Secure Multiparty Machine Learning with Zero-Knowledge Compliance Proofs

I. Technical Architecture

A. Cryptographic Primitives

Let participants $\mathcal{P}_1, \ldots, \mathcal{P}_n$ hold datasets $\mathcal{D}_1, \ldots, \mathcal{D}_n$. Our construction employs:

- **MPC Protocol:** A modified variant of the SPDZ protocol optimized for linear algebra operations. Each party computes local gradient updates $\nabla W_i^{(t)}$ and then secret shares them over a prime field $\mathbb{F}_p$:

  $$\nabla W = \sum_{i=1}^{n} \mathrm{SS}(\nabla W_i^{(t)}) \mod p,$$

  where $\mathrm{SS}(\cdot)$ denotes additive secret sharing.
- **ZKP System:** A Groth16-based proof system over the BN254 curve is used to generate proofs for compliance constraints. For each dataset $\mathcal{D}_i$, a circuit constraint $\phi(\mathbf{x})$ (e.g., for verifying that all records satisfy regulatory criteria) is defined:

  $$\begin{gathered} \phi(\mathbf{x}) := \Big\{ \forall x_j \in \mathcal{D}_i : \; f_{comp}(x_j) = 1 \Big\}, \\ \pi_i \leftarrow \text{Prove}(\text{CRS}, \phi, \mathcal{D}_i), \\ \text{Verify}(\text{CRS}, \pi_i) \in \{0, 1\}. \end{gathered}$$

  To reduce ZKP overhead for domain-specific constraints (e.g., medical compliance), our implementation can optionally incorporate PLONK-style custom gates.

B. Secure Aggregation Protocol

We describe a privacy-preserving federated averaging procedure:

Algorithm 1 Privacy-Preserving Federated Averaging

```
For each party $\mathcal{P}_i$: Compute local gradient update $\nabla W_i$ via stochastic gradient descent.
Secret-share the gradient: $\nabla W_i \leftarrow \mathrm{SS}(\nabla W_i)$.
Run the MPC protocol to compute the aggregated gradient:
    $$\nabla W_{agg} = \frac{1}{n} \sum_{i=1}^{n} \nabla W_i.$$
Each party generates a ZKP $\pi_i$ certifying that its gradient update is computed from data $\mathcal{D}_i$ satisfying $\phi(\mathbf{x})$.
Parties broadcast $(\nabla W_i, \pi_i)$ to a blockchain.
A smart contract verifies $\bigwedge_{i=1}^{n} \text{Verify}(\pi_i)$ before releasing the aggregated gradient.
```

II. Innovative Components

A. Hybrid MPC-ZKP Gradient Flow

Our core innovation is the joint execution of MPC and ZKP to ensure both privacy and regulatory compliance. We define a *trust score* derived from the aggregated proofs that adjusts the gradient aggregation as follows:

$$\underbrace{\mathbb{E}[\nabla W_{agg}]}_{\text{MPC}} = \underbrace{\mathbb{E}\Big[\frac{1}{n}\sum_{i=1}^{n} \nabla W_i\Big]}_{\text{FedAvg}} + \underbrace{\lambda \, \Sigma^{-1}(\pi_1, \ldots, \pi_n)}_{\text{ZKP Trust Score}},$$

where λ balances the contribution of proof verification to the overall gradient update. This design is similar in spirit to recent work (e.g. RiseFL) that uses lightweight ZKPs for robust aggregation in federated settings.

B. ZKP Circuit Design for Compliance

To enforce application-specific policies, we design custom Rank-1 Constraint System (R1CS) circuits. For example, for medical compliance we include:

a) Age Verification:: Ensure that each record satisfies a minimum age requirement:

$$\bigwedge_{k=0}^{7} \, (a_k \in \{0, 1\}), \quad \sum_{k=0}^{7} 2^k a_k \geq 18.$$

b) Differential Privacy Check:: Ensure that the added noise to gradients does not exceed a threshold:

$$\|\nabla W_{noisy} - \nabla W_{true}\|_2 \leq \beta.$$

III. Implementation Challenges

A. Performance Optimizations

- **MPC Communication:** Exploit gradient sparsity and apply compression techniques similar to CCSD to reduce communication overhead.
- **ZKP Efficiency:** Optimize proof generation using custom gate designs (e.g., PLONK-style) and reduce the number of constraints for regulatory proofs.

B. Security Analysis

Our framework's security relies on the following assumptions:

- The discrete logarithm problem (DLP) is hard in groups $\mathbb{G}_1$ and $\mathbb{G}_2$ (using BN254 pairing).
- In the MPC protocol, at most $t < n/2$ parties may be malicious.
- The underlying blockchain uses an honest-majority consensus mechanism.

We additionally address potential side-channel and collusion attacks by integrating established MPC and ZKP protocols with standard cut-and-choose methods.