Block ModShift: Model Privacy via Dynamic Designed Shifts

Nomaan A. Kherani
Electrical and Computer Engineering
University of Southern California
kherani@usc.edu

Urbashi Mitra
Electrical and Computer Engineering
University of Southern California
ubli@usc.edu

Abstract

The problem of mutli-shot model privacy against an eavesdropper (Eve) in a distributed learning environment is investigated. The solution is found via evaluating the Fisher Information Matrix (FIM) for the model learning problem for Eve. Through a model shift design process, the eavesdropper's FIM can be driven to singularity, yielding a provably hard estimation problem for Eve. The solution has time-varying shifts that prevent Eve from using the temporal correlation of the updates to aid her in her estimation. A convergence test for Eve is designed to determine if model updates have been tampered with. However, under a bounded gradient dissimilarity assumption, the Block ModShift strategy passes the test and thus the shifts are not detectable. Block ModShift is compared against a noise injection scheme and shown to offer superior performance. We numerically show the efficacy of Block ModShift in preventing temporal leakage in a setup biased towards Eve's learning ability where she uses Kalman smoothing to estimate updates.

1 Introduction

In federated learning (see *e.g.* [8]), agents provide local model information to a global server while maintaining privacy of the local data resident at each agent. However, inferences about an agent's data can still be made exploiting the shared local model updates, motivating the development of additional strategies to ensure data privacy [3, 7, 1, 2]. Schemes like secure aggregation [3, 7, 21], and differential privacy [1, 12, 22] address privacy of user data, but do not address the privacy of the overall global model.

Recent work has begun to address the issue of **model privacy**[18, 5, 17, 11]. In [18], the model is protected from the participating agents, but not eavesdroppers. In contrast, [5] protects the model from eavesdroppers, but does not enable agent model learning. While [17] protects model privacy from eavesdroppers, a very constrained scenario is considered: only a link between a single agent and the global server (amongst many) can be compromised in order for the scheme to work. Herein, we consider the problem of model privacy when all uplinks between the agents and the global server are eavesdropped.

Our approach is inspired by the creation of statistically hard estimation problems for the eavesdropper through signal shaping [4, 15, 14, 16]. In [4], wireless communications are made private through randomizing the *structure* of the modulation. In [15, 14, 16], localization is made private by modifying the channel perceived by the eavesdropper. The FIM of the estimation problem undertaken by the eavesdropper is driven to singularity through transmitted signal precoding. Such an approach is undertaken herein for the new problem of model privacy in federated learning or distributed

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI and ML for Next-Generation Wireless Communications and Networking (AI4NextG).

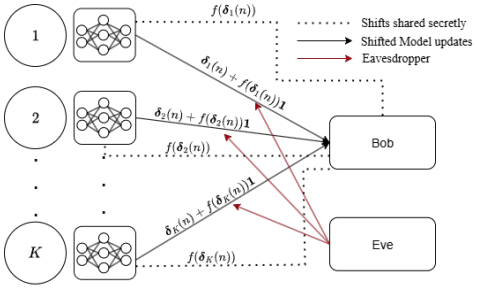


Figure 1: System model for distributed optimization, model shift and eavesdropping.

optimization. As in [4, 15, 16], only a modest amount of information is shared with the global server by the agents in order to achieve model privacy.

ModShift [11] introduces intentional model shifts in the updates shared with the global server, effectively introducing model shifts into the perceived distributions of the data at each agent. However, ModShift does not account for temporal correlation of updates which can be used by Eve to aid her estimation problem. We propose Block ModShift which builds on ModShift and tackles the problem of temporal correlation between updates by strategically selecting shift directions.

2 System Model

We consider a network consisting of K agents that communicate with a central server (Bob) to collectively learn a global model $\boldsymbol{w}^* \in \mathbb{R}^d$. An unauthorized receiver (Eve) tries to learn $\boldsymbol{w}^*$ by eavesdropping on the uplink communication between the agents and the server. Each agent has a local dataset $\boldsymbol{D}_k \in \mathbb{R}^{(d+1)\times m_k}$ comprised of realizations of feature vectors, $\boldsymbol{x}_{k,i} \in \mathbb{R}^d$, and their corresponding labels $y_{k,i}$. Let $m = \sum_{k=1}^{K} m_k$.

The function $l(\boldsymbol{w}, \boldsymbol{x}_{k,i}, y_{k,i})$ represents the loss of the ith data sample of agent k. Our aim is to find the $\boldsymbol{w}^*$ that minimizes the global loss function at the server

$$F(\boldsymbol{w}) = \frac{1}{m} \sum_{k=1}^{K} \sum_{i=1}^{m_k} l(\boldsymbol{w}, \boldsymbol{x}_{k,i}, y_{k,i}). \tag{1}$$

The gradient computed by agent k on dataset $\boldsymbol{D}_k$ with a weight $\boldsymbol{w}$ is given by,

$$\boldsymbol{g}_k(\boldsymbol{w}, \boldsymbol{D}_k) = \sum_{i=1}^{m_k} \frac{\nabla l(\boldsymbol{w}, \boldsymbol{x}_{k,i}, y_{k,i})}{m_k}. \tag{2}$$

We use FedAvg [19] with a slight modification to find the parameter w^*. In iteration n, after performing R steps of local gradient descent with weight $\boldsymbol{w}(n)$, the devices share the *difference* between their resulting local model $\boldsymbol{w}_{k,R}(n)$ and the global model $\boldsymbol{w}(n)$. This formulation introduces an initial model shift due to random initialization and necessitates that Eve eavesdrop in every communication round to attempt to reconstruct the model trajectory.

We assume that each agent uses an orthogonal channel for transmission of $\boldsymbol{\delta}_k(n)$ to Bob, such as orthogonal frequency-division multiplexing (OFDM) [6]. Under the assumption of white Gaussian noise across the sub-channels and flat fading with known channel state information for both Bob and Eve, the effective received signals per orthogonal channel in iteration n for each receiver is

$$\underline{\boldsymbol{y}}_k^u(n) \quad = \quad \boldsymbol{\delta}_k(n) + \underline{\boldsymbol{z}}_k^u(n), \tag{3}$$

where $\underline{\boldsymbol{z}}_k^u(n) \sim \mathcal{CN}(0, \frac{\sigma_u^2}{h_k^u(n)^2}\boldsymbol{I})$ is the distribution of the noise conditioned on the known channel state $h_k^u(n)$ and $u \in \{B, E\}$. Both Bob and Eve wish to estimate the global model w^* from their received signals.

In the sequel, we modify the transmitted signals, sharing this change over a secure channel [1] with Bob, to mislead Eve to the incorrect model.

3 Block ModShift

The shift designs proposed in [11] exploit a single $\underline{\boldsymbol{y}}_k^E(n)$ for fixed k to estimate $\boldsymbol{\delta}_k(n)$. However, updates are temporally correlated and Eve may use this to aid her estimation. Thus, we propose a block version of ModShift to minimize temporal leakage. We make the following assumption:

Assumption 1. *Eve treats $\boldsymbol{\delta}_k(n)$ as a deterministic unknown.*

Eve uses the following N observations to estimate $\boldsymbol{\delta}_k(n)$

$$\underline{\boldsymbol{y}}_k^E(n+i) = \boldsymbol{S}_k(n+i)\boldsymbol{\delta}_k(n+i) + \underline{\boldsymbol{z}}_k^E(n+1), \tag{4}$$

where $i \in \{0, \cdots N-1\}$, $\boldsymbol{S}_k(j) \doteq \left[\boldsymbol{I} + (\boldsymbol{u}_k(j)\boldsymbol{\gamma}_k(j)^T)\right] \in \mathbb{R}^{d\times d}$ and the shift introduced by agent k in round j is given by $\boldsymbol{u}_k(j)\boldsymbol{\gamma}_k(j)^T\boldsymbol{\delta}_k(j)$. In the sequel, we propose a design for $\boldsymbol{\gamma}_k(j)$ and $\boldsymbol{u}_k(j)$ to pose a provably hard estimation problem for Eve when she uses the N observations given in (4) for her estimation.

The exact functional relation between $\boldsymbol{\delta}_k(n+1)$ and $\boldsymbol{\delta}_k(n)$ is complicated to express and is also a function of $\boldsymbol{\delta}_i(n) \forall i \neq k$ and $\boldsymbol{w}(n)$. Thus, to overcome these challenges, we propose a worst case design for our shifts against a strong adversary who has access to $\boldsymbol{\delta}_i(n) \forall i \neq k$ and $\boldsymbol{w}(n)$. We let

$$\boldsymbol{\delta}_k(n+1) = \phi_{k,n}(\boldsymbol{\delta}_k(n)) \tag{5}$$

where $\phi_{k,n}(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$. For clarity of exposition, we drop the reference to $\boldsymbol{\delta}_i(n) \forall i \neq k$ and $\boldsymbol{w}(n)$ from $\phi(\cdots)$ since we assume that Eve has access to these values; thus, we treat these values as constants. Note that while knowledge of $\boldsymbol{\delta}_i(n) \forall i \neq k$ and $\boldsymbol{w}(n)$ may be enough information to recover $\boldsymbol{\delta}_k(n)$, we are only concerned with designing shifts that pose a hard estimation problem for Eve when she uses $\boldsymbol{\delta}_k(n+1)$ to estimate $\boldsymbol{\delta}_k(n)$.

3.1 Shift Design

Consider $R = 1$ where R is the number of local gradient descent steps. We denote by $\boldsymbol{J}_{T_i}(\boldsymbol{\delta}_k(n))$ the FIM of $\boldsymbol{\delta}_k(n)$ from $\underline{\boldsymbol{y}}_k^E(n+i)$. The combined FIM of $\boldsymbol{\delta}_k(n)$ from the N observations given in the signal model in Equation (4) is given by

$$\boldsymbol{J}_N(\boldsymbol{\delta}_k(n)) = \sum_{i=0}^{N-1} \boldsymbol{J}_{T_i}(\boldsymbol{\delta}_k(n)), \tag{6}$$

Our aim is to design $\boldsymbol{u}_k(j), \boldsymbol{\gamma}_k(j)$ for $j \in \{n, \cdots, n+N-1\}$ such that $\boldsymbol{J}_N(\boldsymbol{\delta}_k(n))$ is singular.

Proposition 1. *$\boldsymbol{J}_N(\boldsymbol{\delta}_k(n))$ is singular when*

1. $\boldsymbol{\gamma}_k(n+j)^T\boldsymbol{u}_k(n+j) = -1 \;\; \forall j \in \{0, \cdots, N-1\}$,
2. $\frac{\partial \phi_{k,n+j}(\boldsymbol{\delta}_k(n+j))}{\partial \boldsymbol{\delta}_k(n+j)}\alpha_k(n+j)\boldsymbol{u}_k(n+j) = \boldsymbol{u}_k(n+j+1) \;\; \forall j \in \{0, \cdots, N-1\}$,

where $\alpha_k(n+j)$ is a non-zero scalar which can be chosen to control the magnitude of $\boldsymbol{u}_k(n+j+1)$.

Proof. See Appendix A.1 □

Condition 1 in Proposition 1 can be met by properly designing $\boldsymbol{\gamma}_k(n)$, however, Condition 2 requires knowledge of the function $\phi_{k,n}(\cdot)$. We now find an approximation for $\phi_{k,n}(\cdot)$ and thus $\frac{\partial \phi_{k,n}(\boldsymbol{\delta}_k(n))}{\partial \boldsymbol{\delta}_k(n)}$ for $R = 1$.

[1] Note that communication privacy for a modest amount of shared information can be achieved via the methods in [4, 15].

Proposition 2. *For a first order Taylor series approximation of $\phi_{k,n}(\cdot)$,*

$$\frac{\partial \phi_{k,n}\left(\boldsymbol{\delta}_k(n)\right)}{\partial \boldsymbol{\delta}_k(n)} \approx \boldsymbol{I} - \eta \frac{m_k}{m} \boldsymbol{H}_k(n), \tag{7}$$

where $\boldsymbol{H}_k(n)$ is the Hessian of the local loss function evaluated at $\boldsymbol{w}(n)$.

Proof. See Appendix A.2 □

Note that when working with $R > 1$, each device must keep track of the shift direction in every local gradient descent step. While the functional relationship between consecutive local gradient updates is different from the relationship considered in Proposition 2 due to the updates of other devices not being present, the partial derivative of the approximation is the same since Proposition 2 assumes the updates from the other devices are known. For local gradients, the same approximation may be used with $m_k = m$. Algorithm (1) presents Block ModShift when $R \geq 1$. Block Modshift also requires the devices to share $\boldsymbol{u}_k(n)$ with Bob in each round. While this does not require a secret channel, it adds to the communication cost. Note that we assume that Eve knows $h_k^E(n)$ and Eve may even know $\boldsymbol{\gamma}_k(n)$. In Section 4, we show that despite knowing these quantities, Eve's performance is degraded.

3.2 Convergence

Under ideal conditions where $\sigma_E, \sigma_B \to 0$, Eve expects $||\boldsymbol{w}(n+1)^E - \boldsymbol{w}(n)^E|| \to 0$ when Bob converges. Eve can use this as a test to investigate if gradients have been tampered with. We show that our scheme passes this test despite the addition of shifts. A similar result can be shown for non-zero σ_E, σ_B. In order to show our convergence result, we make the following assumption:

Assumption 2. *(G,B)-BGD or bounded gradient dissimilarity [9]: there exist constants $G \geq 0$ and $B \geq 0$ such that*

$$\sum_{i=1}^{K} \left|\left| \frac{m_k}{m} \boldsymbol{g}_k(\boldsymbol{w}, \boldsymbol{D}_k) \right|\right|^2 \leq G^2 + B^2 \left|\left| \sum_{k=1}^{K} \frac{m_k}{m} \boldsymbol{g}_k(\boldsymbol{w}, \boldsymbol{D}_k) \right|\right|^2 \quad \forall \boldsymbol{w}. \tag{8}$$

Since $\boldsymbol{\delta}_k(n) = -\eta \boldsymbol{g}_k(\boldsymbol{w}(n), \boldsymbol{D}_k)$, the above assumption also holds for all $\boldsymbol{\delta}_k(n)$ with $G_{eff} = \eta G$.

Proposition 3. *Given $||\boldsymbol{w}(n+1) - \boldsymbol{w}(n)|| \leq \epsilon$, Eve's weights are bounded as*

$$||\boldsymbol{w}(n+1)^E - \boldsymbol{w}(n)^E|| \leq \epsilon(1 + \max_k ||\boldsymbol{u}_k(n)|| \max_k ||\gamma_k(n)|| \alpha(n)) \tag{9}$$

where $\alpha(n) = \frac{\sum_{k=1}^{K} \frac{m_k}{m} ||\boldsymbol{\delta}_k(n)||}{\left|\left|\sum_{k=1}^{K} \frac{m_k}{m} \boldsymbol{\delta}_k(n)\right|\right|}$ is bounded under a $(0, B)$-BGD assumption.

Proposition 3 provides an upper bound on $||\boldsymbol{w}(n+1)^E - \boldsymbol{w}(n)^E||$ if $||\boldsymbol{\gamma}_k(n)||$ and $||\boldsymbol{u}_k(n)||$ are bounded.

4 Simulation Results

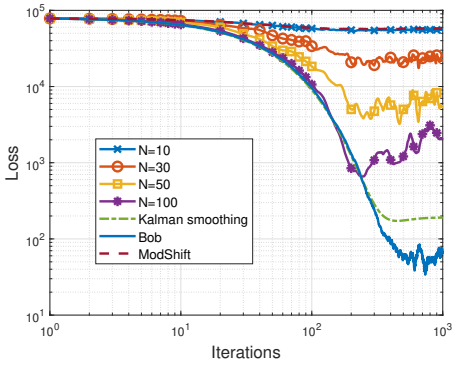
In this Section, we evaluate the performance of Block ModShift on a synthetic dataset and on the MNIST dataset. Unless otherwise mentioned, Eve simply uses $\underline{\boldsymbol{y}}_k^E(n)$ as defined in Equation (4) to update her weights. We also demonstrate the efficacy of Block ModShift in tackling the problem of Eve using temporal correlations for her estimation.

4.1 Eve's strategy

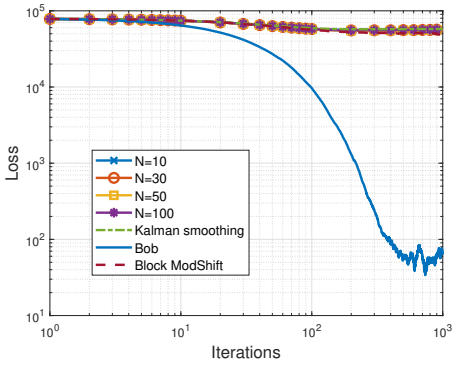
We consider the case where Eve is aware of the strategy being followed by the agents and constructs a learning setup which is designed to favor Eve's estimation ability. We consider a linear regression problem on a synthetic dataset with 1 agent and we set $R = 1$. Thus, $\boldsymbol{\delta}_k(n+1)$ is only a function of $\boldsymbol{w}(n)$ and $\boldsymbol{\delta}_k(n)$, removing the dependence on the updates of other devices. Setting $R = 1$ implies that Eve observes all the updates. The data vectors $\boldsymbol{x}_i$ are of dimension $d = 60$ and are Gaussian distributed with mean 0 and identity covariance matrix. The weight vector $\boldsymbol{w} = [1, 2, \cdots, d]^T$ is fixed and labels for each data vector are generated as $y_i = \boldsymbol{w}^T \boldsymbol{x}_i + n_i$ where n_i is a zero-mean

Algorithm 1 Block ModShift

Server executes:
Initialize $\boldsymbol{w}(0), \{\boldsymbol{u}_1(0), \ldots, \boldsymbol{u}_K(0)\}$
for each round $n = 0, 1, \ldots, N-1$ **do**
  for each client k **in parallel do**
    $\boldsymbol{w}_{k,0}(n) \leftarrow \boldsymbol{w}(n)$
    $\boldsymbol{u}_{k,0}(n) \leftarrow \boldsymbol{u}_k(n)$
    for each local epoch r from 0 to $R-1$ **do**
      $\boldsymbol{w}_{k,r+1}(n) = \boldsymbol{w}_{k,r}(n) - \eta g_k(\boldsymbol{w}_{k,r}(n), \boldsymbol{D}_k)$
      if $r < R-1$ **then**
        $\boldsymbol{u}_{k,r+1}(n) \leftarrow (\boldsymbol{I} - \eta \boldsymbol{H}_{k,r}(n))\boldsymbol{u}_{k,r}(n)$
      else
        $\boldsymbol{u}_{k,R}(n) \leftarrow (\boldsymbol{I} - \eta \frac{m_k}{m} \boldsymbol{H}_{k,R-1}(n))\boldsymbol{u}_{k,R-1}(n)$
      end if
    end for
    $\boldsymbol{\delta}_k(n) \leftarrow \boldsymbol{w}_{k,R}(n) - \boldsymbol{w}(n)$
    Transmit $[\boldsymbol{I} + (\boldsymbol{u}_{k,R-1}(n)\boldsymbol{\gamma}_k(n)^T)]\boldsymbol{\delta}_k(n)$ and $\boldsymbol{u}_{k,R-1}(n)$
    Share $\boldsymbol{\gamma}_k(n)^T \boldsymbol{\delta}_k(n)$ secretly with Bob
    $\boldsymbol{u}_k(n+1) \leftarrow \boldsymbol{u}_{k,R}(n)$
  end for
  $\boldsymbol{w}(n+1) \leftarrow \boldsymbol{w}(n) + \sum_{k=1}^{K} \frac{m_k}{m} \boldsymbol{\delta}_k(n)$
end for



(a) Comparison of Eve's performance with increasing block size under ModShift



(b) Comparison of Eve's performance with increasing block size under Block ModShift

Figure 2: Left: Algorithm for Block ModShift. Right: Performance comparisons under different configurations.

Gaussian random variable with standard deviation 0.1 for all i. The agent has a dataset of size $1,500$. We set $\frac{\sigma_B}{(h_k^B(n))} = \frac{\sigma_E}{(h_k^E(n))} = 0.1 \ \forall k, n$ and $\boldsymbol{w}^{Bob}(0) = \boldsymbol{w}^{Eve}(0) = \boldsymbol{0}$ where $\boldsymbol{0} \in \mathbb{R}^d$ is the zero vector. Since we consider 1 agent, we drop the subscript k.

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m]^T \in \mathbb{R}^{m \times d}$ be the data matrix and $\boldsymbol{y} = [y_1 \cdots y_m]^T$ be the label vector. We assume that Eve knows $\boldsymbol{X}^T\boldsymbol{X}$ and $\boldsymbol{S}(n)$. Note that the relation between $\boldsymbol{\delta}(n+1)$ and $\boldsymbol{\delta}(n)$ is linear. Thus, we have the following

$$\boldsymbol{\delta}(n+1) = \left(\boldsymbol{I} - \frac{2\eta}{m}\boldsymbol{X}^T\boldsymbol{X}\right)\boldsymbol{\delta}(n), \tag{10}$$

$$\underline{\boldsymbol{y}}^E(n) = \boldsymbol{S}(n)\boldsymbol{\delta}(n) + \boldsymbol{z}^E(n). \tag{11}$$

Eve can use Kalman smoothing [20], where $\boldsymbol{\delta}(n)$ is the state vector and $\boldsymbol{y}^E(n)$ is the output vector as defined in [20], on $\boldsymbol{\delta}(i), i \in \{1, \cdots T\}$ where T is the number of iterations, to estimate all the updates from the above equations. We also consider a windowed Kalman smoothing approach to investigate the effect of the block size N on the estimation. For the windowed Kalman smoother, Eve performs Kalman smoothing on $\{\boldsymbol{\delta}(i), \cdots, \min(\boldsymbol{\delta}(i+N-1), \boldsymbol{\delta}(T))\} \ \forall 1 \leq i \leq T$ and takes as output the smoothed estimate of $\boldsymbol{\delta}(i)$. This allows us to see how Eve's estimate varies as more future observations are used for estimation.

Figures 2 shows Eve's loss as a function of iterations when ModShift and Block ModShift respectively are used with varying estimation block sizes, with Kalman smoothing on all updates and if she does not do any estimation (ModShift and Block ModShift). We observe that $N = 10$ gives almost no

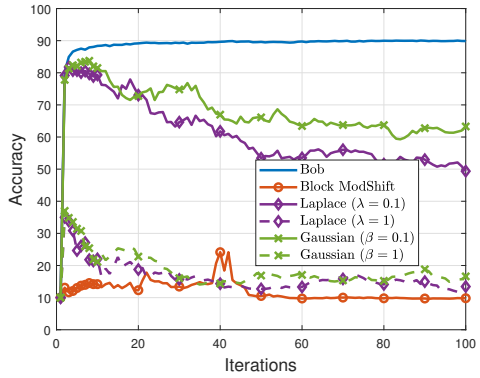


Figure 3: Comparison of accuracy with Block ModShift and noise addition scheme

additional benefit to Eve in both cases. While increasing N does improve Eve's performance when ModShift is used, the loss is still high. On the other hand, we observe that none of the schemes benefit from using future data when Block ModShift is used by agents. Thus, Block ModShift effectively addresses the shortcoming of ModShift which is to mitigate Eve's performance when she uses temporal correlations for estimation.

Note that despite a biased learning setup created to strengthen Eve's estimation capability, her performance is limited in ModShift and completely restricted in Block ModShift. In both cases, her loss is 3 orders of magnitude larger than Bob's loss.

4.2 Logistic Regression

We test Block ModShift on the MNIST dataset [13] for handwritten digit recognition. The training data consists of $60,000$ images which are 28×28 size grayscale images. They are equally divided between 10 devices with each device getting an equal number of images of each number. The pixel values are normalized to lie between 0 and 1. We use logistic regression for the multiclass classification task and use the cross entropy loss. The classes are $C = \{0, 1, 2, \cdots, 9\}$. Let $\boldsymbol{w} \in \mathbb{R}^{784\times 10}$ be the weight matrix and $\boldsymbol{w}_j$ be the column j of $\boldsymbol{w}$ which is the weight corresponding to class j. We set $\frac{\sigma_B}{(h_k^B(n))} = \frac{\sigma_E}{(h_k^E(n))} = 0.1 \ \forall k, n$ and $\boldsymbol{w}^{Bob}(0) = \boldsymbol{w}^{Eve}(0) = \mathbf{0}$ where $\mathbf{0} \in \mathbb{R}^{d\times d}$ is the zero matrix. The noise variance for the downlink transmission from Bob to agents is also set to 0.1. The learning rate is $\eta = 0.8$ and the number of local gradient rounds is $R = 10$. We set

$$\boldsymbol{\gamma}_k(n) = -\frac{\boldsymbol{u}_k(n)}{||\boldsymbol{u}_k(n)||^2} + \left(\boldsymbol{I} - \frac{\boldsymbol{u}_k(n)\boldsymbol{u}_k(n)^T}{||\boldsymbol{u}_k(n)||^2}\right)\boldsymbol{\theta}_k(n) \tag{12}$$

where each entry of $\boldsymbol{\theta}_k(n)$ is drawn independently from a uniform distribution over $[0, 100]$ and $\alpha_k(n)$ is chosen to ensure that $\boldsymbol{u}_k(n+1)$ has magnitude 1. We compare ModShift to a noise injection scheme where each agent adds a noise vector to the differences $\boldsymbol{\delta}_k(n)$. The noise injection scheme is inspired by differential privacy [1]. While we do not formally enforce differential privacy since we do not perform gradient clipping, the added noise serves to obscure individual updates. We consider Gaussian noise with covariance matrix $\beta^2\boldsymbol{I}$ and Laplace noise with scale matrix $\lambda\boldsymbol{I}$. The d elements of the noise vector are shared with Bob via the secret channel.

Figure 3 shows a plot comparing the accuracy of the different schemes on the MNIST training dataset. We observe that Block Modshift leads to the worst performance for Eve. In the presence of Block ModShift, Eve's classification performance is about 10% which is equivalent to random guessing. We also note that the cost of sharing the noise vectors secretly with Bob is large since $d = 7,840$ as opposed to Block ModShift which requires the sharing of a scalar. Furthermore, the *noise addition scheme does not pass the convergence test* proposed in Section 3.2, and thus Eve will be able to detect that Bob is engaging in a privacy preserving scheme.

5 Conclusions

We introduced a dynamic shift based approach to preserve model privacy against an eavesdropper Eve in a federated learning or distributed optimization environment when she uses the temporal correlation of updates to estimate updates. We derived the FIM for the shifted updates and theoretically identified a family of shift designs which drives the FIM to singularity. We also proposed a convergence test which Eve may use to identify tampered gradients and showed that our scheme passes this test. We compare Block ModShift with a noise addition scheme and observe that Block ModShift leads to a worse accuracy of 10% on the MNIST dataset which is equivalent to random guessing and demonstrated Eve's inability to use temporal correlations in her estimation.

Acknowledgments and Disclosure of Funding

This work has been funded in part by one or more of the following grants: ARO W911NF1910269, ARO W911NF2410094, NSF CIF-2311653, NSF CIF-2148313, NSF RINGS 2148313, ONR N00014-22-1-2363, NSF DBI-2412522, and is also supported in part by funds from federal agency and industry partners as specified in the Resilient & Intelligent NextG Systems (RINGS) program)

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[2] Zakaria Abou El Houda, Diala Nabousli, and Georges Kaddoum. Advancing security and efficiency in federated learning service aggregation for wireless networks. In *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1–6. IEEE, 2023.

[3] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, page 1175–1191, New York, NY, USA, 2017. Association for Computing Machinery.

[4] Maxime Ferreira Da Costa, Jianxiu Li, and Urbashi Mitra. Guaranteed private communication with secret block structure. *IEEE Transactions on Signal Processing*, 72:3547–3561, 2024.

[5] Yan Feng, Xue Yang, Weijun Fang, Shutao Xia, and Xiaohu Tang. Practical and bilateral privacy-preserving federated learning. *ArXiv*, abs/2002.09843, 2020.

[6] Andrea Goldsmith. *Wireless communications*. Cambridge university press, 2005.

[7] Swanand Kadhe, Nived Rajaraman, O Ozan Koyluoglu, and Kannan Ramchandran. Fastsecagg: Scalable secure aggregation for privacy-preserving federated learning. *arXiv preprint arXiv:2009.11248*, 2020.

[8] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

[9] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[10] Steven M. Kay. *Fundamentals of statistical signal processing: estimation theory*. Prentice-Hall, Inc., USA, 1993.

[11] Nomaan A Kherani and Urbashi Mitra. Modshift: Model privacy via designed shifts. *arXiv preprint arXiv:2507.20060*, 2025.

[12] Muah Kim, Onur Günlü, and Rafael F Schaefer. Federated learning with local differential privacy: Trade-offs between privacy, utility, and communication. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2650–2654. IEEE, 2021.

[13] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

[14] Jianxiu Li and Urbashi Mitra. Channel state information-free location-privacy enhancement: Delay-angle information spoofing. In *ICC 2024 - IEEE International Conference on Communications*, pages 3767–3772, 2024.

[15] Jianxiu Li and Urbashi Mitra. Channel state information-free location-privacy enhancement: Fake path injection. *IEEE Transactions on Signal Processing*, 2024.

[16] Jianxiu Li and Urbashi Mitra. Optimized parameter design for channel state information-free location spoofing. In *2024 IEEE International Symposium on Information Theory (ISIT)*, pages 3695–3700. IEEE, 2024.

[17] Dipankar Maity and Kushal Chakrabarti. On model protection in federated learning against eavesdropping attacks. *arXiv preprint arXiv:2504.02114*, 2025.

[18] Kalikinkar Mandal and Guang Gong. Privfl: Practical privacy-preserving federated regressions on high-dimensional data over mobile networks. page 57–68, 2019.

[19] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[20] Herbert E Rauch, F Tung, and Charlotte T Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA journal*, 3(8):1445–1450, 1965.

[21] Jinhyun So, Chaoyang He, Chien-Sheng Yang, Songze Li, Qian Yu, Ramy E Ali, Basak Guler, and Salman Avestimehr. Lightsecagg: a lightweight and versatile design for secure aggregation in federated learning. *Proceedings of Machine Learning and Systems*, 4:694–720, 2022.

[22] Naima Tasnim, Jafar Mohammadi, Anand D Sarwate, and Hafiz Imtiaz. Approximating functions with approximate privacy for applications in signal estimation and learning. *Entropy*, 25(5):825, 2023.

A Appendix

We first present a lemma which will be used to prove Proposition 1.

Lemma A.1. *Recall that for a general Gaussian model where $\underline{\boldsymbol{x}} \sim \mathcal{CN}(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{C}(\boldsymbol{\theta}))$, and we estimate the parameter $\boldsymbol{\theta}$ from $\underline{\boldsymbol{x}}$, the Fisher Information Matrix is given by [10]*

$$[\boldsymbol{J}(\boldsymbol{\theta})]_{ij} = 2Re\left\{\left[\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})^{\dagger}}{\partial \theta_i}\right]\boldsymbol{C}^{-1}(\boldsymbol{\theta})\left[\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_j}\right]\right\} + tr\left[\boldsymbol{C}^{-1}(\boldsymbol{\theta})\frac{\partial \boldsymbol{C}(\boldsymbol{\theta})}{\partial \theta_i}\boldsymbol{C}^{-1}(\boldsymbol{\theta})\frac{\partial \boldsymbol{C}(\boldsymbol{\theta})}{\partial \theta_j}\right]. \quad (13)$$

Lemma A.1 provides the expression for the FIM for parametric mean estimation in a Gaussian model.

A.1 Proof of Proposition 1

Proof. We substitute $\boldsymbol{\theta} = \boldsymbol{\delta}_k(n)$ and $\boldsymbol{\mu}(\boldsymbol{\delta}_k(n)) = \boldsymbol{S}_k(n+i)\boldsymbol{\delta}_k(n+i)$ (note that we will represent $\boldsymbol{\delta}_k(n+i)$ as a function of $\boldsymbol{\delta}_k(n)$ in the sequel) into Equation (13) and observe that $\boldsymbol{C}(\boldsymbol{\delta}_k(n)) = \sigma^2\boldsymbol{I}$ is not a function of $\boldsymbol{\delta}_k(n)$; thus, the trace term is 0. The FIM is given by

$$\boldsymbol{J}_{T_i}(\boldsymbol{\delta}_k(n)) = \frac{2h_k^E(n)^2}{\sigma_E^2}\left[\frac{\partial \boldsymbol{\delta}_k(n+i)}{\boldsymbol{\delta}_k(n)}\right]^T \boldsymbol{S}_k(n+i)^T\boldsymbol{S}_k(n+i)\left[\frac{\partial \boldsymbol{\delta}_k(n+i)}{\boldsymbol{\delta}_k(n)}\right]. \quad (14)$$

Simplifying $\left[\frac{\partial \boldsymbol{\delta}_k(n+i)}{\boldsymbol{\delta}_k(n)}\right]$,

$$\frac{\partial \boldsymbol{\delta}_k(n+i)}{\partial \boldsymbol{\delta}_k(n)} = \frac{\partial \boldsymbol{\delta}_k(n+i)}{\partial \boldsymbol{\delta}_k(n+i-1)} \frac{\partial \boldsymbol{\delta}_k(n+i-1)}{\partial \boldsymbol{\delta}_k(n+i-2)} \cdots \frac{\partial \boldsymbol{\delta}_k(n+1)}{\partial \boldsymbol{\delta}_k(n)}, \tag{15}$$

$$= \prod_{j=0}^{i-1} \frac{\partial \phi_{k,n+j}(\boldsymbol{\delta}_k(n+j))}{\partial \boldsymbol{\delta}_k(n+j)} \tag{16}$$

We note that if the condition 2 in Proposition 1 is satisfied with $\alpha_k(n) = 1 \forall n$,

$$\prod_{j=0}^{i-1} \frac{\partial \phi_{k,n+j}(\boldsymbol{\delta}_k(n+j))}{\partial \boldsymbol{\delta}_k(n+j)} \boldsymbol{u}_k(n) = \prod_{j=1}^{i-1} \frac{\partial \phi_{k,n+j}(\boldsymbol{\delta}_k(n+j))}{\partial \boldsymbol{\delta}_k(n+j)} \boldsymbol{u}_k(n+1) \tag{17}$$

$$\vdots$$

$$= \frac{\partial \phi_{k,n+i-1}(\boldsymbol{\delta}_k(n+i-1))}{\partial \boldsymbol{\delta}_k(n+i-1)} \boldsymbol{u}_k(n+i-1), \tag{18}$$

$$= \boldsymbol{u}_k(n+i). \tag{19}$$

Thus,

$$\boldsymbol{J}_{T_i}(\boldsymbol{\delta}_k(n)) \boldsymbol{u}_k(n) = \frac{2h_k^E(n)^2}{\sigma_E^2} \left[\frac{\partial \boldsymbol{\delta}_k(n+i)}{\boldsymbol{\delta}_k(n)}\right]^T \boldsymbol{S}_k(n+i)^T \boldsymbol{S}_k(n+i) \boldsymbol{u}_k(n+i). \tag{20}$$

If condition 1 in Proposition 1 is also satisfied, then $\boldsymbol{S}_k(n+i)\boldsymbol{u}_k(n+i) = 0$ which implies

$$\boldsymbol{J}_{T_i}(\boldsymbol{\delta}_k(n)) \boldsymbol{u}_k(n) = 0. \tag{21}$$

Thus, $\boldsymbol{u}_k(n)$ lies in the nullspace of $\boldsymbol{J}_{T_i}(\boldsymbol{\delta}_k(n))$ for $i \in \{0, \cdots, N-1\}$, which implies that the sum of these matrices is also singular. Note that $\alpha_k(n) = 1 \forall n$ is not necessary, the above can be shown for arbitrary non-zero $\alpha_k(n)$. □

A.2 Proof of Proposition 2

Proof. We note that

$$\boldsymbol{\delta}_k(n+1) = -\eta \boldsymbol{g}_k(\boldsymbol{w}(n+1), \boldsymbol{D}_k). \tag{22}$$

We now use the Taylor series expansion

$$\boldsymbol{\delta}_k(n+1) = -\eta \left[\boldsymbol{g}_k(\boldsymbol{w}(n), \boldsymbol{D}_k) + \nabla \boldsymbol{g}_k(\boldsymbol{w}(n), \boldsymbol{D}_k)(\boldsymbol{w}(n+1) - \boldsymbol{w}(n)) + \text{h.o.t.}\right] \tag{23}$$

$$\approx -\eta \left[\boldsymbol{g}_k(\boldsymbol{w}(n), \boldsymbol{D}_k) + \nabla \boldsymbol{g}_k(\boldsymbol{w}(n), \boldsymbol{D}_k) \sum_{j=1}^{K} \frac{m_j}{m} \boldsymbol{\delta}_j(n)\right], \tag{24}$$

$$= \boldsymbol{\delta}_k(n) - \eta \frac{m_k}{m} \boldsymbol{H}_k(n) \boldsymbol{\delta}_k(n) - \eta \boldsymbol{H}_k(n) \sum_{j=1, k \neq i}^{K} \frac{m_j}{m} \boldsymbol{\delta}_j(n). \tag{25}$$

Since $\boldsymbol{\delta}_i(n) \forall i \neq k$ are known vectors,

$$\frac{\partial \phi_{k,n}\left(\boldsymbol{\delta}_k(n)\right)}{\partial \boldsymbol{\delta}_k(n)} \approx \boldsymbol{I} - \eta \frac{m_k}{m} \boldsymbol{H}_k(n). \tag{26}$$

□

A.3 Proof of Proposition 3

Proof. Given $||\boldsymbol{w}(n+1) - \boldsymbol{w}(n)|| < \epsilon$,

$$\implies \left|\left|\sum_{k=1}^{K} \frac{m_k}{m} \boldsymbol{\delta}_k(n)\right|\right| < \epsilon. \tag{27}$$

$$||\boldsymbol{w}(n+1)^E - \boldsymbol{w}(n)^E|| = \left|\left|\sum_{k=1}^{K} \frac{(m_k[\boldsymbol{\delta}_k(n) + f_{k,n}(\boldsymbol{\delta}_k(n))\boldsymbol{u}_k(n)])}{m}\right|\right|, \tag{28}$$

$$\overset{(a)}{\leq} \epsilon + \left|\left|\sum_{k=1}^{K} \frac{m_k}{m}\gamma_k^T(n)\boldsymbol{\delta}_k(n)\boldsymbol{u}_k(n)\right|\right|, \tag{29}$$

$$\overset{(b)}{\leq} \epsilon + \sum_{k=1}^{K} \left|\left|\frac{m_k}{m}\gamma_k^T(n)\boldsymbol{\delta}_k(n)\boldsymbol{u}_k(n)\right|\right|, \tag{30}$$

$$\overset{(c)}{\leq} \epsilon + \max_k ||\boldsymbol{u}_k(n)|| \sum_{k=1}^{K} \left|\frac{m_k}{m}\gamma_k^T(n)\boldsymbol{\delta}_k(n)\right|, \tag{31}$$

$$\overset{(d)}{\leq} \epsilon + \max_k ||\boldsymbol{u}_k(n)|| \sum_{k=1}^{K} \left|\left|\frac{m_k}{m}\boldsymbol{\delta}_k(n)\right|\right| ||\gamma_k(n)||, \tag{32}$$

$$\overset{(e)}{\leq} \epsilon(1 + \max_k ||\boldsymbol{u}_k(n)|| \max_k ||\boldsymbol{\gamma}_k(n)||\alpha(n)), \tag{33}$$

$$\tag{34}$$

where (a) and (b) are due to the triangle inequality and Inequality (27) which is a bound on the norm of Bob's update, (c) from the simplifying of the norm of a scalar multiple of a vector and (d) from the Cauchy-Schwarz inequality; finally Inequality (27) is used.

□