# CS771 - Intro to ML (Autumn 2024): Mini-project 2
# Learning with Prototypes for CIFAR-10 Classification

Group 26: ML-Mog

## Group Details


| Name | Roll Number | Email |
|---|---|---|
| Abhijeet Verma | 210026 | abhiteet21@iitk.ac.in |
| Aditya Saraf | 210067 | saditya21@iitk.ac.in |
| Chinmay Amrutkar | 210286 | chinmayma21@iitk.ac.in |
| Deependra Verma | 210313 | deependrav21@iitk.ac.in |
| Rohan Phad | 210718 | rohanpv21@iitk.ac.in |
| Sanyam Pasricha | 210929 | sanyamp21@iitk.ac.in |


## 1 Introduction

In this project, we explore the application of Learning with Prototypes (LwP) for image classification on subsets of the CIFAR-10 dataset. We are provided with 20 training datasets ($D_1$ to $D_{20}$) of which only $D_1$ has labeled data and their corresponding held-out evaluation datasets ($\hat{D}_1$ to $\hat{D}_{20}$). The first 10 datasets ($D_1$ to $D_{10}$) share the same input distribution, while the remaining 10 datasets ($D_{11}$ to $D_{20}$) come from slightly different distributions. Our goal is to incrementally train models ($f_1$ to $f_{20}$) using LwP, ensuring that performance on previous datasets does not degrade significantly when models are updated.

## 2 Datasets and Preprocessing

### 2.1 Datasets

The CIFAR-10 dataset consists of 60,000 color images of size $32 \times 32$ pixels, divided into 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. For this project:

- **Training Datasets**: 20 subsets ($D_1$ to $D_{20}$), each containing 2,500 images without labels, except for $D_1$, which is labeled.
- **Held-out Datasets**: 20 corresponding evaluation datasets ($\hat{D}_1$ to $\hat{D}_{20}$), each containing 2,500 labeled images for assessing model performance.

### 2.2 Data Preprocessing

To prepare the data for model training and evaluation, we performed the following preprocessing steps:

1. **Normalization**: Used the CIFAR-10 mean and standard deviation to normalize the images.
2. **Transformations**: Applied random cropping with a padding size of 4 and horizontal flipping for data augmentation during feature extractor training.

### 2.3 Feature Extraction

Used a pre-trained ResNet18 model (trained on CIFAR-10) as a feature extractor by removing its final fully connected layer.

# 3 Learning with Prototypes (LwP)

## 3.1 Model Initialization

We initialized our models using the labeled dataset $D_1$:

- Extracted features from images in $D_1$ using the feature extractor.
- Computed class prototypes by averaging features for each class.

$$\mathbf{p}_c^{(1)} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{f}_c^j \tag{1}$$

  where $\mathbf{p}_c^{(1)}$ is the prototype for class $c$ at iteration 1, $\mathbf{f}_c^j$ is the extracted feature from an image of $D_1$ from class $c$ and $n$ is the number of images under class $c$.

- The initial model $f_1$ consists of these prototypes.

## 3.2 Model Updates (Task 1)

For datasets $D_2$ to $D_{10}$, we updated models incrementally:

1. **Feature Extraction**: Extracted features from unlabeled images in $D_i$.
2. **Label Prediction**: Predicted labels using the previous model's prototypes by assigning the class of the nearest prototype.
3. **Confidence Calculation**: Computed confidence scores based on distances to prototypes.
4. **Prototype Update**: Updated prototypes using high-confidence predictions:

$$\mathbf{p}_c^{(i)} = \alpha \mathbf{p}_c^{(i-1)} + (1 - \alpha)\bar{\mathbf{f}}_c \tag{2}$$

   where $\mathbf{p}_c^{(i)}$ is the prototype for class $c$ at iteration $i$, $\bar{\mathbf{f}}_c$ is the mean of new features extracted from $D_i$, with confidence above **0.5** for class $c$, and $\alpha$ is a weighting factor (set to **0.7**).

## 3.3 Model Updates (Task 2)

For datasets $D_{11}$ to $D_{20}$, we adapted the update procedure to account for distributional shifts:

1. Used the same feature extraction and label prediction methods as in Task 1.
2. Adjusted the prototype update parameters to be more conservative due to input distribution differences:
   - Increased the confidence threshold to **0.7** including only high-confidence predictions.
   - Set $\alpha = \mathbf{0.5}$ to give less weight to previous prototypes.

## 3.4 Justification for LwP

Learning with Prototypes is suitable for this incremental learning scenario because:

- **Simplicity**: Prototypes provide a compact representation of each class.
- **Efficiency**: Updating prototypes is computationally inexpensive.
- **Incremental Learning**: Easily accommodates new data without retraining the entire model.
- **Robustness**: By adjusting update parameters, we can mitigate the impact of distributional shifts.

# 4 Model Evaluation

We evaluated each model $f_i$ on held-out datasets $\hat{D}_1$ to $\hat{D}_i$:

1. **Feature Extraction**: Extracted features from evaluation images.
2. **Label Prediction**: Predicted labels using the model's prototypes.
3. **Accuracy Calculation**: Computed accuracy scores by comparing predicted labels with true labels.

## 4.1 Results for Task 1

The accuracy matrix for Task 1 is shown in Table 2.

Table 2: Accuracy Matrix for Task 1

| **Model** | $\hat{D}_1$ | $\hat{D}_2$ | $\hat{D}_3$ | $\hat{D}_4$ | $\hat{D}_5$ | $\hat{D}_6$ | $\hat{D}_7$ | $\hat{D}_8$ | $\hat{D}_9$ | $\hat{D}_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $f_1$ | 79.20% | | | | | | | | | |
| $f_2$ | 79.20% | 80.32% | | | | | | | | |
| $f_3$ | 79.20% | 80.32% | 78.32% | | | | | | | |
| $f_4$ | 79.20% | 80.32% | 78.32% | 79.60% | | | | | | |
| $f_5$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | | | | | |
| $f_6$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | | | | |
| $f_7$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | | | |
| $f_8$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | | |
| $f_9$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | |
| $f_10$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |

## 4.2 Results for Task 2

The accuracy matrix for Task 2 is shown in Table 3.

Table 3: Accuracy Matrix for Task 2

| **Model** | $\hat{D}_1$ | $\hat{D}_2$ | $\hat{D}_3$ | $\hat{D}_4$ | $\hat{D}_5$ | $\hat{D}_6$ | $\hat{D}_7$ | $\hat{D}_8$ | $\hat{D}_9$ | $\hat{D}_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $f_{11}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{12}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{13}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{14}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{15}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{16}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{17}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{18}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{19}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| $f_{20}$ | 79.20% | 80.32% | 78.32% | 79.60% | 79.04% | 80.68% | 78.52% | 79.16% | 80.00% | 80.24% |
| **Model** | $\hat{D}_{11}$ | $\hat{D}_{12}$ | $\hat{D}_{13}$ | $\hat{D}_{14}$ | $\hat{D}_{15}$ | $\hat{D}_{16}$ | $\hat{D}_{17}$ | $\hat{D}_{18}$ | $\hat{D}_{19}$ | $\hat{D}_{20}$ |
| $f_{11}$ | 65.68% | | | | | | | | | |
| $f_{12}$ | 65.68% | 70.68% | | | | | | | | |
| $f_{13}$ | 65.68% | 70.68% | 70.40% | | | | | | | |
| $f_{14}$ | 65.68% | 70.68% | 70.40% | 60.20% | | | | | | |
| $f_{15}$ | 65.68% | 70.68% | 70.40% | 60.20% | 75.36% | | | | | |
| $f_{16}$ | 65.68% | 70.68% | 70.40% | 60.20% | 75.36% | 64.56% | | | | |
| $f_{17}$ | 65.68% | 70.68% | 70.40% | 60.20% | 75.36% | 64.56% | 62.88% | | | |
| $f_{18}$ | 65.68% | 70.68% | 70.40% | 60.20% | 75.36% | 64.56% | 62.88% | 67.60% | | |
| $f_{19}$ | 65.68% | 70.68% | 70.40% | 60.20% | 75.36% | 64.56% | 62.88% | 67.60% | 73.60% | |
| $f_{20}$ | 65.68% | 70.68% | 70.40% | 60.20% | 75.36% | 64.56% | 62.88% | 67.60% | 73.60% | 71.00% |

### 4.3 Analysis

From the results:

- **Task 1**: The models maintain high accuracy on previous datasets, indicating that the incremental updates are effective without significant degradation.
- **Task 2**: The accuracies decrease gradually due to distributional shifts in $D_{11}$ to $D_{20}$. Adjusting the update parameters helps mitigate performance degradation.

## 5 Conclusion

We successfully implemented an incremental learning approach using Learning with Prototypes for CIFAR-10 classification. By adjusting update parameters, we accounted for distributional shifts in the datasets. The models maintained their performance on previous datasets while adapting to new data.

## Problem 2

The paper *"Deja vu: Continual Model Generalization for Unseen Domains"* (ICLR 2023), addresses the problem of continual and lifelong domain adaptation. The paper proposes novel methods to generalize models for unseen domains by leveraging continual learning approaches and domain-agnostic representations.

The following is the link to the YouTube video presentation:

`https://youtu.be/2Z3Rf0oFUZE`