Mitigating Data Injection Attacks on Federated Learning

Or Shalom[1], Amir Leshem[2], Waheed U. Bajwa[3]

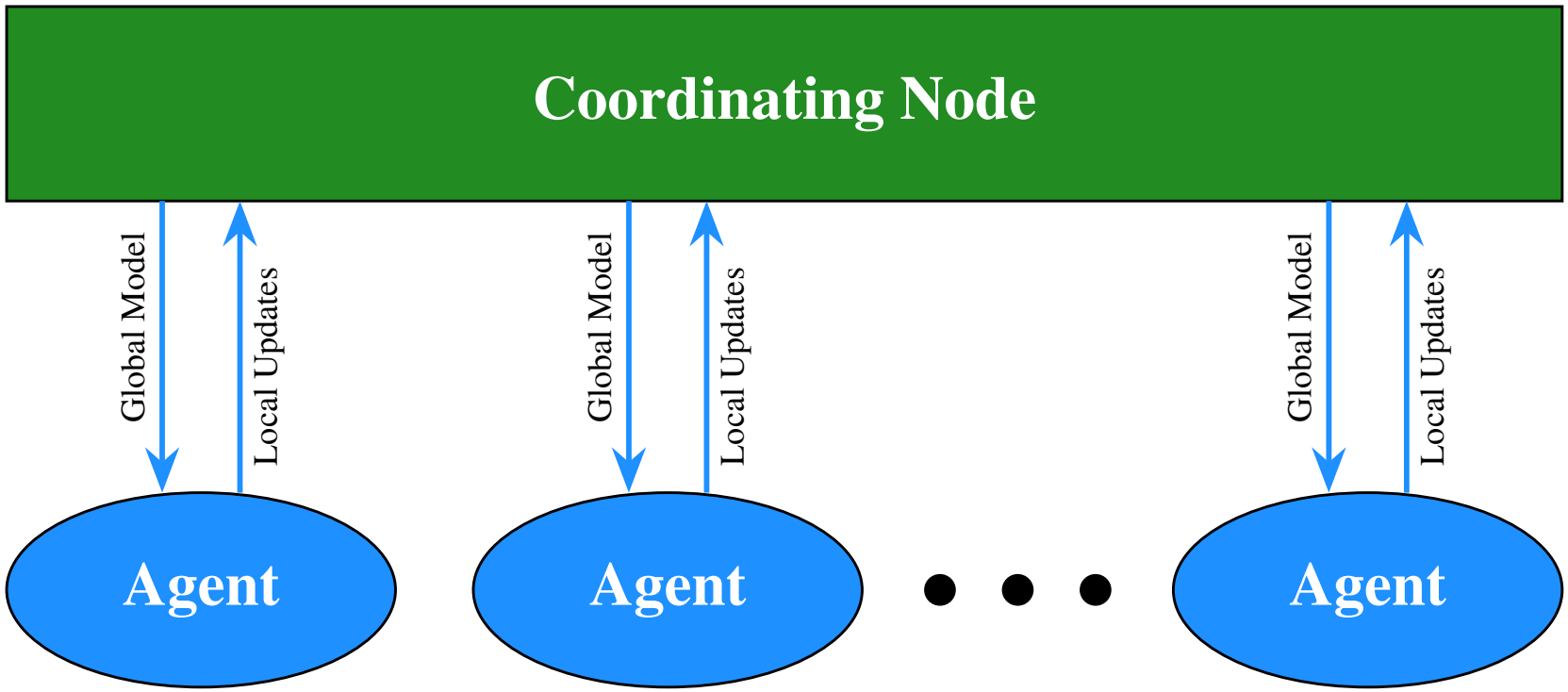
[1,2]Faculty of Engineering, Bar-Ilan University, Ramat Gan 5290002, Israel
[3]Dept. of Electrical & Computer Engineering, Rutgers University–New Brunswick, NJ 08854 USA

Accessing Data While Preserving Privacy

- Accessing data has become a challenge [1, 2].
- Some datasets are private and can't be shared (for example medical / financial records, intellectual property, etc..).
- How can we benefit from data that **isn't shared with us**?
- How can we **scale up the learning** to improve model accuracy and diversity if we can only collect limited data?

Federated Learning

- Agents perform local learning on a private model using private data.
- The agents exchange the local model parameters with the coordinating node.
- The coordinating node aggregates the local model parameters and broadcasts back a joint model.



Federated Averaging (FedAvg)

- The coordinating node averages the local parameter updates and construct a joint model.
- There is no assumption on the distribution of the private datasets (iid / non-iid).
- There is no assumption on the private model initializations.
- Each agent performs 1 or more gradient steps in the learning phase.

Problem Formulation

- Consider a dataset $\mathcal{D}$ labeled with labels from a set C.
- Within this framework, the edge agents iteratively refine their model parameters during the learning phase using this labeled dataset.
- The objective typically involves minimizing a function using a gradient descent approach:

$$\min_{W} F(W), \text{ where } F(W) := \sum_{k=1}^{N} p_k F_k(W), \tag{1}$$

  where N represents the number of participating agents, $\sum_k p_k = 1$, and F_k is the local empirical risk function for the k-th agent.
- Although $p_k = \frac{1}{N}$ is common, varying these values can prioritize the risk of certain agents.

Backdoor Attacks on Federated Learning

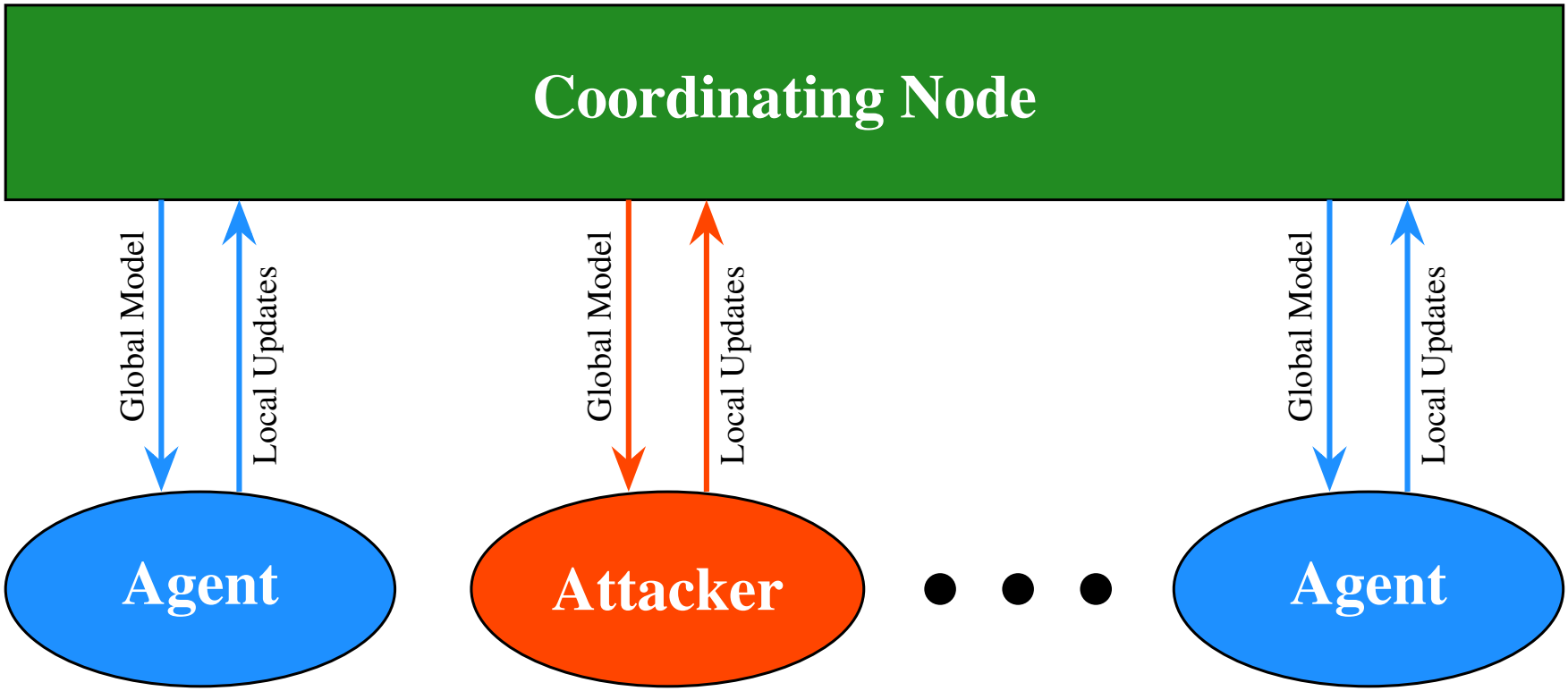
- Assume that coordinated malicious agents are present in the network.
- The malicious agents are participating in the joint model learning stages.
- The malicious agents can broadcast incorrect model parameters to the coordinating node.
- We have previously shown [3] that this malicious behavior forces the joint model to converge to the malicious agents' desired model.

Backdoor Attacks on Federated Learning

Attack Types:

- Data Poisoning Attacks - A malicious agent modifies the labels during the local learning stages, i.e. learning a new model using incorrect labels.
  - Constant Output Attack - The malicious model constantly outputs the same label.
  - Label-Flipping Attack - The malicious model flips some or all of the labels.
- Model Poisoning Attacks - A malicious agent skips the learning stage and broadcasts a previously learned model.
- Clean Label Attacks - A malicious agent modifies the data during the local learning stages, leaving the labels untouched.

Backdoor Attacks on Federated Learning



Attack Scheme

- Consider a scenario where some agents participating in federated learning are malicious.
- Denote the set of attackers $A \subset 1, \ldots, N$ and let $n_a = |A|$.
- We assume that $0 \leq n_a < N/2$, and $n_t = N - n_a$ is the number of trustworthy agents.
- w.l.o.g we assume there is a single malicious agent, say agent a, influencing the joint model training.
- Agent a's goal is to manipulate the joint model training by transmitting false parameters.

Attack Scheme

- To reduce the statistical discrepancy between the malicious agents' response and the other agents' response, the attacker combines a false model and a true one, thus adding a bias to the reported model.

- While the attacker's main goal is to manipulate the joint model parameters and prevent convergence to a steady optimal point, it also has a secondary goal of remaining hidden and disguising the attack.

Attack Scheme

- Let $W_{a,r}(t)$ represent the model parameters that the attacker, posing as a regular agent, would have updated by reliably updating the model $W(t-1)$ provided by the coordinating agent with correct data at time t.
- Mark with $W_{a,f}$ a pre-trained false model, classifying labels according to the attacker's desired attack scheme.

Building on this concept, the attack can be formally described as follows: A malicious agent a responds at time t by sending

$$W_a(t) := g(t)W_{a,r}(t) + (1 - g(t))W_{a,f}, \tag{2}$$

where $W_a(t)$ denotes the set of parameters transmitted by agent a at time t, and $g(t)$ is a non increasing series, varying from 1 to 0. While the monotonicity of $g(t)$ can be relaxed, it is essential for ensuring convergence to the attacker's desired model.

Attacker Detection

Defense mechanisms against backdoor attacks in federated learning can take place in different phases of the learning process:

- Pre Aggregation - The coordinating node aims to detect the attacker prior to averaging it's model parameters, thus allowing for trustworthy parameters aggregation only at each stage of the learning process.
- In Aggregation - The coordinating node uses a more robust aggregation technique while joint model updating procedure is conducted (such as byzantine techniques).
- Post Aggregation - The coordinating node aims to repair the joint model after completing the aggregation process.

The success of any detection method relies on the attacker being unfamiliar with the detection method chosen or the timing and duration in which it takes place.

Attacker Detection and Avoidance

- The coordinating agent compares the updates received from edge agents over time.
- The private datasets are assumed to be identically distributed and therefore if an agent is malicious and its model parameters update differently, it will stand out and be considered malicious.
- To localize the attacker, we propose a low-complexity metric, computed over time by the coordinating agent once every ΔT updates.
- When the coordinating agent suspects an edge agent to be an attacker, it ignores its parameter updates for the next ΔT updates.

Attacker Detection and Avoidance

Define the two hypotheses tested over the interval $I_k = [(k-1)\Delta T + 1, k\Delta T]$:

$$\mathcal{H}^0_{j,k} \text{ – agent } j \text{ is trustworthy.}$$
$$\mathcal{H}^1_{j,k} \text{ – agent } j \text{ is malicious.}$$

The proposed detection metric for a given interval I_k, computed over time for agent j's model parameters, is given by

$$\Delta U_{j,k} := \frac{1}{\Delta T} \sum_{t \in I_k} U_j(t) \underset{\mathcal{H}^1_j}{\overset{\mathcal{H}^0_j}{\lessgtr}} \delta_u \sqrt{N}, \tag{3}$$

$$U_j(t) := \|\Delta W_j(t) - \text{median}\{\Delta W_\ell(t) : \ell \in \{1, \ldots, N\} \setminus \{j\}\}\|_\infty. \tag{4}$$

Here, the median is a coordinatewise operation, $\Delta W_j(t) := W_j(t) - W_j(t-1)$, and δ_u is a predefined threshold.

Attacker Detection Success Rate

- The proposed detector facilitates continuous operation, unaffected by the convergence time of the joint model.
- Let d_k denote the outcome of applying (3) on interval I_k. This results in a sequence of decisions $d_1, d_2, \ldots$, where $d_i = 1$ if the examined agent crosses the threshold, and $d_i = 0$ otherwise.
- At the conclusion of $K\Delta T$ updates, the coordinating agent assesses each edge agent's behavior. If an edge agent's average decision score over these K intervals, calculated as $\frac{1}{K}\sum_{k=1}^{K} d_k$, exceeds $1/2$, the agent's input is excluded for the next segment I_{K+1}.
- Nonetheless, the coordinating agent continues to compute the statistics (3) during this period and the agent is added back to the list of trustworthy agents if

$$\frac{1}{K}\sum_{k=1}^{K} d_k < \frac{1}{2}. \tag{5}$$

Attacker Detection Success Rate

The validity of this approach is encapsulated in the following lemma:

Lemma

Assume we set δ_u, ΔT such that for each k, $P_{FA}(I_k) < 1/2 < P_D(I_k)$. Then, with probability 1, there exists a sufficiently large k_0 such that the presented scheme (cf. (3)–(5)) ignores all the malicious agents after time $k_0 \Delta T$, while ensuring that updates from all trustworthy agents are incorporated beyond this time.

The proof of Lemma 1 is based on a sequence of decisions where for each interval I_k of length ΔT the detector is applied to obtain a decision d_k. Then a majority among all prior decisions is used to decide whether to disconnect the agent. By the assumption $P_{FA}(I_k) < \frac{1}{2} < P_D(I_k)$ and the Borel Cantelli lemma the proof follows.

Simulations

The MIT-BIH Arrhythmia dataset

- The MIT-BIH Arrhythmia dataset [4, 5] is a sample set of ECG strips, derived from over 4000 long-term Holter recordings (48 subjects aged 23 to 89) that were obtained by the Beth Israel Hospital Arrhythmia Laboratory between 1975 and 1979. The MIT-BIH includes 17 labels including 'Supraventricular tachyarrhythmia' (SVTA), 'Idioventricular rhythm' (IVT), and many more. An example taken from the MIT-BIH dataset can be seen in Figure 1.

MIT-BIH Arrhythmia Constant Output Attack

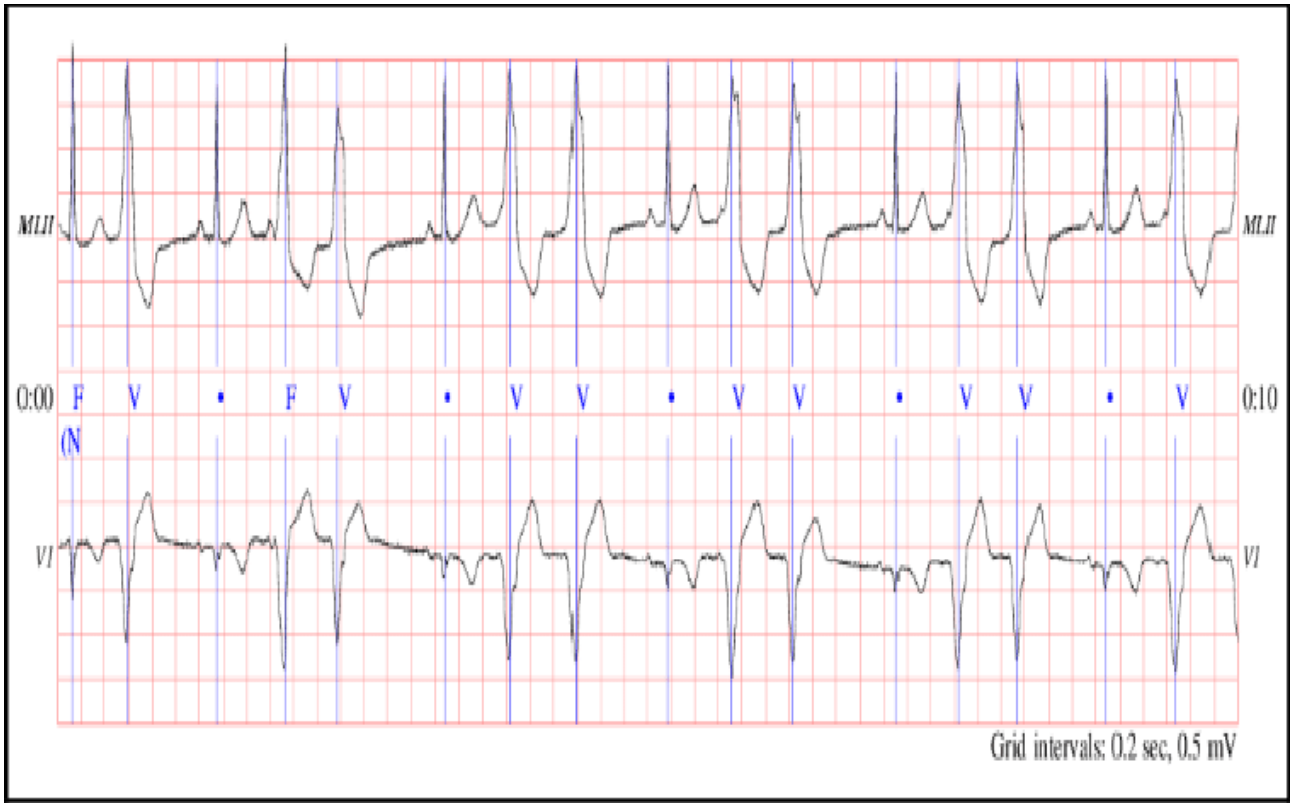


Figure 1: Example (10s) of annotations in MIT-BIH database [4].

MIT-BIH Arrhythmia Constant Output Attack

- In this example, the agents are attempting to learn a classifier for the ECG MIT-BIH dataset, while the attacker (w.l.o.g marked as agent 1) is aiming to inject a model that consistently outputs 'Supraventricular tachyarrhythmia' (SVTA).
- In this example we have 4 trustworthy agents in the network, the dataset is divided between the agents in a non-iid manner such that each agent $2, 3, 4, 5$ receives records from 10 different subjects, with no overlap.
- Note that each record is individually labeled thus each agent may be exposed to all 17 labels.

MIT-BIH Arrhythmia Constant Output Attack

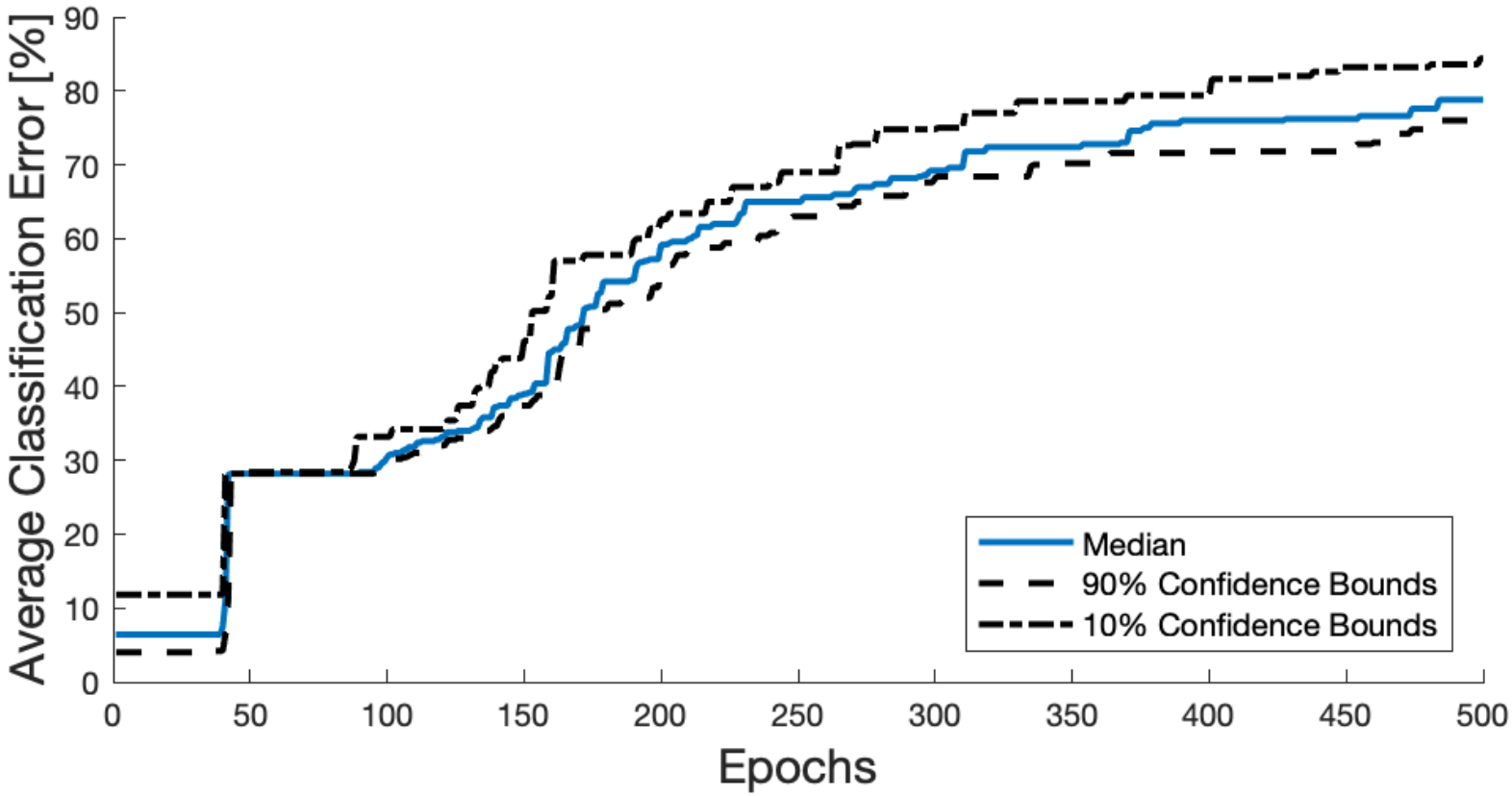


Figure 2: Classification Error Without Detection (100 Experiments)

MIT-BIH Arrhythmia Constant Output Attack

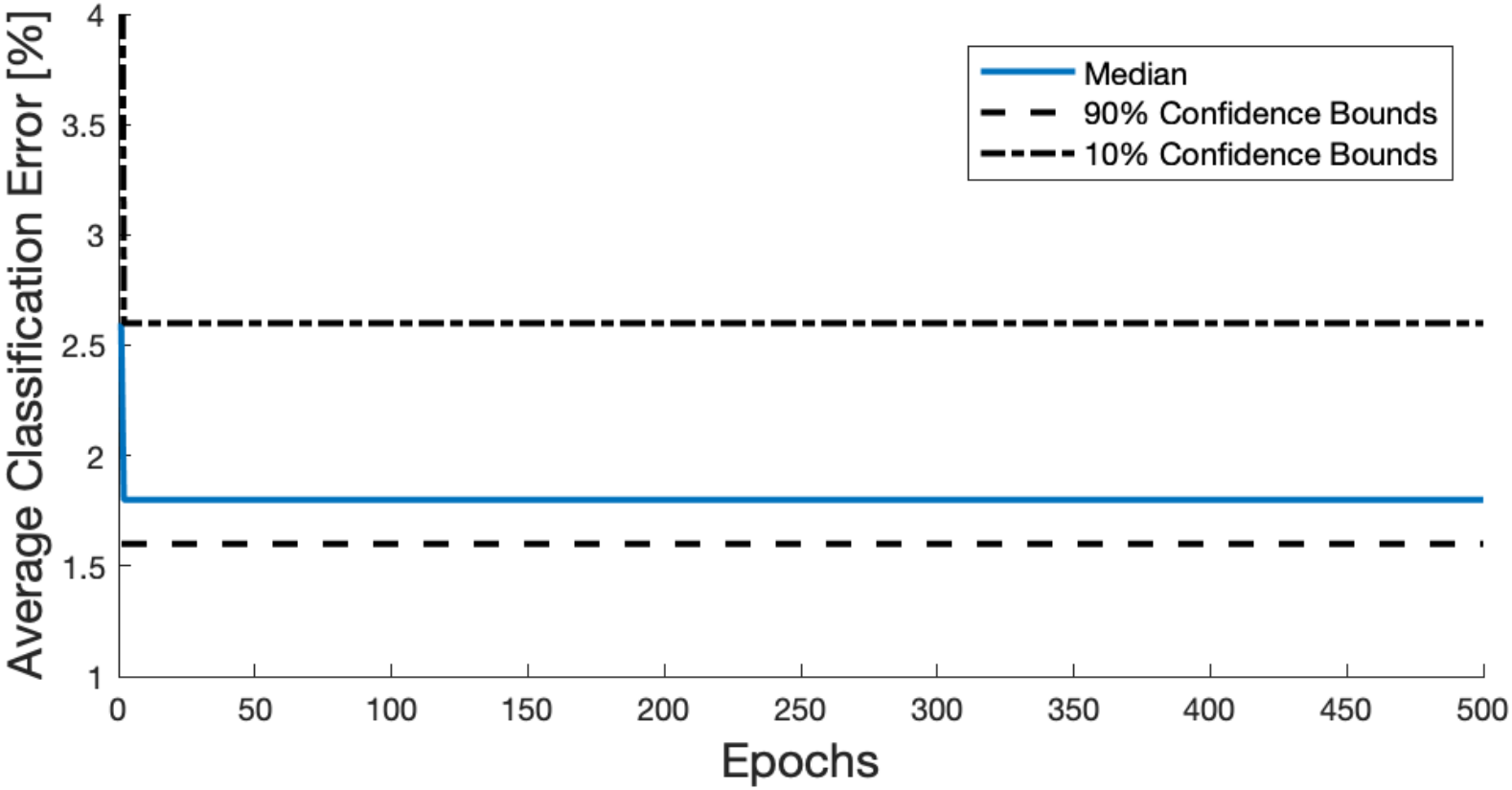


Figure 3: Classification Error with Detection (100 Experiments).

Conclusions

- We have presented a robust federated learning algorithm that can operate in the presence of data injection attacks.
- We have provided conditions for the identification of malicious agents.
- We have demonstrated the performance of the proposed technique on various attacks.
- Detailed proofs of the lemmas as well as bounds on the attacker detection probability and the false-alarm probability are presented in an extended version of this work.

[1] B. Kaneshiro, O. Geling, K. Gellert, and L. Millar, "The challenges of collecting data on race and ethnicity in a diverse, multiethnic state," *Hawaii medical journal*, vol. 70, no. 8, p. 168, 2011.

[2] D. Casey, "Challenges of collecting data in the clinical setting," *NT Research*, vol. 9, no. 2, pp. 131–141, 2004.

[3] O. Shalom, A. Leshem, and A. Scaglione, "Localization of data injection attacks on distributed m-estimation," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 655–669, 2022.

[4] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[5] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE engineering in medicine and biology magazine*, vol. 20, no. 3, pp. 45–50, 2001.