AI4CI

Distributed and Federated Learning – TP3
Understanding Federated Learning attacks and Counteract Schemes

Dmytro Rohovyi NTUU

23/05/2025

TP3: Understanding Federated Learning attacks and Counteract Schemes

Introduction

In the first practical session (TP1), you implemented a complete Federated Learning project using the FedAvg algorithm, gaining hands-on experience with the FLOWER framework. In the second practical session (TP2), you investigated the effects of data heterogeneity and client drift and explored two advanced optimization techniques, FedProx and SCAFFOLD, designed to improve robustness and convergence in non-IID scenarios.

In this third session (TP3), we shift our focus to the security challenges of Federated Learning. While FL offers strong privacy guarantees by keeping data local, it remains vulnerable to adversarial behaviors by participating clients. In particular, we will explore two common types of attacks: data poisoning and model poisoning, which aim to degrade the global model's performance for malicious goals.

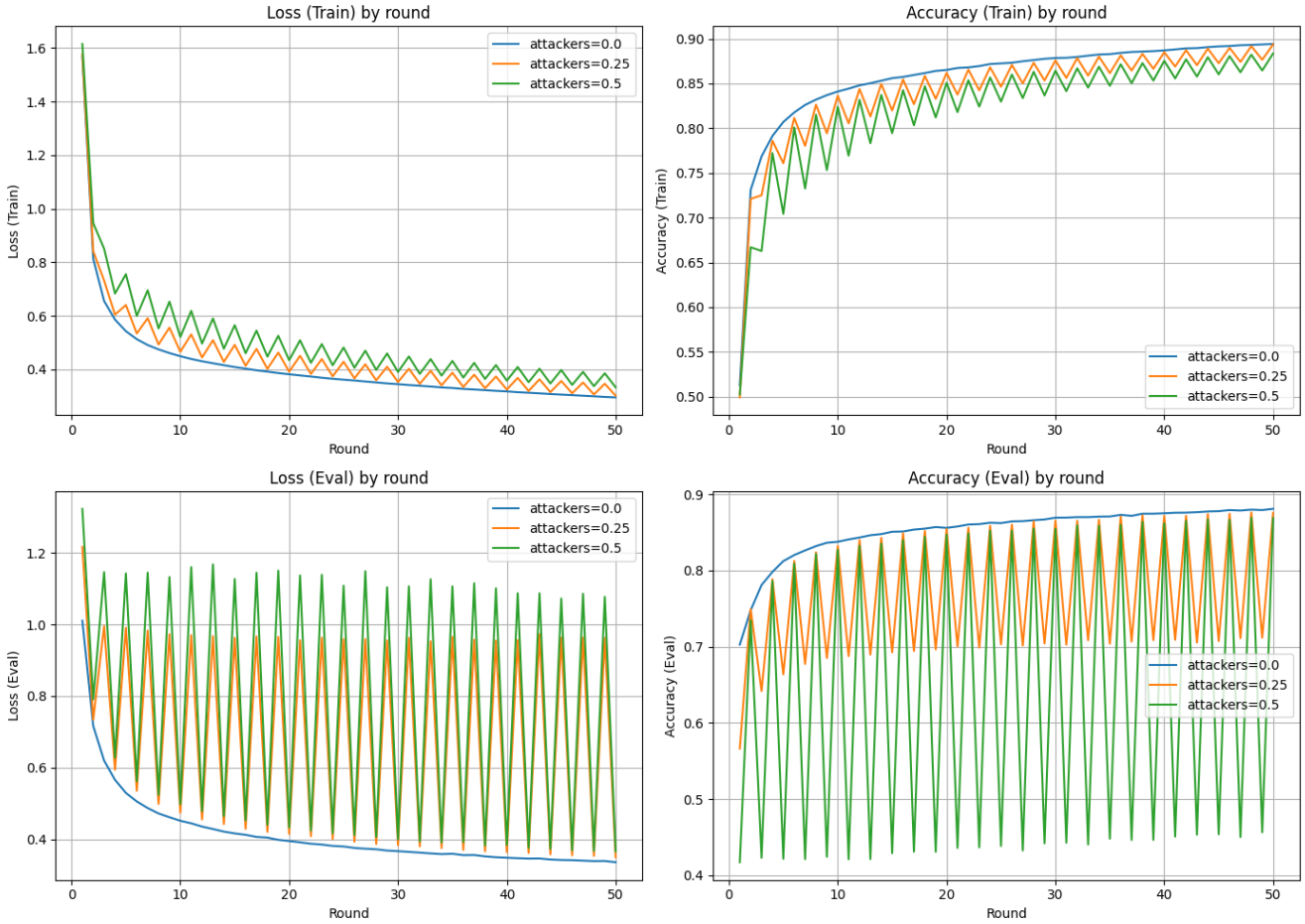
You will implement both types of attacks by modifying the client logic and simulate their impact on the training process. To defend against such attacks, we will also study and implement two robust aggregation schemes, FedMedian and Krum, which aim to mute or filter out malicious updates during aggregation. Finally, we will analyze the trade-offs these defenses introduce, especially under varying degrees of data heterogeneity.

Objectives

This TP aims to introduce you to security threats in Federated Learning and guide you through the implementation of common attack strategies and corresponding defense mechanisms. Specifically, you will:

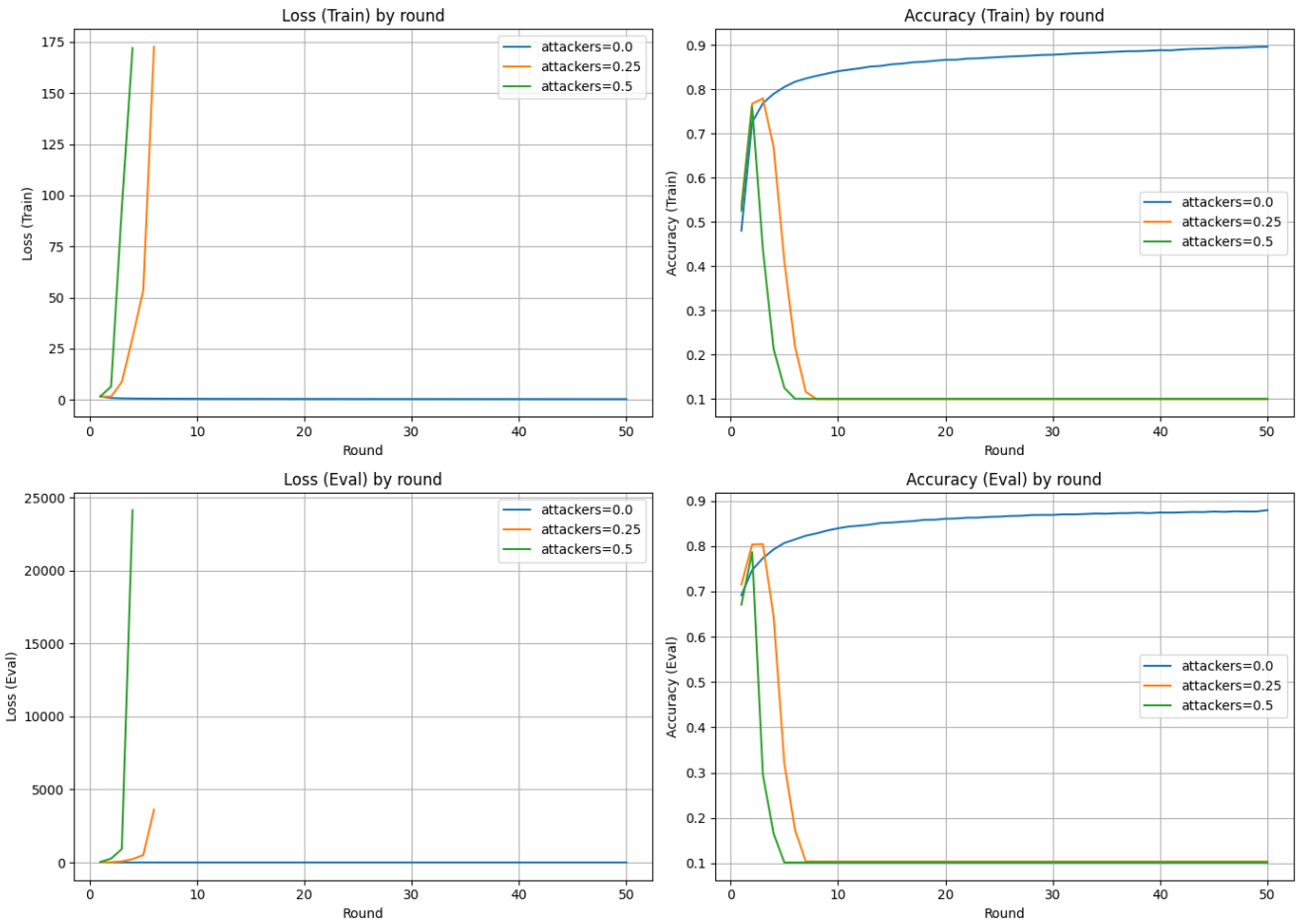
- Understand the concepts of data poisoning and model poisoning in Federated Learning, including their goals, mechanisms, and effects on global model performance.
- Implement malicious client behaviors that perform either data or model poisoning within a standard Federated Learning pipeline.
- Evaluate the impact of different proportions of malicious clients (e.g., 0%, 25%, 50%) on model performance using the FedAvg aggregation scheme.
- Implement and test two robust aggregation methods — **FedMedian** and **Krum** — and compare their effectiveness in mitigating poisoning attacks.
- Investigate how these defense mechanisms behave under different levels of data heterogeneity (using varying Dirichlet α values) and reflect on the trade-off between robustness to attacks and tolerance to legitimate variations among clients.

Data Poisoning



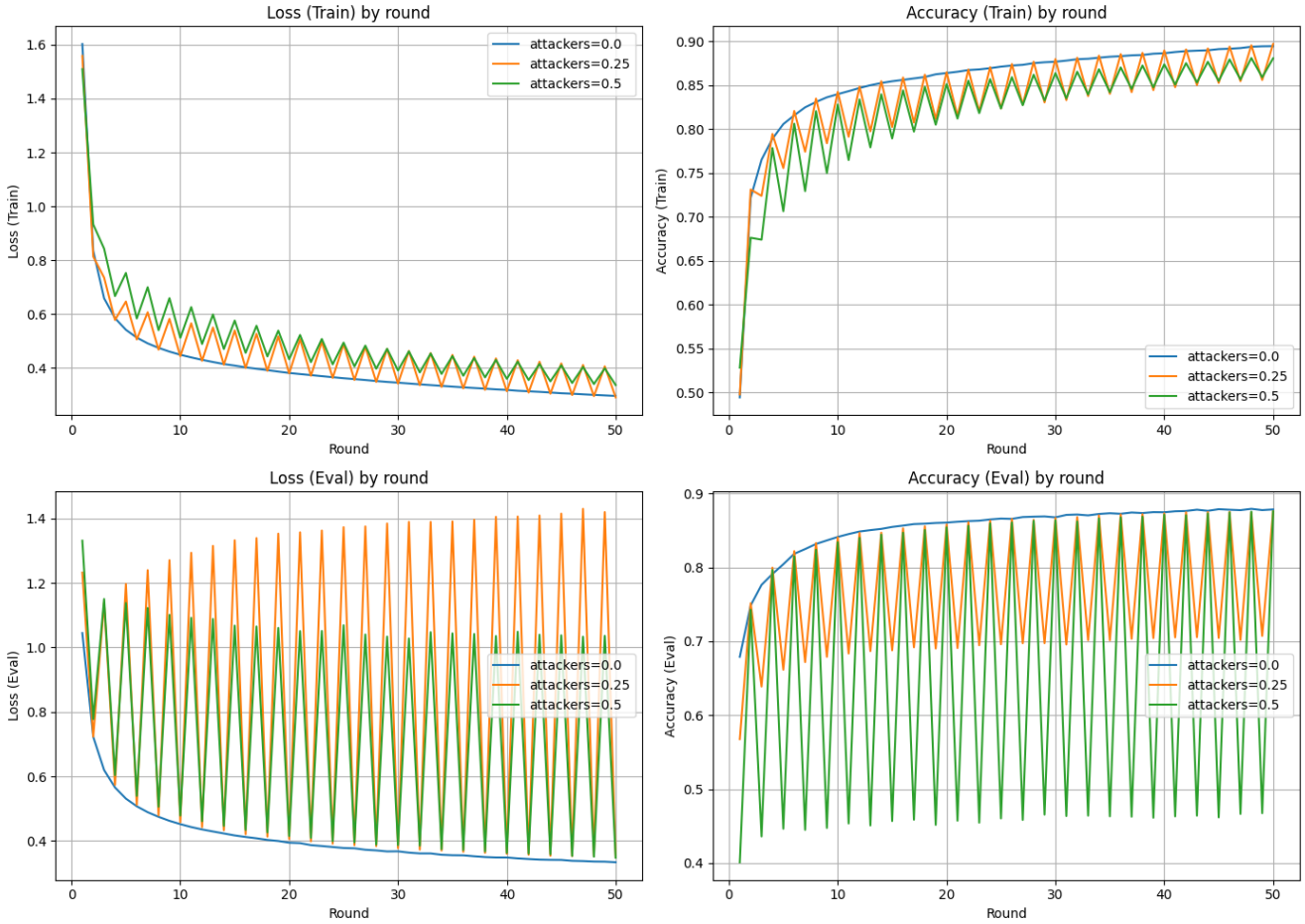
Effects of data poisoning are obvious and clearly visible on plot. Increasing number of attackers increases oscillation of global model.

Model Poisoning



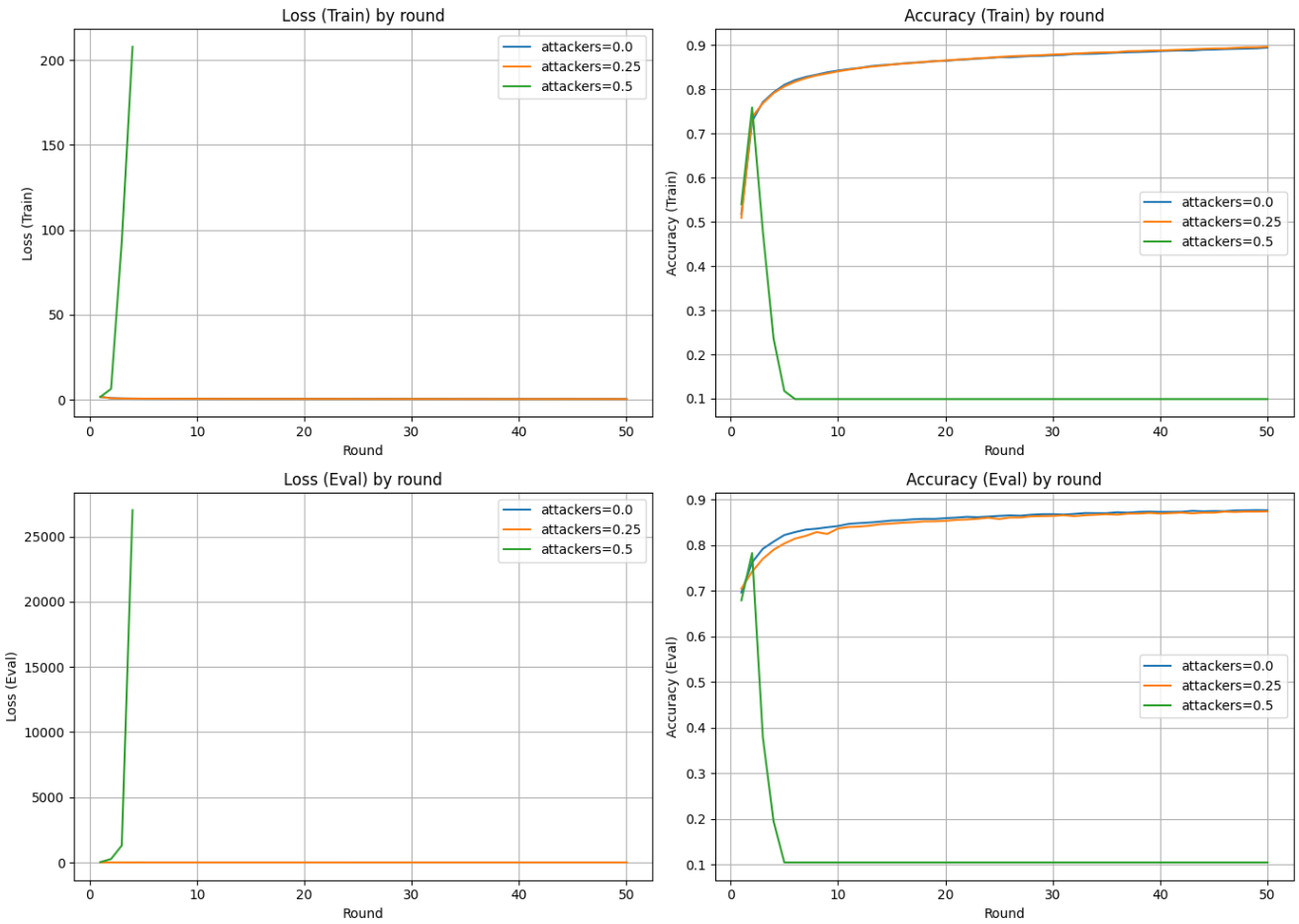
Model poisoning explodes loss function beyond float limit within first 10 rounds, stopping any training. Increasing number of attackers speeds up explosion

FedMedian Data Poisoning



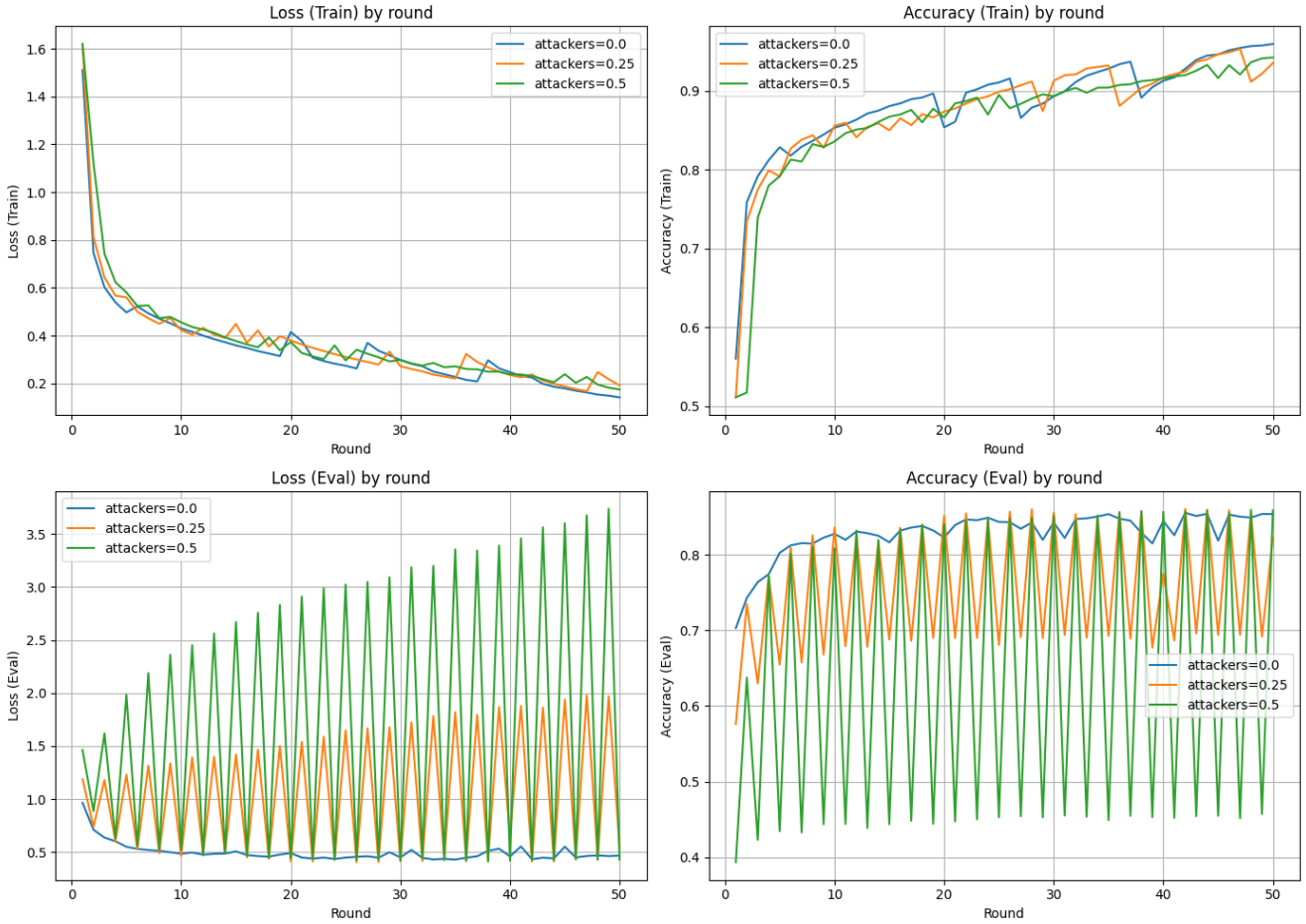
It’s clear that FedMedian can’t effectively deal with data poisoning. Comparison of all 3 methods for this will be later in report.

FedMedian Model Poisoning



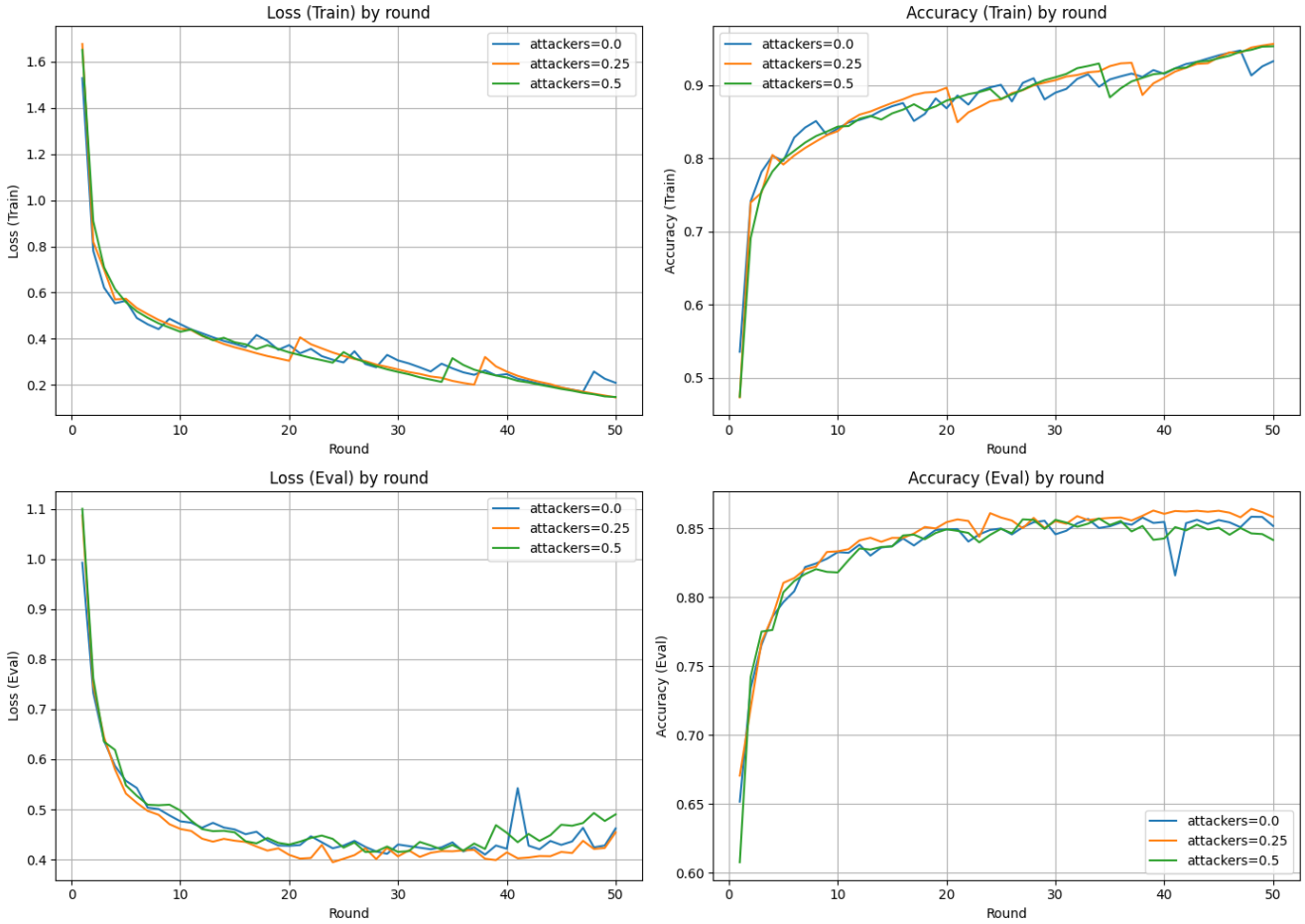
Fed median manages to fully negate any effects from model poisoning when less than half of clients are attackers(25% in our case), but fails to do so when 50% of clients are attackers.

Krum Data Poisoning



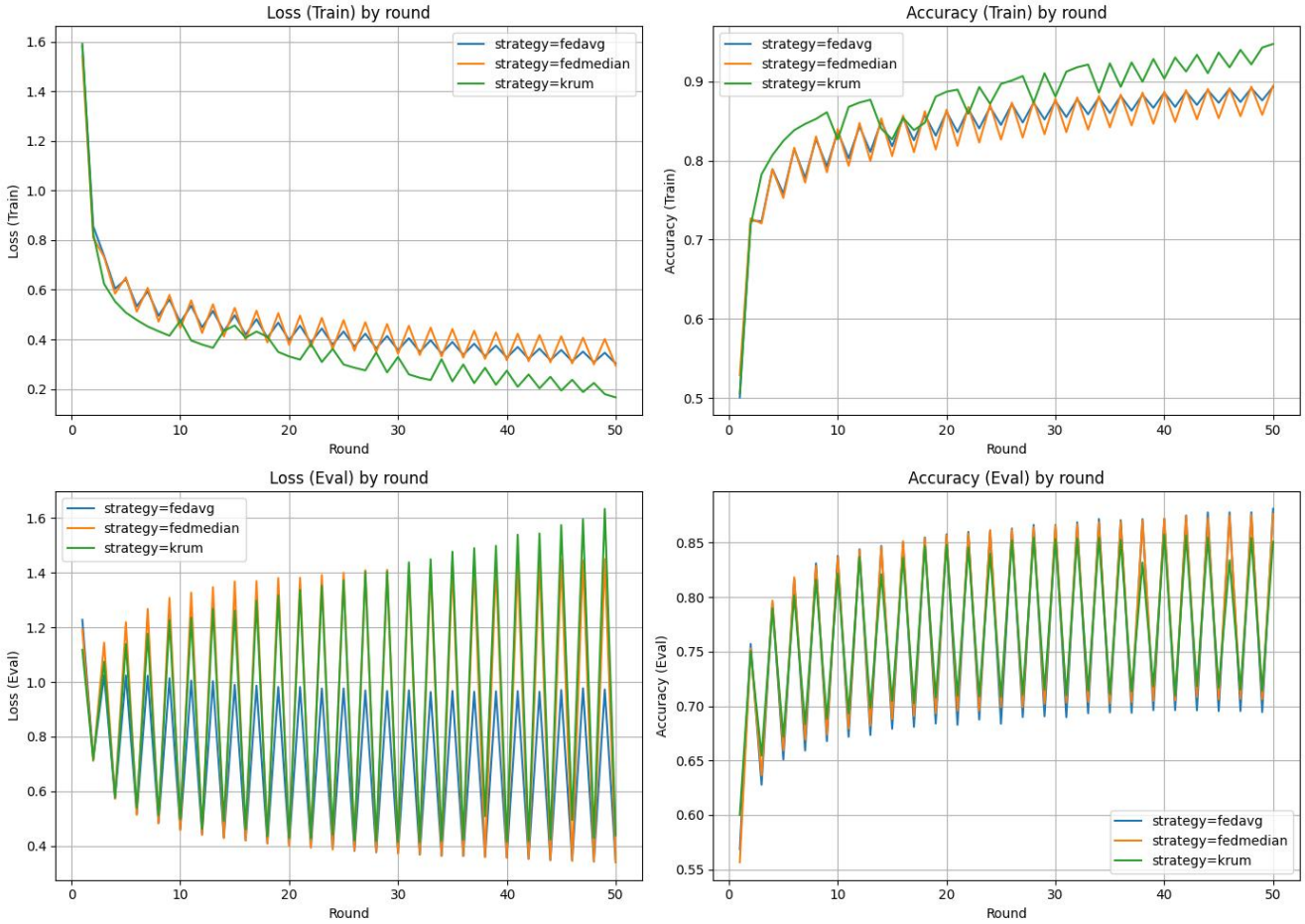
Same as FedMedian, it’s hard to determine whether Krum has any effect on data poisoning, so additional comparison will be provided.

Krum Model Poisoning



Krum effectively protects from model poisoning even when half of clients are attackers. We can see fluctuations in curves, but results are only marginally worse than Fedavg for honest training.

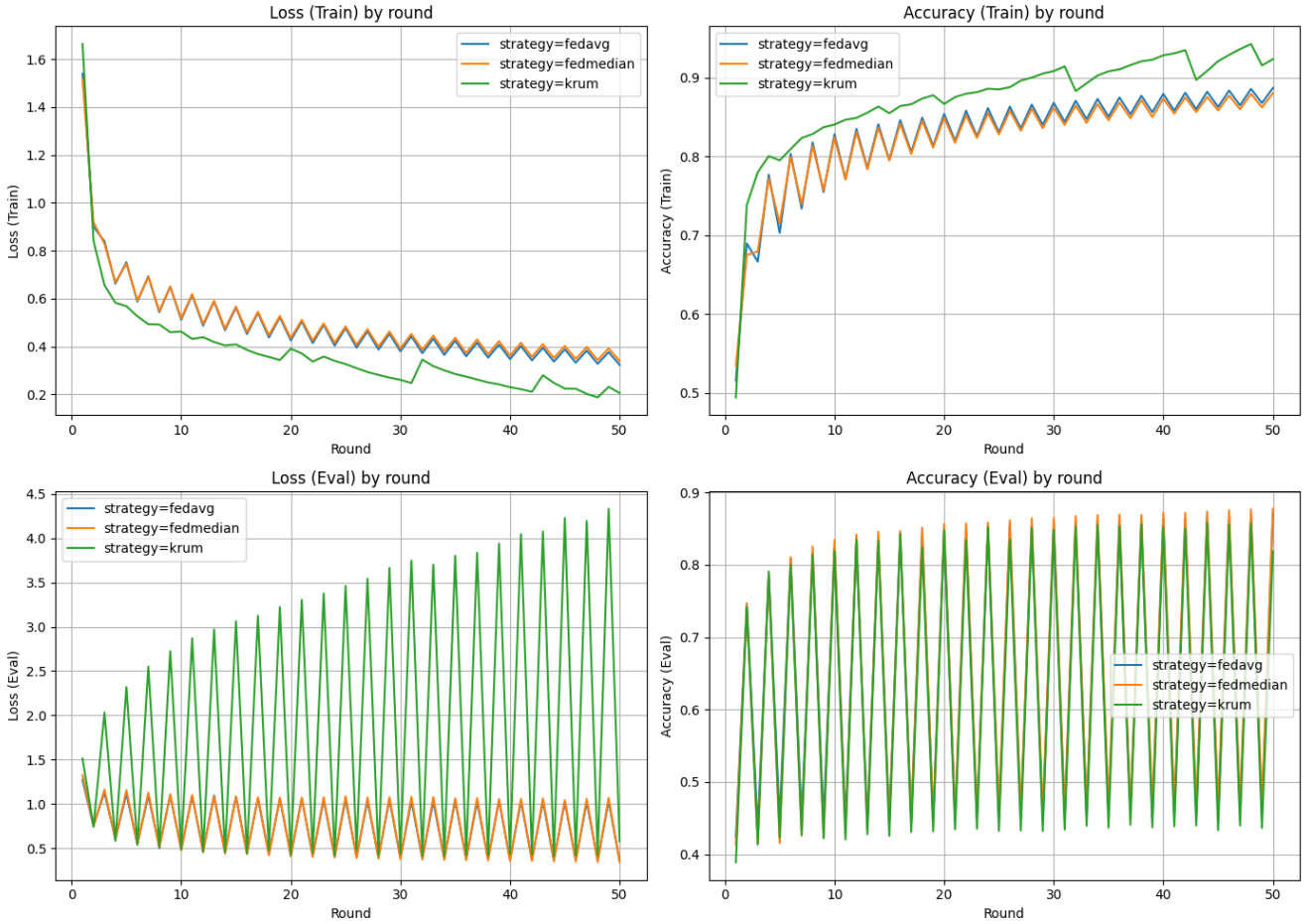
Data Poisoning Comparison for 25%



Neither of strategies were effective against data poisoning, but Krum clearly shows best results among them. I yields relatively stable training curves, while slightly dampening oscillation of accuracy but increasing oscillations of loss curve. This is result of Krum creating a model that makes consistent prediction, no matter right or wrong, and since loss function is more sensitive to high-confidence false classification it results in this oscillations.

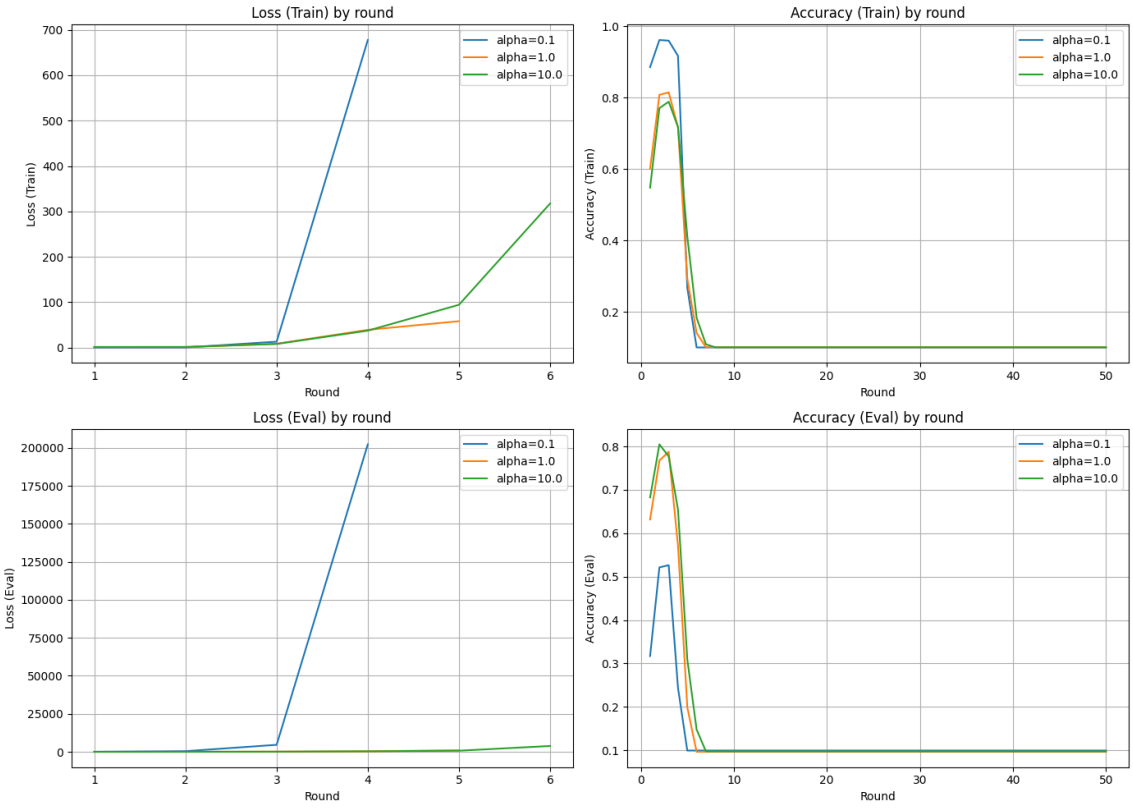
Also we clearly see how FedMedian causes more oscillations both on training and evaluation data, which shows main problem of this strategy. Even without attack, FedMedian is unstable, due to it’s way of going through parameters of model. This effect is amplified by data poisoning, which leads to worse performance than FedAvg.

Data Poisoning Comparison for 50%

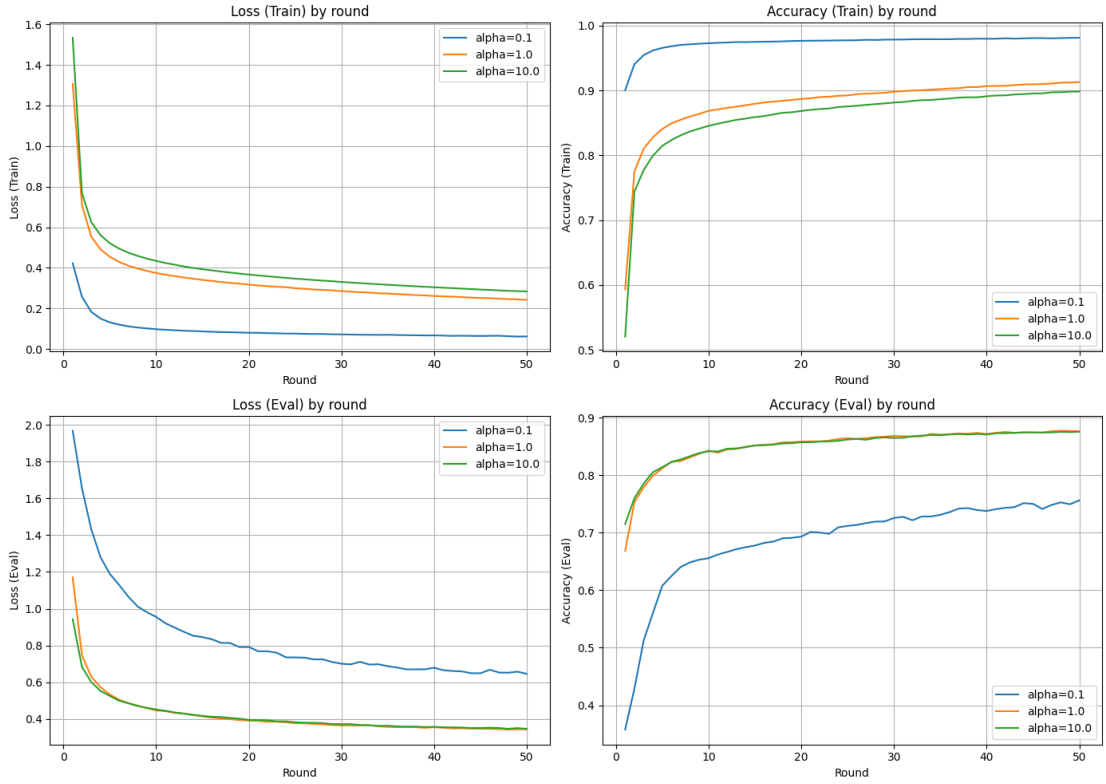


Increasing number of attackers severely degrades Krum's performance, causing it to change model closer to dishonest client's results and leading to more severe oscillations.

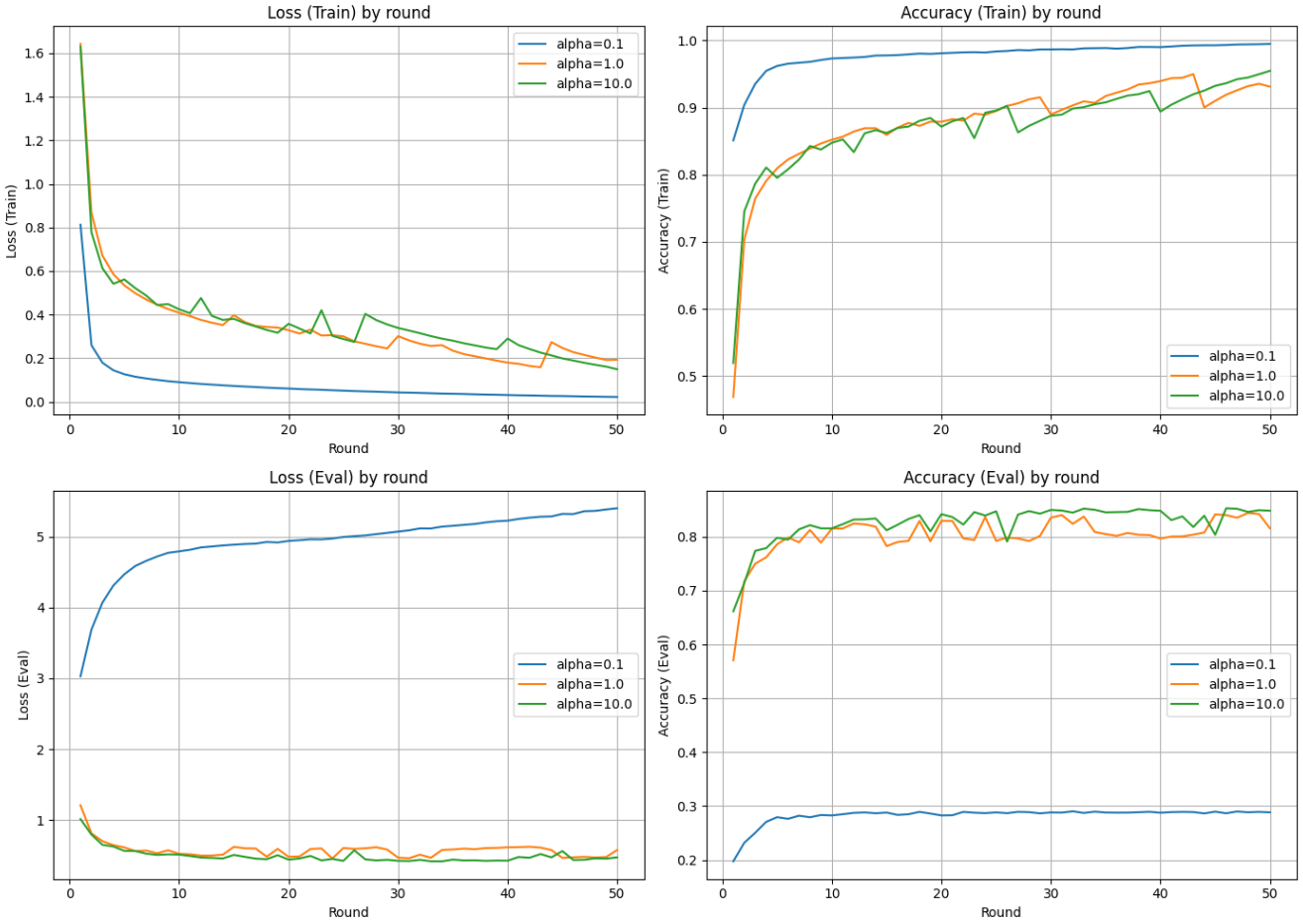
Data Heterogeneity Test for Model Poisoning



FedAvg



FedMedium



Krum

We can make several observations from this plots.

Firstly, since FedMedium completely negates Model Poisoning attacks if attackers are less than half of clients, heterogeneity impacts it's performance in same way it impacts FedAvg with no attack – higher alpha results in lower overall performance.

Krum, on the other hand, visibly suffers from low alpha much more than other algorithms, showing extremely low 30% accuracy. This is result of Krum's design. Krum always assumes that all "honest" clients will produce similar updates, and can't differentiate between honest clients producing different results due to high heterogeneity and clients, producing different results due to attacking.