Secure Outsourcing of Logistic Regression Classification using Homomorphic Encryption

(Project Type 4)

By

Team ENIGMA

Andy Chen, Avinash Narasimhan, Roman Negri

Applied Cryptography CS-GY 6903

Project Overview:

In this project, we aim to design and implement a secure solution that allows a cloud service provider to perform machine learning classification tasks on encrypted data supplied by a data owner. The key challenge is to ensure the confidentiality of the data owner's input data while still enabling the cloud service provider to compute the desired classification results.
To address this challenge, we will leverage the power of homomorphic encryption, which allows computations to be performed directly on encrypted data without decryption. This enables the cloud service provider to apply a machine learning classifier on the data owner's encrypted inputs and return the encrypted classification results, which the data owner can then decrypt to obtain the final output.

.

Chosen ML Classifier:

For this project, we have chosen to implement a Logistic Regression classifier as the machine learning model. Logistic Regression is a widely used and well-understood supervised learning algorithm suitable for binary classification tasks. It models the probability of a binary outcome as a function of one or more predictor variables. It is a relatively simple and interpretable model, which makes it easier to integrate with homomorphic encryption schemes. The mathematical operations involved in Logistic Regression, such as matrix multiplication and element-wise operations, can be effectively implemented using homomorphic encryption. Logistic Regression is a commonly used algorithm in many real-world applications, making it a relevant and practical choice for the project.

Homomorphic Encryption Scheme:

To enable the secure outsourcing of the Logistic Regression classifier, we will be using the TFHE (Torus Fully Homomorphic Encryption) scheme as the underlying homomorphic encryption mechanism.

TFHE is a fast and versatile fully homomorphic encryption scheme that supports boolean and arithmetic circuit computations. This makes it well-suited for implementing machine learning algorithms like Logistic Regression, which involves a mix of boolean and arithmetic operations.

TFHE is known for its computational efficiency, which is important for practical applications involving machine learning tasks. The ability to handle both boolean and arithmetic circuits allows TFHE to be used for a wide range of machine-learning models and algorithms.
TFHE is based on the learning with errors (LWE) problem, which provides strong cryptographic security guarantees. By leveraging the TFHE homomorphic encryption scheme, we can enable the cloud service provider to perform the Logistic Regression classification on the data owner's encrypted inputs, while preserving the confidentiality of the data.

Workflow:

The workflow of the solution will consist of the following steps:

1. **Setup**:

   A. Carol initializes the logistic regression model by setting up parameters such as weights and biases. These parameters are essential for the model to make predictions.

   B. Alice generates a pair of public and secret keys using the TFHE encryption scheme. The public key will be used for encrypting the data, while the secret key will be used for decryption.

2. **Encryption**:

   A. After preparing her test data, Alice encrypts it using the TFHE public key. This ensures that the data remains confidential during transmission to Carol.

   B. Once the test data is encrypted, Alice sends it securely to Carol, the cloud service provider, for evaluation.

3. **Evaluation**:

   A. Carol receives the encrypted test data from Alice.

   B. Carol utilizes the Concrete-ML library, which is designed to handle homomorphic computations, to evaluate the logistic regression model on the encrypted test data. This involves performing computations directly on the encrypted data without decrypting it.

   C. After completing the evaluation, Carol sends the encrypted classification results back to Alice for decryption.

4. **Decryption:**

   A. Alice receives the encrypted classification results from Carol.

   B. Using her secret key, Alice decrypts the received classification results, revealing the plaintext classification output. This step ensures that the confidentiality of the results is maintained until the data owner decrypts them.

Implementation

The implementation will involve the following steps:

A. **Logistic Regression Model Training** - During this phase, Carol will train the Logistic Regression model using a publicly available dataset. Carol will preprocess the dataset, handle missing values, and perform feature scaling and encoding as necessary. Once the data is prepared, Carol will train the model using techniques such as gradient descent or other optimization algorithms to minimize the loss function. After training, Carol will save the model parameters, including the weights and biases, which will be utilized during the evaluation phase to classify the encrypted test data.

B. **Homomorphic Encryption Setup**: In this step, Alice will generate the TFHE public and secret keys required for homomorphic encryption. Using TFHE, Alice will create a pair of keys, where the public key will be used for encrypting the test data samples, and the secret key will be kept confidential for decryption purposes. Generating these keys ensures that the encrypted data remains secure during transmission to Carol for evaluation.

C. **Data Encryption**: Once the TFHE keys are generated, Alice will proceed to encrypt her test data samples using the TFHE public key. Each test data sample will be encrypted individually to maintain confidentiality. The encryption process ensures that the sensitive information within the data remains protected while being transmitted to Carol, the cloud service provider, for evaluation.

D. **Homomorphic Evaluation**: During this phase, Carol will utilize the Concrete-ML library to evaluate the Logistic Regression classifier on the encrypted test data. The library is designed to handle homomorphic operations efficiently, allowing Carol to perform computations directly on the encrypted data without needing to decrypt it. This approach ensures that the confidentiality of the data is maintained throughout the evaluation process, enabling secure outsourcing of machine learning tasks to the cloud.

E. **Decryption**: Upon receiving the encrypted classification results from Carol, Alice will decrypt the results using her TFHE secret key to obtain the plaintext classification output. By decrypting the results locally using her secret key, Alice ensures that the confidentiality of the classification results is maintained until they are accessed by the data owner. This step completes the workflow, providing Alice with the final classification output while preserving data privacy.

Performance Analysis

We will analyze the performance of the proposed solution by measuring the following metrics:

A. **Encryption Time**: The time it takes for Alice to encrypt the test data samples.

B. **Evaluation Time**: The time it takes for Carol to evaluate the Logistic Regression classifier on the encrypted test data.

C. **Decryption Time**: The time it takes for Alice to decrypt the classification results.

D. **Accuracy**: The accuracy of the Logistic Regression classifier on the test data, compared to the accuracy obtained when using plaintext data.

Presentation

The project presentation will include the following:

A. Overview of the chosen ML classifier (Logistic Regression) and the homomorphic encryption scheme (TFHE).

B. Detailed explanation of the workflow and implementation steps.

C. Performance analysis results, including comparisons with the plaintext method.

D. Demonstration of the solution using sample test data and classification results.