Few-Shot Classification using MAML on Moving Circle Dataset

1 Problem Overview

In this assignment, we implemented Model-Agnostic Meta-Learning (MAML) for a synthetic few-shot classification problem known as the **Moving Circle Dataset**.

Each task consists of a binary classification problem where:

- Input space: $x \in R^2$

- A circle of fixed radius $r = 2$

- Center (c_x, c_y) sampled uniformly from $[-3, 3]$

The goal is to train a model that can adapt to a new unseen circle using only $K = 10$ support samples and **one gradient update**.

2 Methods Implemented

2.1 1. MAML (Meta-Learning Approach)

MAML learns an initialization θ such that after a small number of gradient steps, the model adapts quickly to a new task.

The objective is:

$$\min_{\theta} \sum_{T_i \sim p(T)} \mathcal{L}_{T_i} \left(\theta - \alpha \nabla_{\theta} \mathcal{L}_{T_i}(\theta) \right)$$

Where:

- Inner loop: One gradient descent step

- Outer loop: Update initialization based on post-adaptation performance

Thus, MAML optimizes for **fast adaptability**.

2.2 2. Baseline (Standard Joint Training)

The baseline model is trained on samples from many tasks simultaneously without meta-learning.

At test time:

- The pretrained model is fine-tuned on the support set

- Standard gradient descent is used

This approach learns an averaged decision boundary rather than a fast-adapting initialization.

3 Quantitative Comparison

We evaluated both methods by plotting:
Test Accuracy vs Number of Gradient Steps (0–10)

Observed Behavior

- MAML starts with higher initial accuracy.

- MAML improves rapidly within the first 1–2 steps.

- Baseline starts lower.

- Baseline improves more slowly.

Reason for MAML Superiority

The baseline minimizes:

$$\min_\phi \mathcal{L}(D_{train})$$

It learns a general classifier averaged over many circles.
MAML instead minimizes performance after adaptation:

$$\min_\theta \mathcal{L}_{query}(\theta - \alpha \nabla_\theta \mathcal{L}_{support})$$

Therefore:

- MAML learns parameters sensitive to task-specific updates.

- Baseline learns parameters robust across tasks.

Hence, MAML adapts faster with limited data.

4 Qualitative Comparison

Decision boundary visualizations after one gradient step show:

- Baseline produces a smooth, centralized boundary.

- MAML shifts its boundary toward the true circle location.

This demonstrates that MAML initialization is closer to the correct task-specific solution.

5 Bonus Question

Why does the meta-loss bounce up and down even when training is successful?

Explanation

In classical supervised learning:

- The same dataset is used across epochs.

- The objective function remains fixed.

- Therefore, the loss typically decreases monotonically.

However, in meta-learning:

- A new task is sampled in every iteration.

- Each task has a different circle center.

- Therefore, the loss surface changes at every epoch.

Mathematically:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}_{T_t}(\theta_t)$$

Since T_t changes each iteration, the optimization target changes as well.
Thus:

- The meta-loss does not decrease monotonically.

- It oscillates depending on task difficulty.

- However, the overall trend decreases over time.

Key Insight

Although the loss fluctuates:

- The initialization improves.

- Post-adaptation accuracy increases.

- Few-shot performance becomes better.

Therefore, bouncing loss does not indicate failure in meta-learning. It reflects optimization over a distribution of tasks rather than a single dataset.

6   Conclusion

- MAML significantly outperforms the baseline in few-shot adaptation.

- It achieves higher initial performance.

- It adapts faster with limited support data.

- Loss oscillation is natural due to task distribution sampling.

MAML successfully learns an initialization optimized for rapid adaptation across varying circular decision boundaries.