CLAIMS

1. A computer-implemented method, for training multi-party secure logistic regression models, the computer-implemented method comprising:

receiving, by each of a plurality of secure computation nodes (SCNs), a plurality of random numbers from a random number provider;

encrypting, by each SCN, data stored at the respective SCN using the random numbers to generate a first encrypted data;

transmitting, by each SCN, the first encrypted data to another SCN of the plurality of SCNs;

receiving, by each SCN and from the another SCN, a second encrypted data that was encrypted using a second random number of the plurality of random numbers from the random number provider;

iteratively updating, by each SCN, a secure logistic regression model (SLRM) by using the encrypted data and the second encrypted data to generate an interim computation result that is divided into a plurality of shares that are transmitted to the plurality of SCNs to be used as inputs to the SLRM to determine a column vector of a prediction error as a difference between a sigmoid fitting function applied to a respective share and a label of a sample of a mini batch of a Stochastic Gradient Descent method, each of the plurality of shares hiding original data such that the original data cannot be recovered by using a single share of the plurality of shares; and

after iteratively updating the SLRM, outputting a result of the SLRM, wherein the result is configured to enable a service to be performed by each SCN of the plurality of SCNs, wherein the plurality of random numbers comprise a random matrix u, a random vector v, and a random number z, and are distributed between the each SCN and the another SCN, the first encrypted data and the second encrypted data comprise data encrypted by using the random matrix u, shares of the random vector v that are expressed as v0 and v1, and shares of the random number z that are expressed as z0 and z1, which are obtained at the each SCN and the another SCN, shares of the random number z are generated by a homomorphic calculation, and a sum of the shares of the corresponding random number z0 and z1 corresponds to a product between the random matrix u and the shares of the random vector v.

2. The computer-implemented method of claim 1, wherein each random number is at least one of a number, a vector, or a matrix.

3. The computer-implemented method of claim 1, wherein at least one of the random numbers is generated by a third-party agent.

4. The computer-implemented method of claim 1, wherein iteratively updating the SLRM is based on a Secret Sharing (SS) scheme.

5. The computer-implemented method of claim 1, wherein iteratively updating the SLRM is based on an event-driven model.

6. The computer-implemented method of claim 1, wherein iteratively updating the SLRM comprises determining a column vector of a prediction error based on equation:

$E = g(A) - Y_i$;

wherein g represents a fitting function of a sigmoid function, which can be a polynomial function fitting.

7. The computer-implemented method of claim 1, further comprising:

prior to receiving a plurality of random numbers from the random number provider, initializing parameters associated with the SLRM.

8. The computer-implemented method of claim 1, wherein iteratively updating the SLRM continues until a difference between two consecutive iteration results is less than a predetermined threshold.

9. A non-transitory computer-readable storage medium coupled to one or more computers and having instructions for training multi-party secure logistic regression models thereon executable by the one or more computers to:

receive, at a plurality of secure computation nodes (SCNs), a first random number of a plurality of random numbers from a random number provider;

encrypt, at each SCN, data stored at the SCN using the first random numbers;

transmitting, by each SCN, the first encrypted data to another SCN of the plurality of SCNs; receiving, by each SCN and from another SCN, a second encrypted data that was encrypted using a second random number of the plurality of random numbers from the random number provider;

iteratively update a secure logistic regression model (SLRM) by using the encrypted data and the second encrypted data to generate an interim computation result that is divided into a plurality of shares that are transmitted to the plurality of SCNs to be used as inputs to the SLRM to determine a column vector of a prediction error as a difference between a sigmoid fitting function applied to a respective share and a label of a sample of a mini batch of a Stochastic Gradient Descent method, each of the plurality of shares hiding original data such that the original data cannot be recovered by using a single share of the plurality of shares; and

after iteratively updating the SLRM, output a result of the SLRM, wherein the result is configured to enable a service to be performed by each SCN of the plurality of SCNs, wherein the plurality of random numbers comprise a random matrix u, a random vector v, and a random number z, and are distributed between the each SCN and the another SCN, the first encrypted data and the second encrypted data comprise data encrypted by using the random matrix u, shares of the random vector v that are expressed as v0 and v1, and shares of the random number z that are expressed as z0 and z1, which are obtained at the each SCN and the another SCN, shares of the random number z are generated by a homomorphic calculation, and a sum of the shares of the corresponding random number z0 and z1 corresponds to a product between the random matrix u and the shares of the random vector v.

10. The non-transitory computer-readable storage medium of claim 9, wherein each random number is at least one of a number, a vector, or a matrix.

11. The non-transitory computer-readable storage medium of claim 9, wherein at least one of the random numbers is generated by a third-party agent.

12. The non-transitory computer-readable storage medium of claim 9, wherein iteratively updating the SLRM is based on a Secret Sharing (SS) scheme.

13. The non-transitory computer-readable storage medium of claim 9, wherein iteratively updating the SLRM is based on an event-driven model.

14. The non-transitory computer-readable storage medium of claim 9, wherein iteratively updating the SLRM comprises determining a column vector of a prediction error based on equation:

$E = g(A) - Y_i$;

wherein g represents a fitting function of a sigmoid function, which can be a polynomial function fitting.

15. The non-transitory computer-readable storage medium of claim 9, the instructions thereon further executable by the one or more computers to:

prior to receiving the plurality of random numbers from the random number provider, initialize parameters associated with the SLRM.

16. The non-transitory computer-readable storage medium of claim 9, wherein iteratively updating the SLRM continues until a difference between two consecutive iteration results is less than a predetermined threshold.

17. A system, comprising:

one or more computers; and

one or more computer-readable memories coupled to the one or more computers and having instructions thereon executable by the one or more computers to:

receive, at a plurality of secure computation nodes (SCNs), a plurality of random numbers from a first random number of a random number provider;

encrypt, at each SCN, data stored at the SCN using the received random numbers first random number to generate a first encrypted data;

transmitting the first encrypted data to another SCN of the plurality of SCNs;

receiving, by each SCN and from another SCN, a second encrypted data that was encrypted using a second random number of the plurality of random numbers from the random number provider;

iteratively update a secure logistic regression model (SLRM) by using the encrypted data and the second encrypted data to generate an interim computation result that is divided into a plurality of shares that are transmitted to the plurality of SCNs to be used as inputs to the SLRM to determine a column vector of a prediction error as a difference between a sigmoid fitting function applied to a respective share and a label of a sample of a mini batch of a Stochastic Gradient Descent method, each of the plurality of shares hiding original data such that the original data cannot be recovered by using a single share of the plurality of shares; and

after iteratively updating the SLRM, output a result of the SLRM, wherein the result is configured to enable a service to be performed by each SCN of the plurality of SCNs, wherein the plurality of random numbers comprise a random matrix u, a random vector v, and a random number z, and are distributed between the each SCN and the another SCN, the first encrypted data and the second encrypted data comprise data encrypted by using the random matrix u, shares of the random vector v that are expressed as v0 and v1, and shares of the random number z that are expressed as z0 and z1, which are obtained at the each SCN and the another SCN, shares of the random number z are generated by a homomorphic calculation, and a sum of the shares of the corresponding random number z0 and z1 corresponds to a product between the random matrix u and the shares of the random vector v.

18. The system of claim 17, wherein each random number is at least one of a number, a vector, or a matrix.

19. The system of claim 17, wherein at least one of the random numbers is generated by a third-party agent.

20. The system of claim 17, wherein iteratively updating the SLRM is based on a Secret Sharing (SS) scheme.