	**70.51** ±1.81	**72.01** ±2.11	**68.63** ±1.79	**67.91** ±1.93	**64.71** ±2.30

Split-CIFAR100 Dataset

Results on Real-world Dataset [19, 20]

- Average accuracy for Helicopter Attitude dataset in class incremental setting.
- The average accuracy is computed as $Avarege\ Acc = \frac{1}{T}\sum_{i=1}^{T} R_{T,i}$

Sample images for nine different classes

Class	Description	Pitch (P)	Roll (R)
0	NU	$P > \alpha$	$-\alpha \leq R \leq +\alpha$
1	ND	$P < -\alpha$	$-\alpha \leq R \leq +\alpha$
2	RP	$-\alpha \leq P \leq +\alpha$	$R > \alpha$
3	RN	$-\alpha \leq P \leq +\alpha$	$R < -\alpha$
4	NU & RP	$P > \alpha$	$R > \alpha$
5	NU & RN	$P > \alpha$	$R < -\alpha$
6	ND & RP	$P < -\alpha$	$R > \alpha$
7	ND & RN	$P < -\alpha$	$R < -\alpha$
8	L	$-\alpha \leq P \leq +\alpha$	$-\alpha \leq R \leq +\alpha$

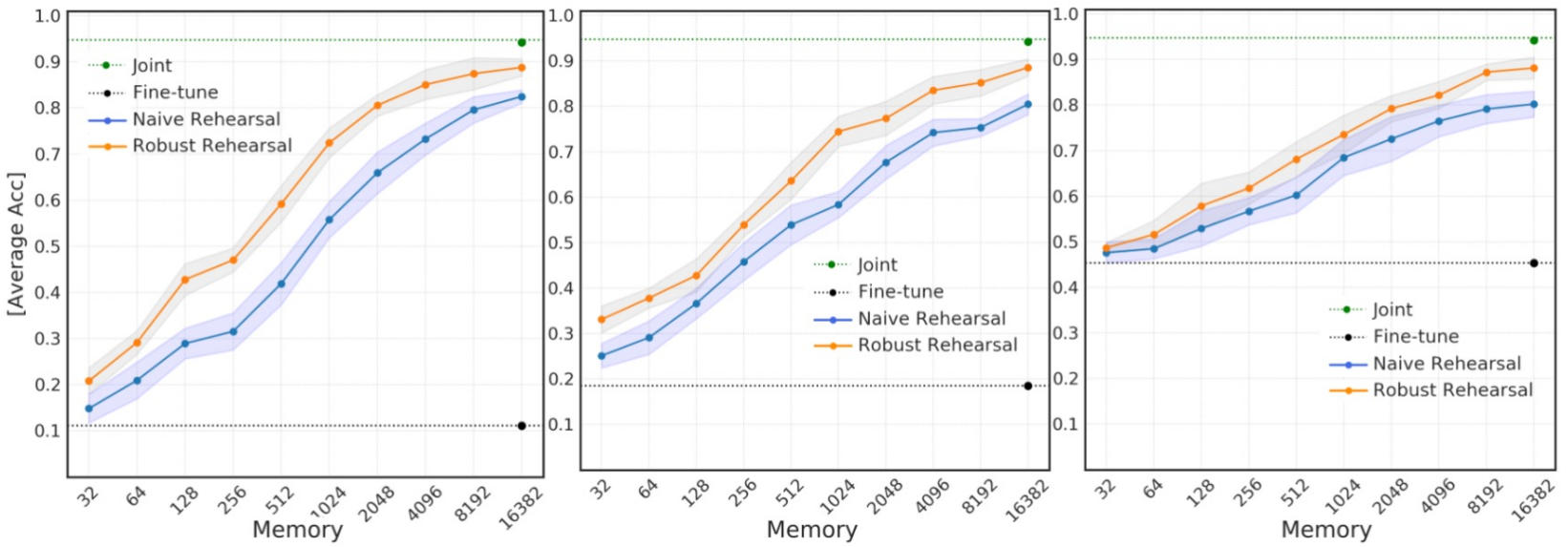
A threshold value, denoted by$\alpha = 3$, is applied to Flight Data Recorder (FDR) metrics for pitch and roll, establishing nine mutually exclusive discrete attitude classes. Abbreviations used: NU - nose up, ND - nose down, RP - roll positive; RN - roll negative, and L - level or steady-state.

CL Method	S-76 Helicopter Attitude Dataset		
	8 steps	5 steps	3 steps
Joint		88.6±0.61	
Fine-tune	12.9±0.83	21.65±0.68	30.77±1.12
iCaRL [19]	44.98 ±0.73	52.75 ±1.15	61.28 ±0.93
iCaRL-RR	**48.32** ±0.82	**60.45** ±1.45	**68.21** ±1.41
PODNet [29]	51.27 ±1.76	62.33 ±2.06	79.12 ±0.87
PODNet-RR	**58.96** ±1.71	**69.15** ±2.21	**83.25** ±1.37
DER [30]	50.83 ±1.24	65.88 ±1.71	77.20 ±0.82
DER-RR	**60.97** ±1.13	**72.52** ±0.99	**85.51** ±1.39
WA [108]	43.61 ±1.61	57.83 ±0.89	64.87 ±1.29
WA-RR	**45.72** ±1.36	**58.89** ±1.02	**65.60** ±1.10
FOSTER [109]	48.69 ±1.28	58.77 ±1.66	65.73 ±0.86
FOSTER-RR	**54.59** ±0.94	**60.55** ±1.39	**69.64** ±0.91
BiC [83]	37.51 ±0.94	48.30 ±1.71	65.23 ±0.82
BiC-RR	**41.63** ±1.08	**50.93** ±1.78	**66.35** ±1.19

Helicopter Attitude Dataset

Results (Continued)

- Effect of the Rehearsal Buffer Size
  - We assessed the impact of rehearsing the CL-robust samples across ten sizes of rehearsal memory on the CIFAR10 dataset.
  - Robust rehearsal demonstrated superior accuracy compared to naive rehearsal methods across ten different rehearsal memory sizes.



Results (Continued)

- Diverse Feature Specialization in Deep Learning
  - We empirically demonstrate that deep learning models acquire diverse features when trained with distinct training protocols.
  - Different features solve different optimization objectives.

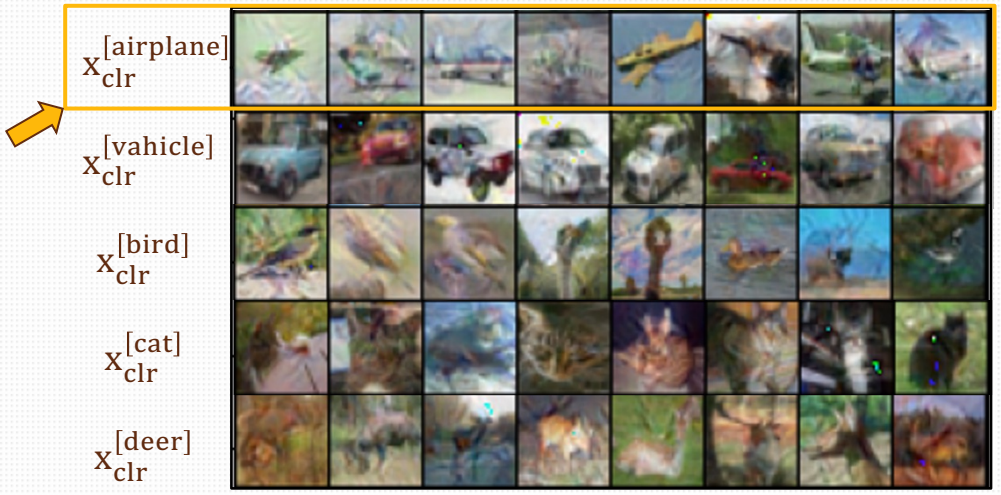
a) Specialized feature of multi-task model

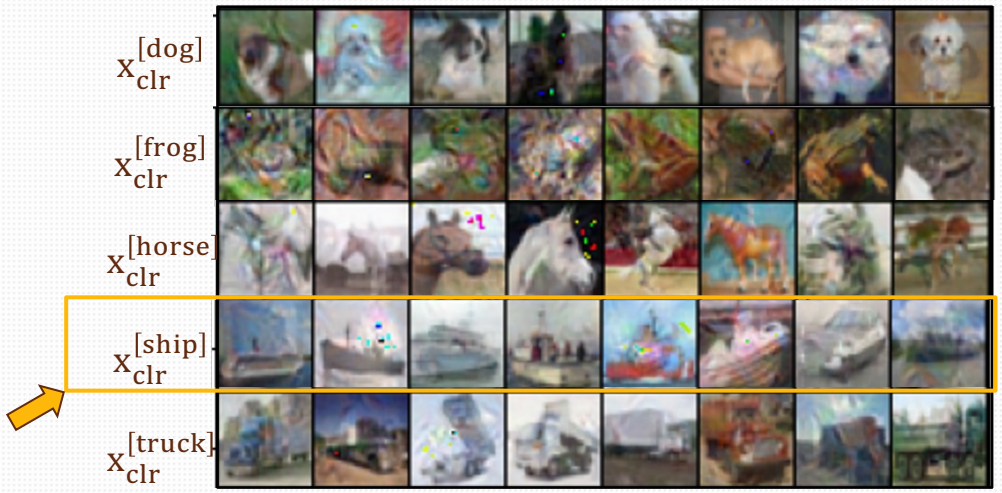
b) Specialized feature of adversarial model

c) Specialized feature of CL model

Results (Continued): A Dream-like Features

- Visualizing the Specialized Features Underscored the Vulnerabilities of CL Approaches Against Adversarial Attacks [40]
  - Intriguingly, the visual resemblance between CL-robust samples and adversarially compromised images suggests that CL-robust features and adversarially non-robust features coexist.
  - Further, it highlights the vulnerability of CL tasks within CL approaches against adversarial attacks [40].





Conclusions

- Our approach eliminates the necessity for pre-trained Oracle CL models and pre-distilled CL robustified datasets for CL [7, 8].
- We introduce a novel framework that not only distills CL-robust features using the CL model at time t, but also perform re-distillation (re-consolidation), allowing for the assimilation of updated understanding of prior task knowledge [8].
- Our investigation into the impact of various optimization training objectives within joint, continual, and adversarial learning highlighted that the optimization objective in deep neural networks essentially dictates feature learning.
- Our findings suggest that closely adhering to neuroscience principles can mitigate the long-standing challenge of catastrophic forgetting in CL models.

Outline

4.

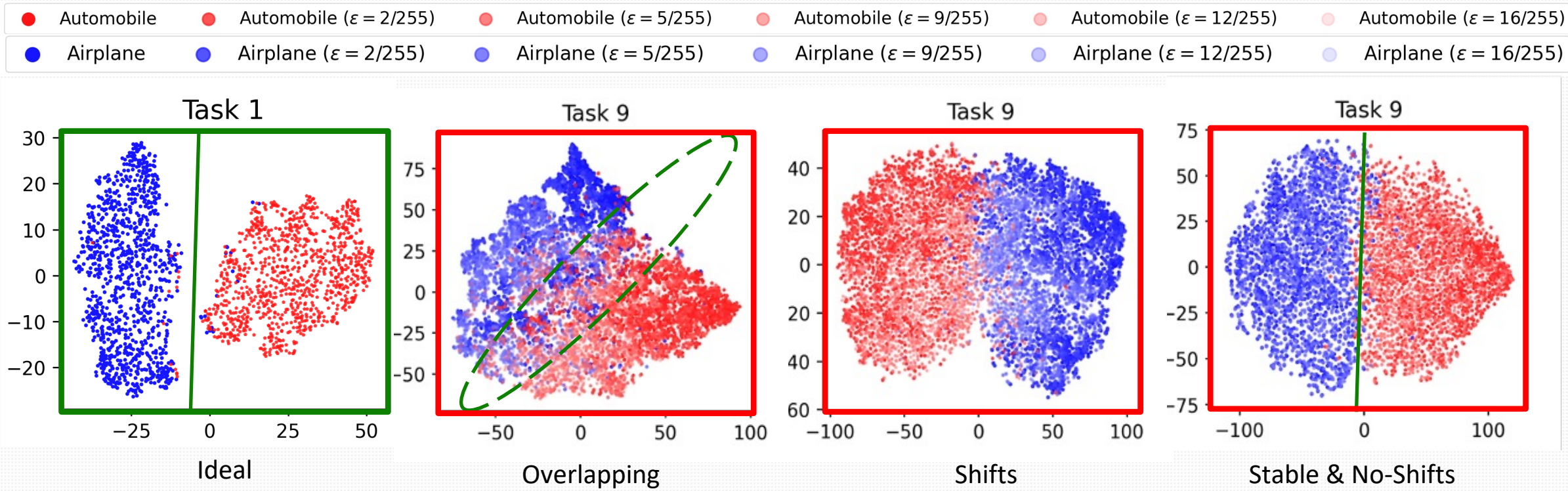
Can rehearsal memory be diversified to prevent rehearsal memory overfitting in CL models?

Hikmat. K, et all, "Adversarially Robust Continual Learning," *2022 International Joint Conference on Neural Networks (IJCNN)*, Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892970.

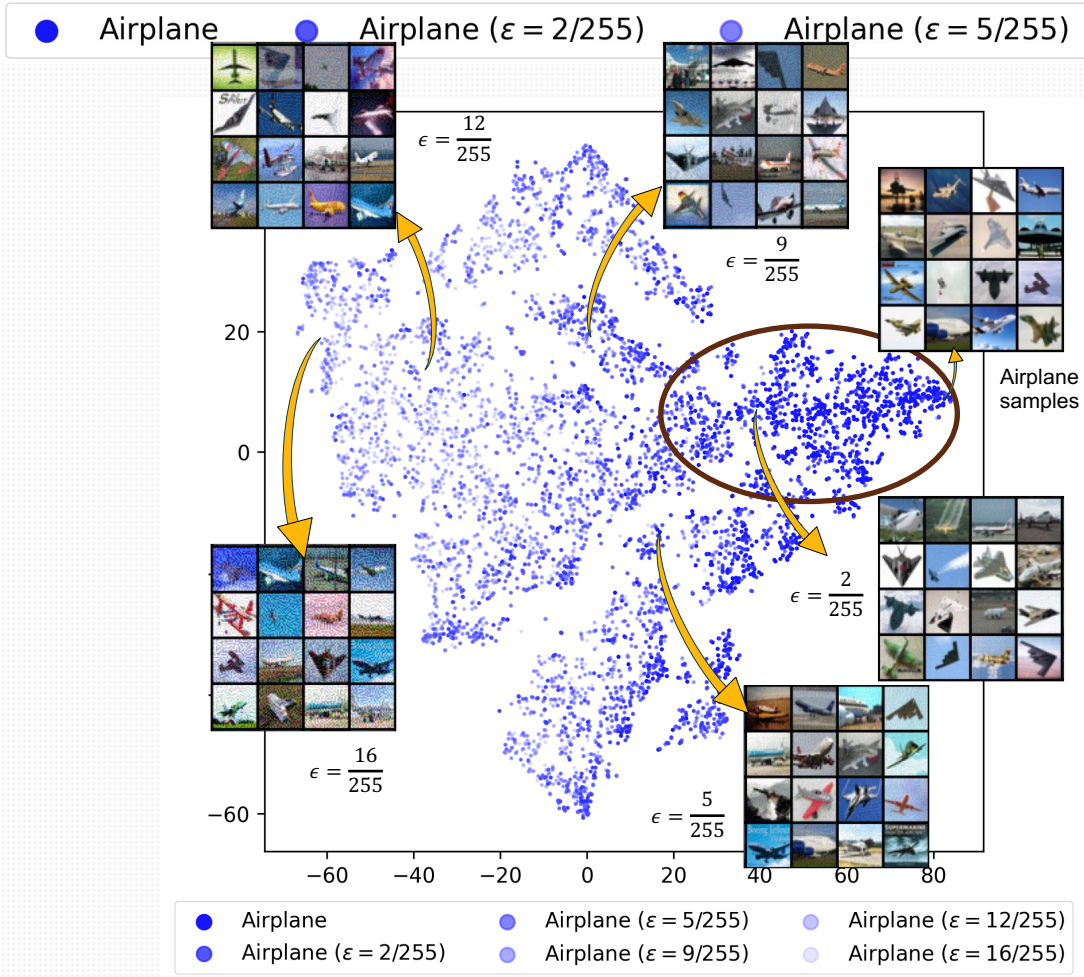
Rehearsal Memory Overfitting in CL

- Rehearsal-based approaches are widely adopted to mitigate catastrophic forgetting in continual learning models.
- However, these approaches encounter a rehearsal memory overfitting, where the model becomes overly specialized on a limited set of memory samples, resulting in catastrophic forgetting.
- Leading to a progressive decay in the effectiveness of the rehearsal memory, ultimately causing the model to forget previously learned tasks.

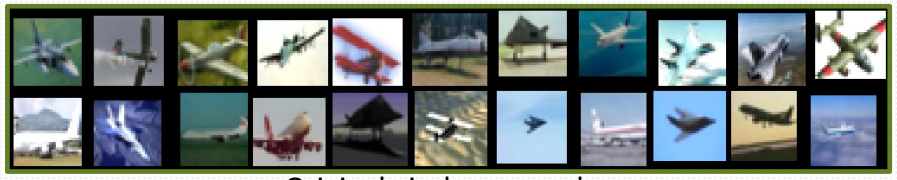
Idea Illustration: Adversarially Diversified Rehearsal Memory



Idea Illustration: Adversarially Diversified Rehearsal Memory



t-SNE visualization showing the dispersion of 'airplane' class features under increasing adversarial perturbations



Original airplane samples

Correctly classified adversarially diversified airplane samples

Incorrectly classified adversarially diversified airplane samples

Adversarially Diversified Rehearsal Memory (ADRM)

- Rehearsal-Based Continual Learning

$$\min_{\theta \in \Theta} \Big[\mathbb{E}_{(x_t, y_t) \sim \mathcal{D}_t} \overbrace{\mathcal{L}(\theta, x_t, y_t)}^{\text{Current task}} + \mathbb{E}_{(x_m, y_m) \sim \mathcal{M}_t} \overbrace{\mathcal{L}(\theta, x_m, y_m)}^{\text{Memory}} \Big]$$

- Aims to increase the complexity of memory samples using FGSM [10]

$$x_{diversified} = x_m + \epsilon \,.\, sign(\nabla_{x_m} J(\theta,\ x_m,\ y_m))$$

- Benefits of ADRM:
  - Prevents rehearsal memory overfitting.
  - Maintains the effectiveness of rehearsal memory samples.
  - Enables CL models to learn new tasks without performance degradation.

Baselines and Results

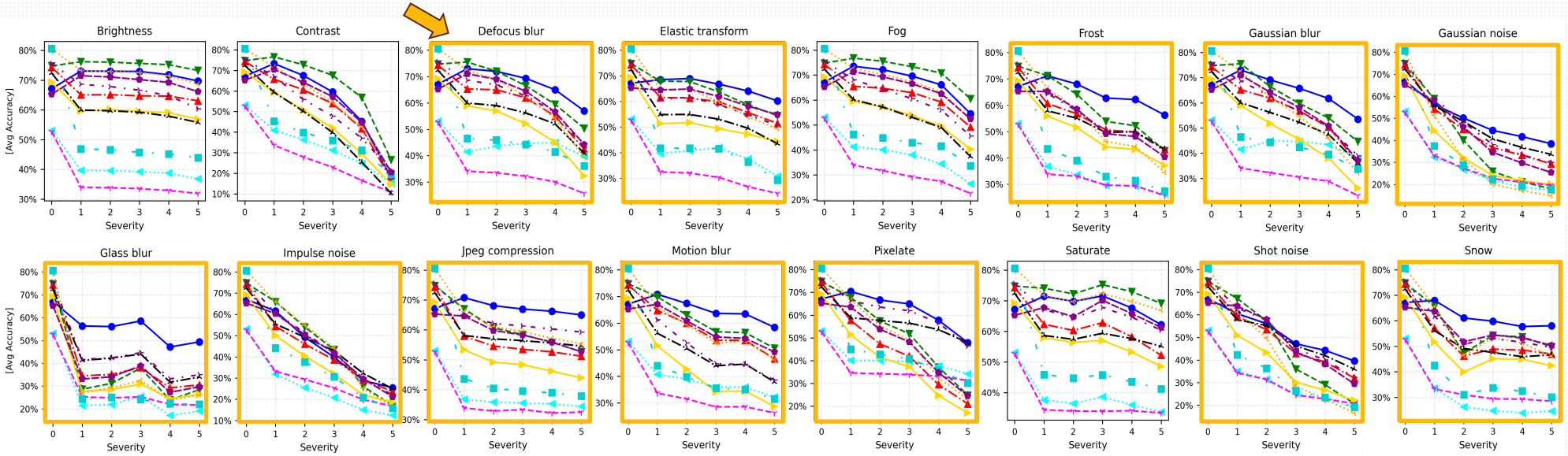
- The ADRM outperforms several established CL approaches and achieves comparable results to state-of-the-art CL approaches.
- ADRM, when rehearsed with 10% adversarially diversified memory, demonstrates superior performance over variants incorporating 25%, 50%, 75%, and 100% diversified memory.
- The 10% adversarially diversified memory suggests that a minimal level of adversarial diversification, can prevent the rehearsal memory overfitting in CL.

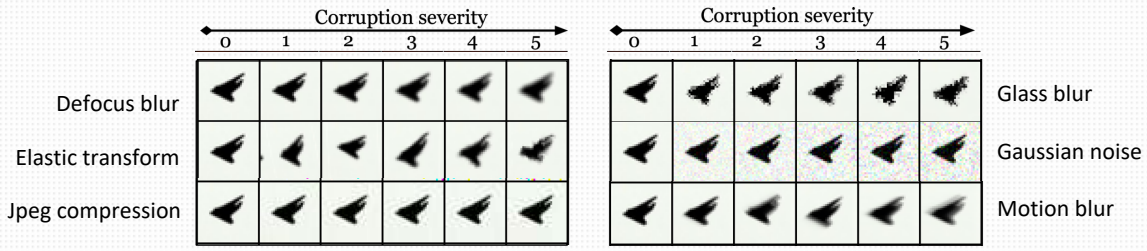
CL Methods	Split-CIFAR10		
	9 steps	5 steps	2 steps
Joint		94.8	
Fine-tune	11.11	18.54	45.31
Experience Replay [6]	67.62	75.97	80.26
iCaRL [33]	69.25	74.85	79.98
BiC [26]	53.11	71.01	86.57
PODNet [34]	72.41	76.96	87.64
WA [25]	75.56	81.67	85.34
DER [35]	74.33	78.14	82.57
SimpleCIL [27]	52.67	54.54	75.01
FOSTER [36]	74.61	80.40	83.89
FETRIL [37]	65.26	67.51	84.52
MEMO [38]	80.60	85.93	87.81
ADRM (0.1)	74.76	80.59	83.95
ADRM (0.25)	72.39	79.15	85.39
ADRM (0.5)	69.20	76.14	81.41
ADRM (0.75)	68.59	76.36	80.17
ADRM (1)	65.39	76.21	82.86

Comparative performance of CL methods on the Split-CIFAR10 dataset over 9 steps, 5 steps, and 2 steps.

Evaluating Against Common Corruptions

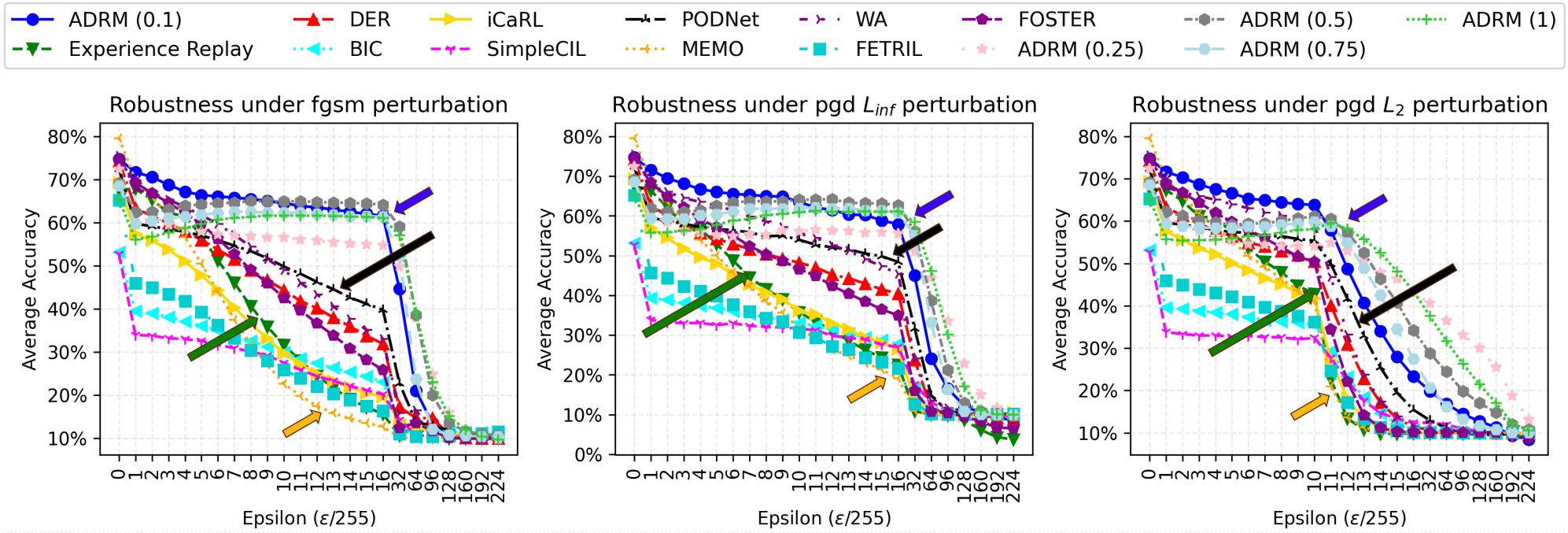
- CIFAR10-C (Corrupted) is a variant of the original CIFAR10 dataset that contains corrupted images [6, 21].
- The ADRM model performed relatively best (in 15 out of 19 cases) in various noisy conditions.





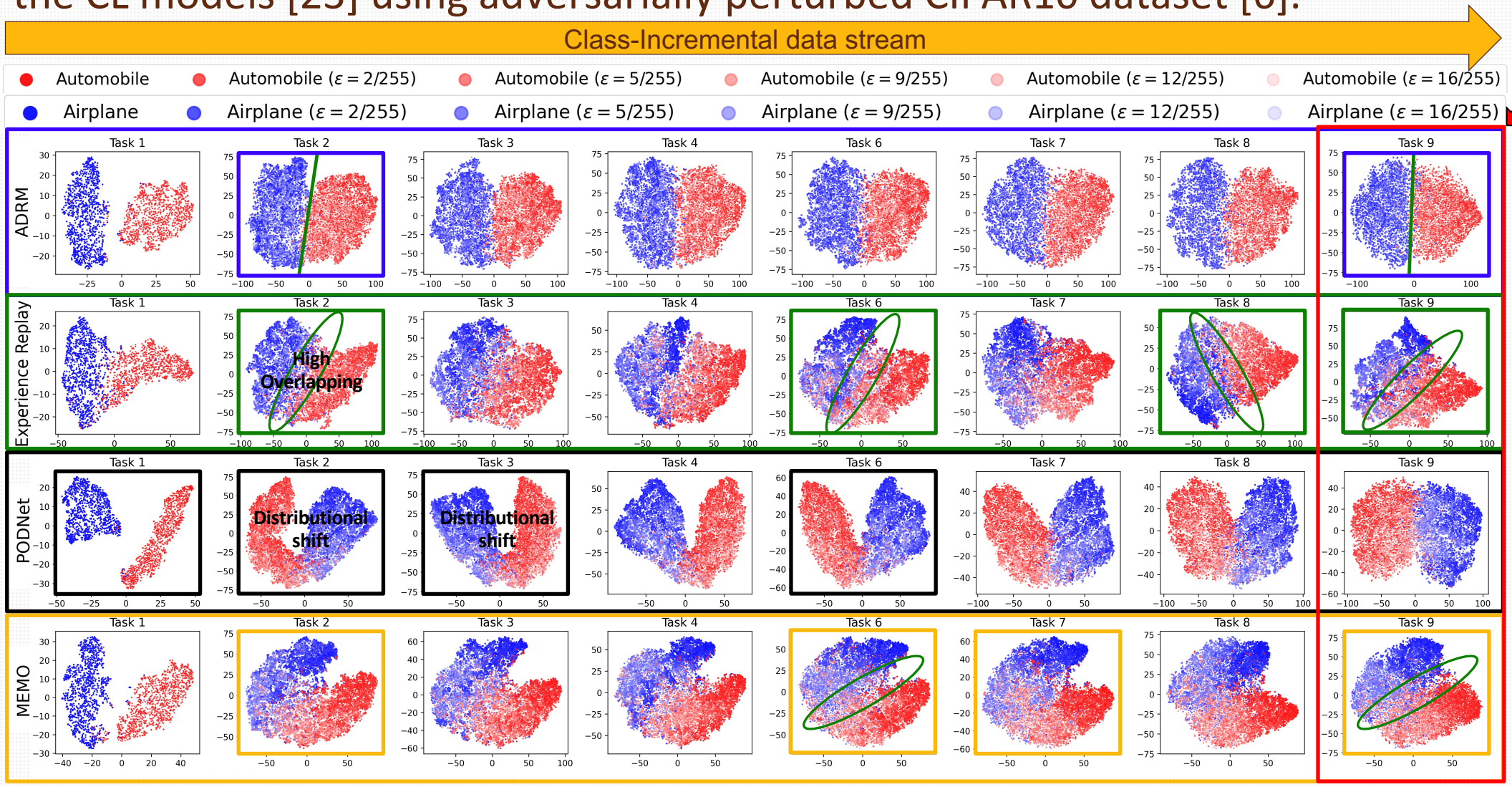
Evaluating Against Adversarial Attacks

- All ADRM variants experienced less catastrophic forgetting in adversarial conditions.



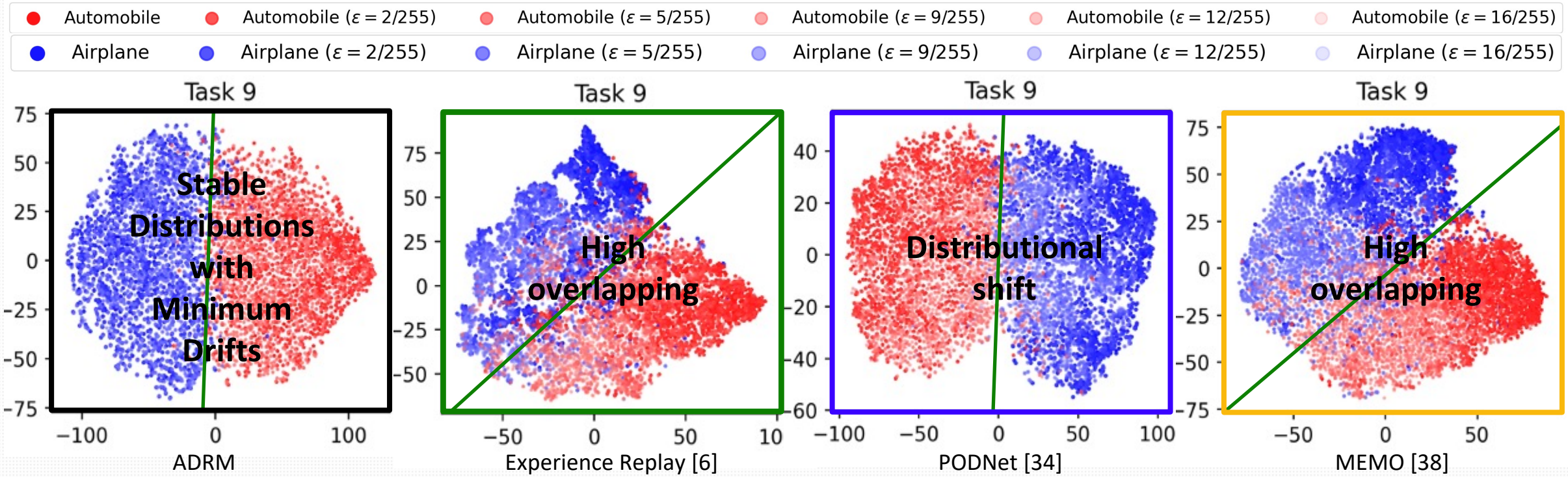
t-SNE Visualization of Latent Feature Distributions in CL

- We utilized t-SNE to visualize the distributions of latent features at the logit layer of the CL models [23] using adversarially perturbed CIFAR10 dataset [6].



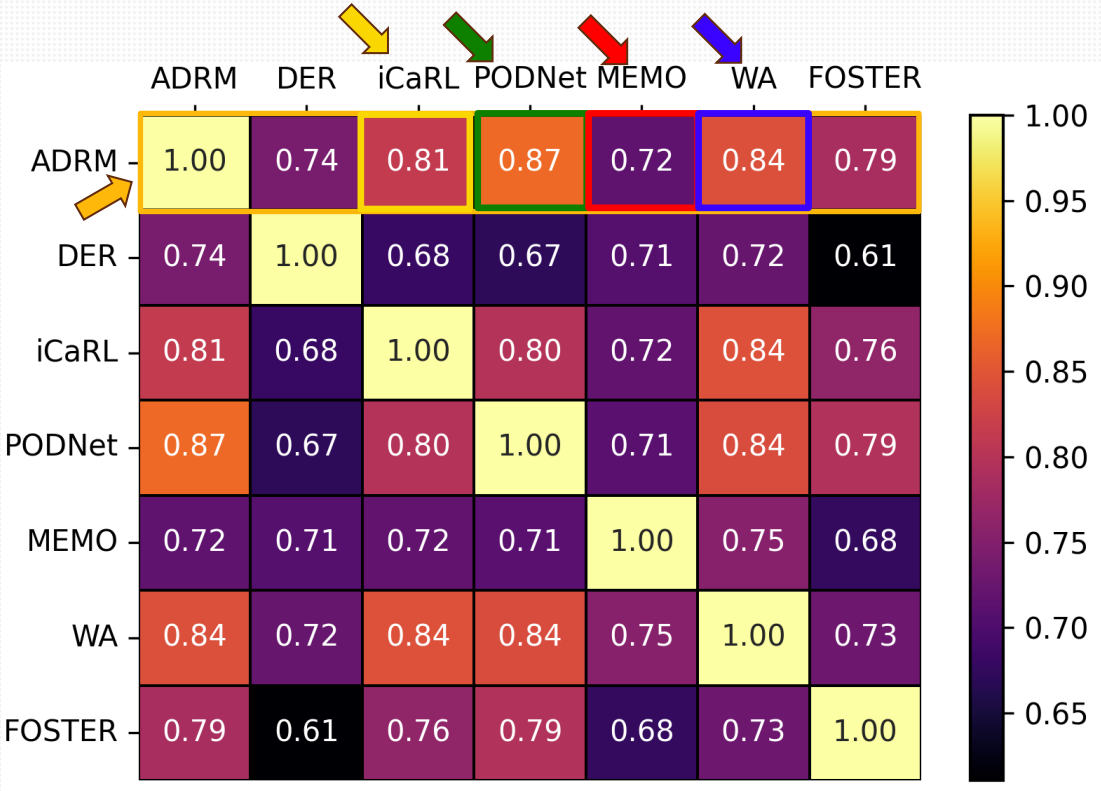
t-SNE Visualization of Latent Feature Distributions in CL (Continued)

- Visualization of the learned latent feature distributions for airplanes and automobiles (from the adversarially perturbed CIFAR10 dataset [6]) at the logit layer of CL models [23].

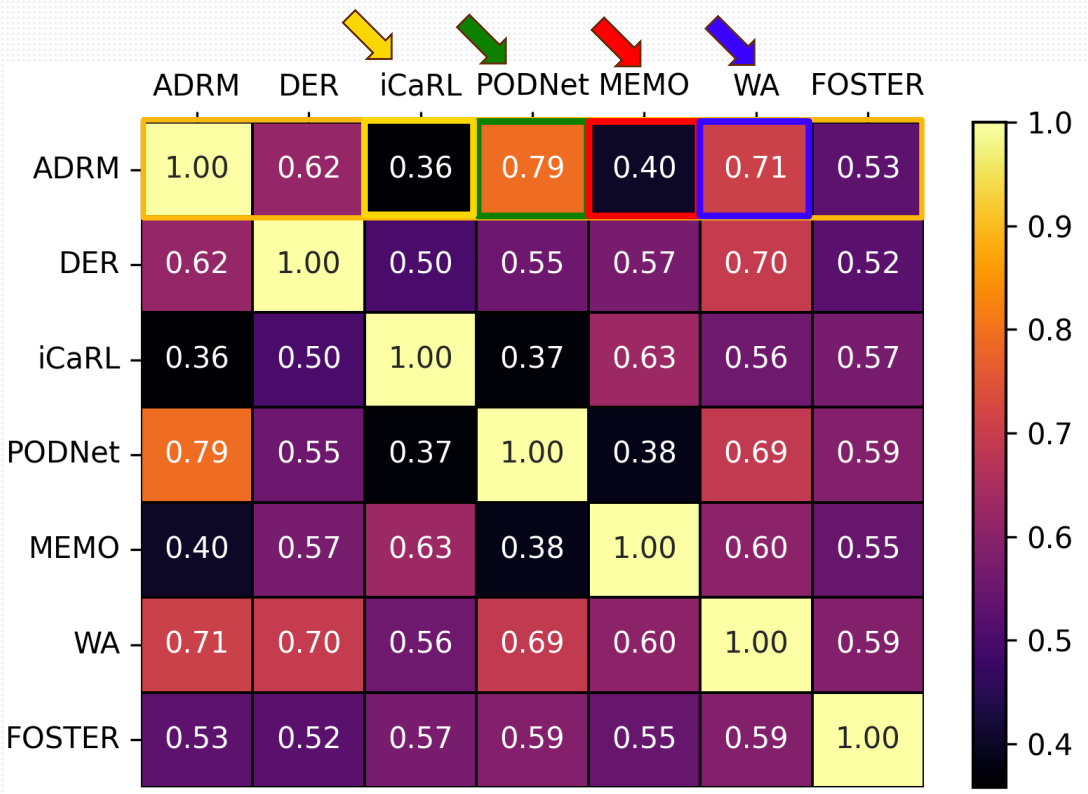


Features Similarity Matrices: A Central Kernel Alignment Analysis [24]

- The CL models displayed higher feature similarities (Central Kernel Alignment [24]) to ADRM on Standard CIFAR10 but differed in feature similarities on adversarially perturbed CIFAR10 [6].



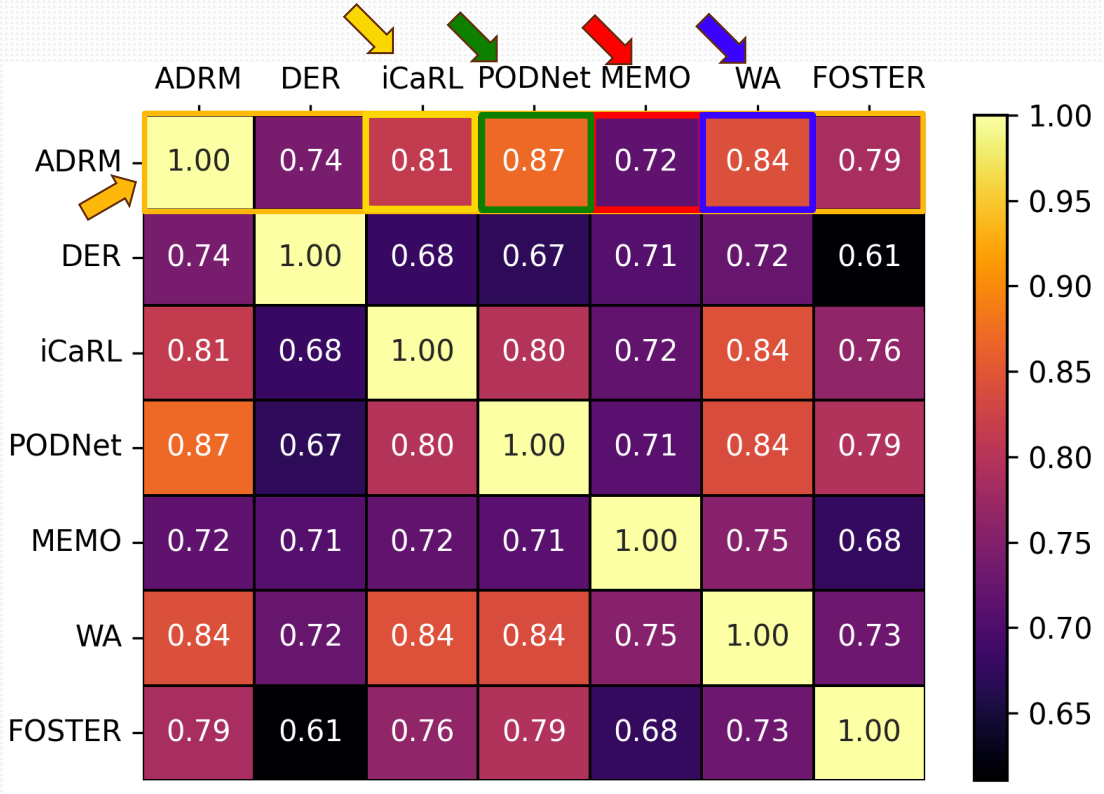
a) Feature Similarity on Standard CIFAR10



b) Feature Similarity on adversarial perturbed CIFAR10

Features Similarity Matrices: A Central Kernel Alignment Analysis [24]

- The CL models displayed higher feature similarities (Central Kernel Alignment [24]) to ADRM on Standard CIFAR10 but differed in feature similarities on adversarially perturbed CIFAR10 [6].



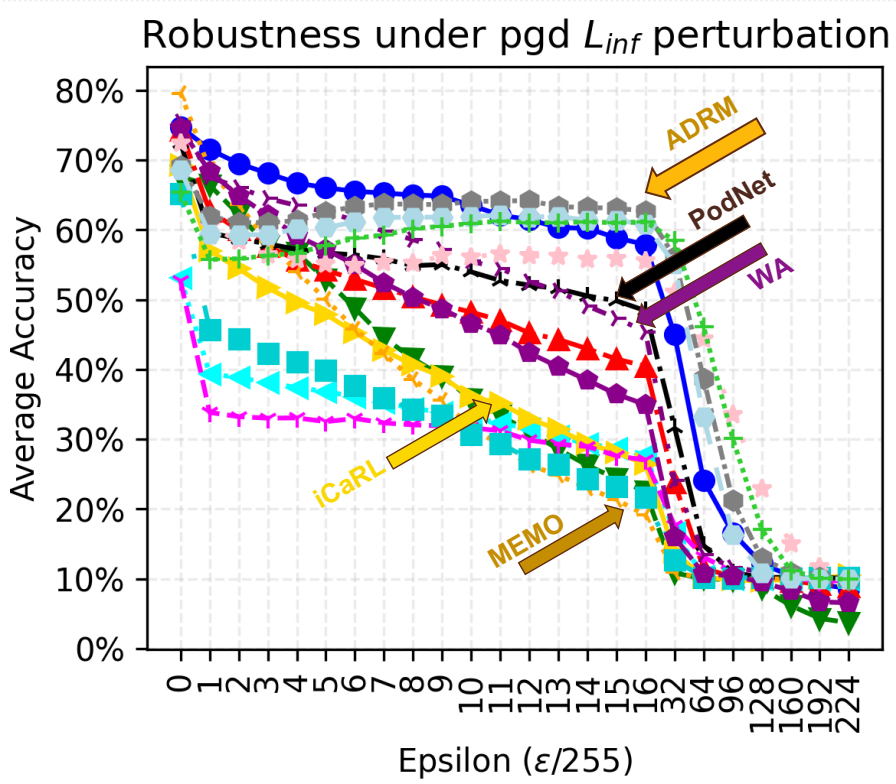
a) Feature Similarity on Standard CIFAR10

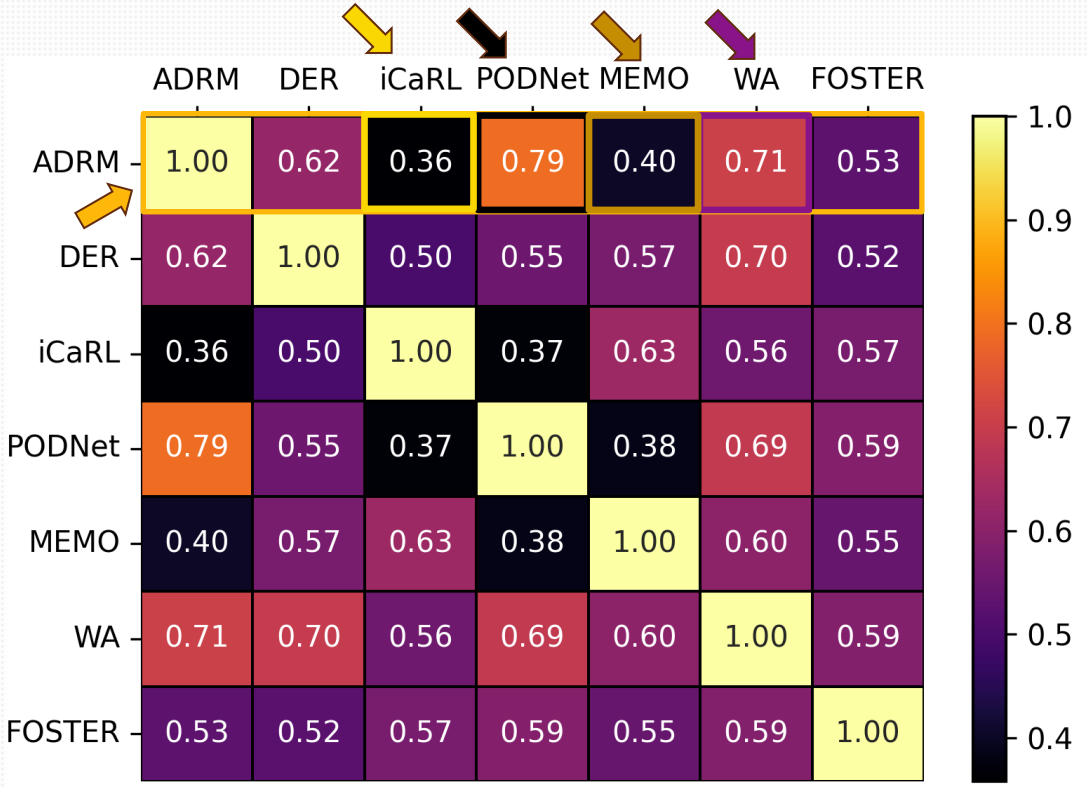
CL Methods	Split-CIFAR10		
	9 steps	5 steps	2 steps
Joint		94.8	
Fine-tune	11.11	18.54	45.31
Experience Replay [6]	67.62	75.97	80.26
iCaRL [33]	69.25	74.85	79.98
BiC [26]	53.11	71.01	86.57
PODNet [34]	72.41	76.96	87.64
WA [25]	75.56	81.67	85.34
DER [35]	74.33	78.14	82.57
SimpleCIL [27]	52.67	54.54	75.01
FOSTER [36]	74.61	80.40	83.89
FETRIL [37]	65.26	67.51	84.52
MEMO [38]	80.60	85.93	87.81
ADRM (0.1)	74.76	80.59	83.95
ADRM (0.25)	72.39	79.15	85.39
ADRM (0.5)	69.20	76.14	81.41
ADRM (0.75)	68.59	76.36	80.17
ADRM (1)	65.39	76.21	82.86

Comparative performance of CL methods on the Split-CIFAR10 dataset over 9 steps, 5 steps, and 2 steps.

Features Similarity Matrices: A Central Kernel Alignment Analysis [24]

- The CL models displayed higher feature similarities (Central Kernel Alignment [24]) to ADRM on Standard CIFAR10 but differed in feature similarities on adversarially perturbed CIFAR10 [6].





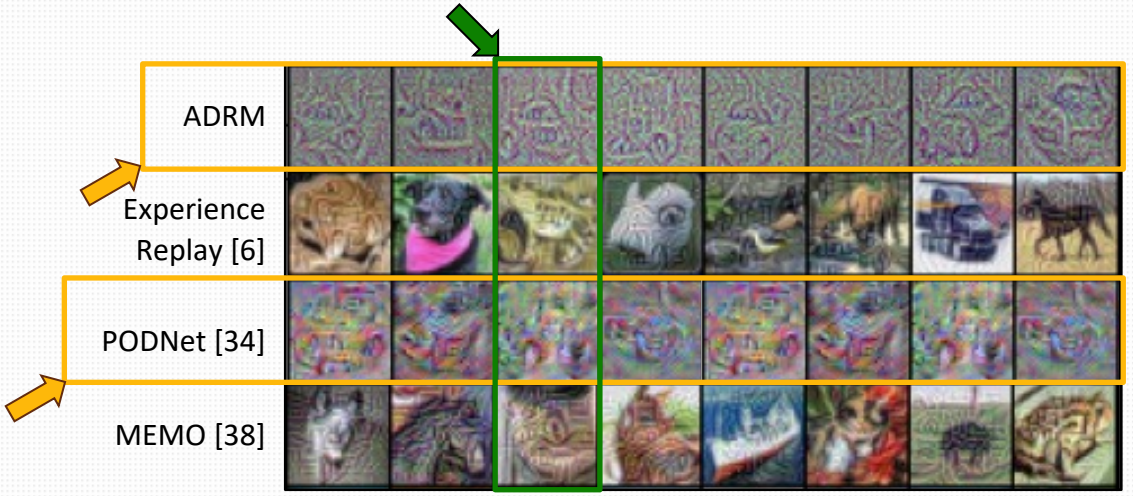
b) Feature Similarity on adversarial perturbed CIFAR10

Visualizing CL Models' Features for Automobile Class in Input Space

- The features visualization for the CL model $f^*_{\theta_t}$ is performed by solving the below:

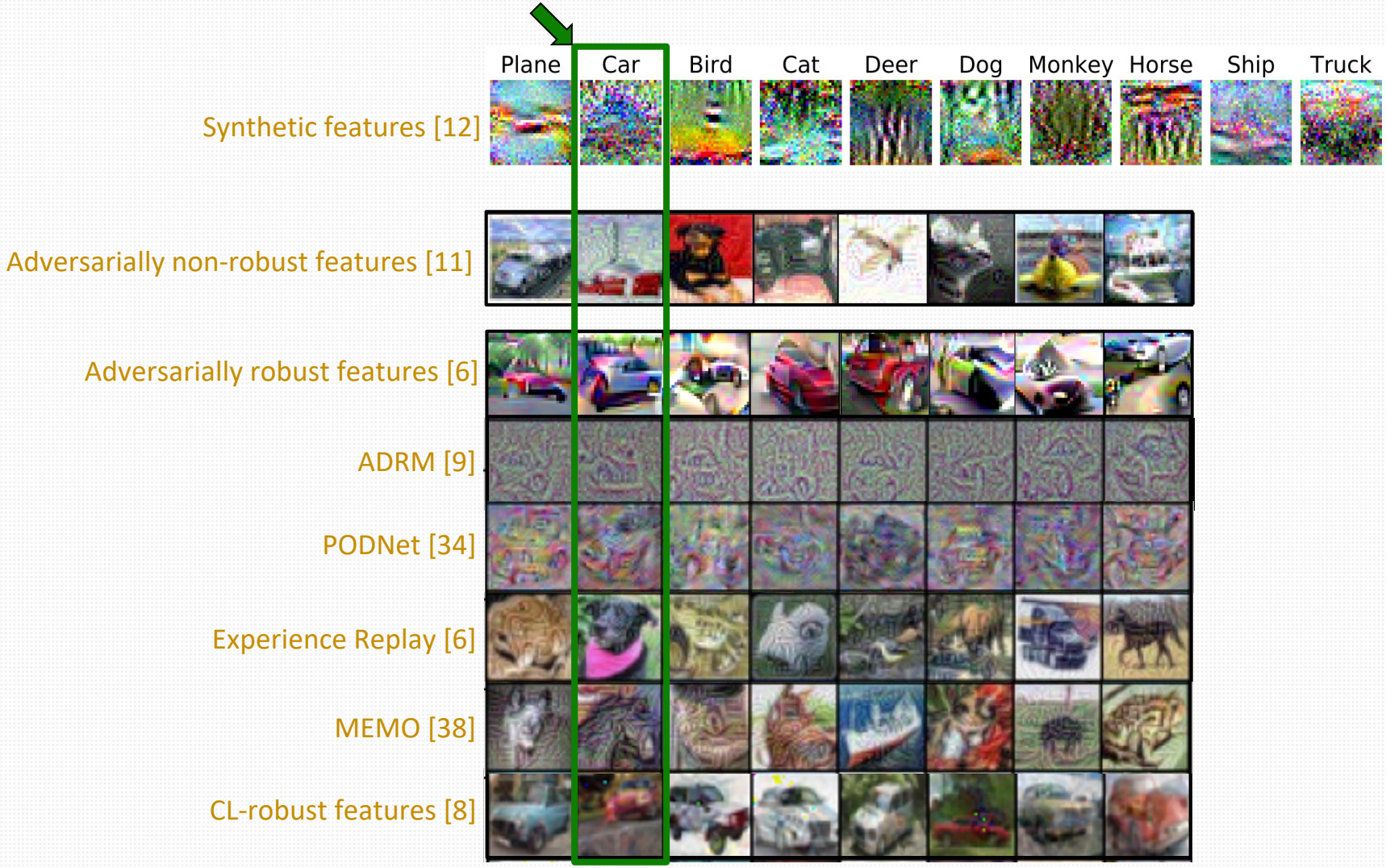
$$\min_{x_r} \| f^*_{\theta_t}(x_r) - f^*_{\theta_t}(x) \|_2^2, \quad x_r \sim D$$

- ADRM learned the salient features of the object as compared to the others CL models.



a) Visualization of disentangled features for car class in input space

Automobile (Car) Features (Continued)



Outline

Conclusions

- Features in the input dataset played a vital role in the CL model's robustness and catastrophic forgetting [6].
- CL-robust features can be disentangled to train the CL model to alleviate catastrophic forgetting [7].
- Brain-inspired memory consolidation and re-consolidation can be incorporated into the design of the CL approach to distilling the CL-robust features to mitigate catastrophic forgetting [8].
- Rehearsal memory can be diversified to prevent rehearsal memory overfitting in the CL models [9].

Outline

I. Introduction
   1. Continual Learning
II. Related Work
III. Research Questions
IV. Our Work: A Multidisciplinary Approach
V. Primary Contributions
   1. Adversarially Robust Continual Learning
   2. The Importance of Robust Features in Mitigating Catastrophic Forgetting
   3. Brain-Inspired Continual Learning Robust Feature Distillation and Re-consolidation for Class Incremental Learning
   4. Adversarially Diversified Rehearsal Memory (ADRM): Mitigating Memory Overfitting Challenge in Continual Learning
VI. Conclusions
VII. Future Work

Future Work

- Explore the possibility of developing continually and adversarially robust features.
- Explore the possibility of using self-attention-like modifications to distill CL-robust features during learning, eliminating extra computational time for CL-robust samples.
- Expand to other tasks, including continual object segmentation, etc.
- Include more recent architectures in consideration.

First-Author Publications

1. H. Khan, N. C. Bouaynaya and Ghulam. Rasool, "Adversarially Diversified Rehearsal Memory (ADRM): Mitigating Memory Overfitting Challenge in Continual Learning," 2024 International Joint Conference on Neural Networks (IJCNN), Yokohama, Japan, 2024, pp. 1-8.
2. H. Khan, N. C. Bouaynaya and Ghulam. Rasool, "Brain-Inspired Continual Learning: Robust Feature Distillation and Re-Consolidation for Class Incremental Learning," in IEEE Access, vol. 12, pp. 34054-34073, 2024, doi: 10.1109/ACCESS.2024.33694.88.
3. H. Khan, N. C. Bouaynaya and G. Rasool, "The Importance of Robust Features in Mitigating Catastrophic Forgetting," in 2023 IEEE Symposium on Computers and Communications (ISCC), Gammarth, Tunisia, 2023 pp. 752-757.
4. H. Khan, N. C. Bouaynaya and G. Rasool, "Adversarially Robust Continual Learning," 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892970. doi: 10.1109/ISCC58397.2023.10218203.
5. H. Khan, G. Rasool, N. C. Bouaynaya, T. Tyler, T Lacey and C. C. Johnson. " Deep Ensemble for Rotorcraft Attitude Prediction." In Vertical Flight Society's 77th Annual Forum Technology Display. 2021.
6. H. Khan, G. Rasool, N. C. Bouaynaya, T. Travis, T. Lacey, and C. C. Johnson. " Explainable AI: Rotorcraft Attitude Prediction." In Vertical Flight Society's 76th Annual Forum and Technology Display. 2020.
7. H. Khan, G. Rasool, N. C. Bouaynaya, and C. C. Johnson. "Rotorcraft Flight Information Inference from Cockpit Videos using Deep Learning." In Vertical Flight Society's 75th Annual Forum and Technology Display. 2019.

References

[1]. De Lange, Matthias, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. "A continual learning survey: Defying forgetting in classification tasks." *IEEE transactions on pattern analysis and machine intelligence* 44, no. 7 (2021): 3366-3385.

[2]. Van de Ven, Gido M., and Andreas S. Tolias. "Three scenarios for continual learning." arXiv preprint arXiv:1904.07734 (2019).

[3]. Mermillod, Martial, Aurélia Bugaiska, and Patrick Bonin. "The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects." *Frontiers in psychology* 4 (2013): 504.

[4]. French, Robert M. "Catastrophic forgetting in connectionist networks." Trends in cognitive sciences 3, no. 4 (1999): 128-135.

[5]. Wang, Liyuan, Xingxing Zhang, Hang Su, and Jun Zhu. "A comprehensive survey of continual learning: Theory, method and application." IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).

[6]. H. Khan, N. C. Bouaynaya and G. Rasool, "Adversarially Robust Continual Learning," 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892970.
doi: 10.1109/ISCC58397.2023.10218203.

[7]. H. Khan, N. C. Bouaynaya and G. Rasool, "The Importance of Robust Features in Mitigating Catastrophic Forgetting," in 2023 IEEE Symposium on Computers and Communications (ISCC), Gammarth, Tunisia, 2023 pp. 752-757.

[8]. H. Khan, N. C. Bouaynaya and G. Rasool, "Brain-Inspired Continual Learning: Robust Feature Distillation and Re-Consolidation for Class Incremental Learning," in IEEE Access, vol. 12, pp. 34054-34073, 2024, doi: 10.1109/ACCESS.2024.33694.88.

[9]. H. Khan, N. C. Bouaynaya and G. Rasool, "Adversarially Diversified Rehearsal Memory (ADRM): Mitigating Memory Overfitting Challenge in Continual Learning," 2024 International Joint Conference on Neural Networks (IJCNN), Yokohama, Japan, 2024, pp. 1-8.

[10]. Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." arXiv preprint arXiv:1412.6572 (2014).

[11]. Ilyas, Andrew, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. "Adversarial examples are not bugs, they are features." Advances in neural information processing systems 32 (2019).

[12]. Wang, Tongzhou, Jun-Yan Zhu, Antonio Torralba, and Alexei A. Efros. "Dataset distillation." arXiv preprint arXiv:1811.10959 (2018).

[13]. Kim, Jang-Hyun, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. "Dataset condensation via efficient synthetic-data parameterization." In International Conference on Machine Learning, pp. 11102-11118. PMLR, 2022.

[14]. Deng, Zhiwei, and Olga Russakovsky. "Remember the past: Distilling datasets into addressable memories for neural networks." Advances in Neural Information Processing Systems 35 (2022): 34391-34404.

References (Continued)

[15]. Rolnick, David, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. "Experience replay for continual learning." Advances in neural information processing systems 32 (2019).

[16]. Kirkpatrick, James, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan et al. "Overcoming catastrophic forgetting in neural networks." Proceedings of the national academy of sciences 114, no. 13 (2017): 3521-3526.

[17]. Yan, Shipeng, Jiangwei Xie, and Xuming He. "Der: Dynamically expandable representation for class incremental learning." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3014-3023. 2021.

[18]. Douillard, Arthur, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. "Podnet: Pooled outputs distillation for small-tasks incremental learning." In Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part XX 16, pp. 86-102. Springer International Publishing, 2020.

[19]. H. Khan, C. Johnson, R. University, N. Bouaynaya, G. Rasool, T. Travis, et al., "Explainable AI: Rotorcraft attitude prediction", Proc. Vertical Flight Soc. 76th Annu. Forum, Oct. 2020.

[20]. H. Khan, C. Johnson, R. University, N. Bouaynaya, G. Rasool, L. Thompson, et al., "Deep ensemble for rotorcraft attitude prediction", Proc. Vertical Flight Soc. 77th Annu. Forum, May 2021.

[21]. Hendrycks, Dan, and Thomas Dietterich. "Benchmarking neural network robustness to common corruptions and perturbations." arXiv preprint arXiv:1903.12261 (2019).

[22]. Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9, no. 11 (2008).

[23]. Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9, no. 11 (2008).

[24]. Kornblith, Simon, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. "Similarity of neural network representations revisited." In International conference on machine learning, pp. 3519-3529. PMLR, 2019.

[25]. Zhao, Bowen, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. "Maintaining discrimination and fairness in class incremental learning." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13208-13217. 2020.

[26]. Wu, Yue, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. "Large scale incremental learning." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 374-382. 2019.

[27]. Zhou, Da-Wei, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. "Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need." arXiv preprint arXiv:2303.07338 (2023).

[28]. Wamsley, Erin J. "Dreaming and offline memory consolidation." Current neurology and neuroscience reports 14 (2014): 1-7.

[29]. Lee, Albert K., and Matthew A. Wilson. "Memory of sequential experience in the hippocampus during slow wave sleep." Neuron 36.6 (2002): 1183-1194.

References (Continued)

[30]. Wamsley, Erin J. "Dreaming and offline memory consolidation." Current neurology and neuroscience reports 14 (2014): 1-7.

[31] Wamsley, Erin J., et al. "Cognitive replay of visuomotor learning at sleep onset: temporal dynamics and relationship to task performance." Sleep 33.1 (2010): 59-68.

[33]. Rebuffi, Sylvestre-Alvise, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. "icarl: Incremental classifier and representation learning." In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 2001-2010. 2017.

[34]. Douillard, Arthur, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. "Podnet: Pooled outputs distillation for small-tasks incremental learning." In Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part XX 16, pp. 86-102. Springer International Publishing, 2020.

[35]. Yan, Shipeng, Jiangwei Xie, and Xuming He. "Der: Dynamically expandable representation for class incremental learning." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3014-3023. 2021.

[36]. Wang, Fu-Yun, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. "Foster: Feature boosting and compression for class-incremental learning." In European conference on computer vision, pp. 398-414. Cham: Springer Nature Switzerland, 2022.

[37]. Petit, Grégoire, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. "Fetril: Feature translation for exemplar-free class-incremental learning." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 3911-3920. 2023.

[38]. Tsipras, Dimitris, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. "Robustness may be at odds with accuracy." arXiv preprint arXiv:1805.12152 (2018).

[39]. Zhao, Bo, Konda Reddy Mopuri, and Hakan Bilen. "Dataset condensation with gradient matching." arXiv preprint arXiv:2006.05929 (2020).

[40]. Li, Yiming, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. "Backdoor learning: A survey." IEEE Transactions on Neural Networks and Learning Systems (2022).

[41]. Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network." arXiv preprint arXiv:1503.02531 (2015).

[42]. Waseda, Futa, Sosuke Nishikawa, Trung-Nghia Le, Huy H. Nguyen, and Isao Echizen. "Closer look at the transferability of adversarial examples: How they fool different models differently." In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1360-1368. 2023.

[43]. Tsipras, Dimitris, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. "Robustness may be at odds with accuracy." arXiv preprint arXiv:1805.12152 (2018).

[44]. Kim, Junho, Byung-Kwan Lee, and Yong Man Ro. "Distilling robust and non-robust features in adversarial examples by information bottleneck." Advances in Neural Information Processing Systems 34 (2021): 17148-17159.

[45]. Huang, Tianjin, Vlado Menkovski, Yulong Pei, and Mykola Pechenizkiy. "Bridging the performance gap between fgsm and pgd adversarial training." arXiv preprint arXiv:2011.05157 (2020).

References (Continued)

[46]. Madry, Aleksander, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. "Towards deep learning models resistant to adversarial attacks." arXiv preprint arXiv:1706.06083 (2017).

Thank you!

Dissertation Chair & Academic Advisor: **Nidhal Carla Bouaynaya, Ph.D.**, Professor and Associate Dean for Research, Department of Electrical and Computer Engineering, Rowan University, NJ, USA

Ghulam Rasool, Ph.D., Assistant Member, Department of Machine Learning, Moffitt Cancer Center and Research Institute, FL, USA

Ravi Prakash Ramachandran, Ph.D., Professor, Department of Electrical and Computer Engineering, Rowan University, NJ, USA

Shlomo Engelberg, Ph.D., Professor, Department of Electrical & Electronics and Engineering, Jerusalem College of Technology, Israel

Robi Polikar, Ph.D., Professor, Department Head, Department of Electrical and Computer Engineering, Rowan University, NJ, USA

Cliff Johnson
Senior Research And Development Engineer at Federal Aviation Administration:
Research Engineer, Program Manager, and Flight Test Engineer for the FAA's Aviation Research Division, specializing in research to improve the safety of rotorcraft. Capabilities and skills include performing flight tests, simulations, and studies related to Vertical Lift (Helicopter Flight Data Monitoring, Vision Systems, Acoustic Noise Modelling, Night Vision Goggles, Simulator Device Fidelity), Unmanned Aircraft Systems (Maintenance and Repair, Standardized Procedures, Navigation, Unmanned Aircraft System Safety and Lethality), and eVTOL/Urban Air Mobility (UAM).

Brain-Inspired Continual Learning: Rethinking the Role of Features in the Stability-Plasticity Dilemma

Doctoral Dissertation Presentation

Thank you!

Hikmat Khan
Department of Electrical & Computer Engineering
Rowan University
khanhi83@rowan.edu
Hikmat.khan179@gmail.com

Advisor: Dr. Nidhal C. Bouaynaya

Thank you!

U.S. Department of Education

Federal Aviation Administration

U.S. National Science Foundation

Rowan University

NSF I-Corps Hub
Northeast Region

Questions!