Efficient Privacy-Preserving Machine Learning for Blockchain Network

Prof. B. P. N. Madhukumar, AssocProf CSE:BVCE, bpnmadhukumar@gmail.com
Prof. V. S. Ramakrishna, AssocProf CSE:BVCE, ramakrishnavasamsetty@gmail.com
Konki Jaya Lakshmi, CSE:BVCE
Kotteti Santosh Kumar, CSE:BVCE
Kumpatla Sai Chaitanya, CSE:BVCE
Bonam Gayathri, CSE:BVCE

Article Info

Page Number: 5227 - 5241

Publication Issue:

Vol 71 No. 4 (2022)

Abstract

A blockchain is a trustworthy and secure decentralised and distributed network that can be used in many places, like banking, finance, insurance, healthcare, and business. Recently, many communities in blockchain networks want to use machine learning models to get useful information from the large amounts of data that each participant owns but is spread out in different places. Distributed machine learning (DML) for blockchain networks has been studied as a way to run a learning model without putting all the data in one place. Even though there have been a number of ideas, privacy and security have not been dealt with well enough. As we will show later, the architecture has flaws and is not as efficient as it could be. In this paper, we propose a DML model for a permission block chain that protects privacy and solves privacy, security, and performance problems in a structured way. As core primitives, we come up with a stochastic gradient descent method with different levels of privacy and an error-based aggregation rule. Our model can handle any kind of differentially private learning algorithm that needs to define non-deterministic functions. The proposed error-based aggregation rule is a good way to stop attacks by a malicious node that tries to make DML models less accurate. In a differentially private scenario, the results of our experiments show that our proposed model is more resistant to attacks from the outside than other aggregation rules. Lastly, we show that our proposed model is very useful because it is easy to understand and takes little time to process.

Article History

***Article Received**: 25 March 2022*

***Revised**: 30 April 2022*

***Accepted**: 15 June 2022*

***Publication**: 19 August 2022*

Keywords: Data Modeling Language, Blockchain, Aggregation Rules, Error-Based Aggregation, Propose-Test-Release.

1. INTRODUCTION:

Recently, a lot of communities in blockchain networks want to use machine learning models to get results from statistical computing or data analysis. For example, a medical researcher who wants to provide patient-specific treatment can train a predictive model of disease by working with secure medical communities in a blockchain network without having to negotiate with each other for a database. But it can be hard to store large amounts of data from different parts of the world in one place because of usability, privacy concerns, security improvements, policies, and regulations like GDRP. A distributed machine learning (DML) model for block chain networks should be built with multiple entities, such as a computing node and multiple workers for parallel processing, so that a learning model can run without centralised data. This means that the computing node figures out the global weight by taking into account only the learning results from each worker in each round. Without a central data server, a DML can make it easy to use data that is spread across multiple domains.

1.1 Blockchain: A block chain is a trusted and secure decentralised and distributed network that allows interactions between participants, such as communities made up of people, companies, or governments with the same or similar goals. For example, crypto currencies, sharing medical information in healthcare, and exchanging goods in a business setting are all examples of interactions that take place on a block chain. Each participant in the blockchain shares a ledger (a list of transactions) with the others to make sure that every transaction is consistent and can't be changed. Each transaction is checked for validity by a majority of nodes coming to a consensus. If a transaction has been confirmed, a participant makes a group of transactions into a block and adds it to the last block in the shared ledger. In the ledger, these blocks are linked by a hash value, and the transactions can't be changed by changing the hash chain. By limiting who can join, permissionless blockchain networks are different from permissioned blockchain networks.

Permissionless blockchain: It aims to be a fully decentralised network that can grow in any situation or environment.

Permissioned blockchain, on the other hand, gives a way for a group of entities that share a goal but don't fully trust each other to communicate in a safe way. This could be a community for banking, finance, insurance, healthcare, and businesses that exchange information, money, or goods.

Distributed Machine Learning Model (DML):

It is a machine learning (ML) system with multiple nodes that improves performance, increases accuracy, and can handle larger amounts of data. It makes the machine less likely to make mistakes and helps people analyse and make decisions based on a lot of data. Distributed machine learning algorithms have grown to be able to deal with very large amounts of data.

When designing a DML model for a blockchain network, privacy and security should be taken into account in the right way. First of all, the DML model has to use a learning process to make sure that data doesn't leak private information. Most databases have private information about people, like diseases in medical records. As shown in Reference, even if a learning process only gives you summary information, you can still get some sensitive information from it. Differential privacy (DP), which is based on adding noise in a random way, is a promising way to protect privacy and stop privacy leaks.

On the system security side, computation in a distributed network with multiple parties can sometimes cause the system to fail because of a mistake in the computation. This is caused by workers who aren't reliable, either on their own or by working together to do something bad. The goal of this kind of collusion attack is to make the DML model less accurate by giving the wrong local gradient. These kinds of attacks have been thought about and planned for, and a rule to stop collusion attacks has been studied to stop attacks like these.

3.1 Algorithms: Differentially private algorithm:

It is a set of rules and systems that help keep people's information safe and private. In machine learning solutions, differential privacy may be required for regulatory compliance. In traditional situations, files and databases are used to store raw data.

3.1.1 Simulation Phase: During the simulation phase, each participant in the network simulates and calculates each local gradient under the current global weight on their own local datasets. Then, each learning round, the participants send the local gradients to the authority node, which makes a block. In this algorithm, there are 10 participants, so the dataset will be split among them. Each participant will train a model and store it in a Blockchain node. Any user with the right permissions can send a request to a blockchain node, which will then guess the value of that request. Blockchain is a list of computers, or nodes, that are all connected to each other. Each node will have a copy of the DML model, and for each request, each node will make a hashcode. This hashcode will be checked for all requests, and if the check fails, that node will be considered attacked and users can connect to other working nodes.

3.1.4 Hyperledger Fabric: It's just called Fabric, and it's one of the most popular permissioned blockchain systems. Before the ordering phase, Fabric adds an extra validation step called "simulation." This checks that all transactions in the current local state ledger are valid for certain peers, who are called "endorsing peers" in Fabric.

This shows how transactions work on Fabric, which is based on a three-phase architecture called "simulate, order, and execute."

First, a client runs the chaincode for a proposed transaction. The client, which is also called a client application, is a part of the Fabric network that runs chaincode. The chaincode is a set of functions for programmable transaction logics. It is often called a "smart contract."

These functions must be set up and run on a trusted distributed application before transactions can be done. Their functions are made up of a set of queries for doing things like loading, saving, executing, etc.

3.4 Algorithm and Process Design:

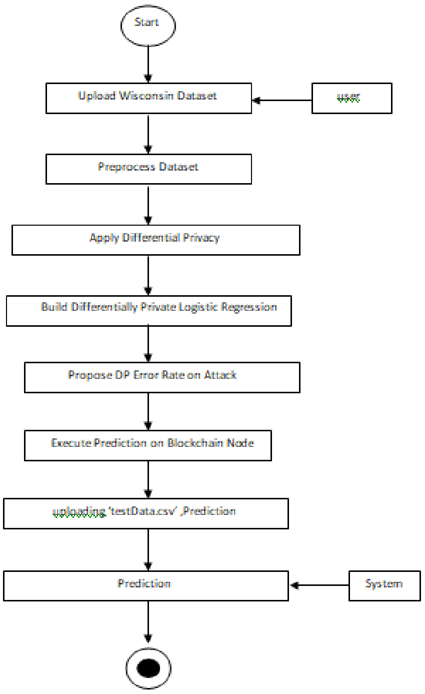

Start
Upload Wisconsin Dataset
user
Preprocess Dataset
Apply Differential Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
uploading 'testData.csv' ,Prediction
Prediction
System

Fig.4. **Process Design**

IV. Implementation and Outcome

We use Python to implement three aggregation rules: our error-based aggregation rule, the l nearest aggregation rule in LeaningChain, and multi-Krum on differentially private settings in a distributed way. We run Ubuntu 18.04 on an i7-8700K with 16 GB of RAM. On the hyperparameter side of machine learning, we set the learning rate to 0.1 and the clipping size C to 0.2 for the Wisconsin breast cancer dataset, which is computed empirically in a non-private scenario. Also, for our aggregation rule, we set = 2 so that the gradient scale doesn't get too big.

In this case, we think about two attack scenarios that involve a bad local gradient. First, we use it so that each of the f attackers proposes a local gradient drawn from a Gaussian distribution with a mean of 0 and an isotropic covariance matrix with a standard deviation of 200. This is known as a Gaussian attack.

Second, we use collusion attacks so that all f attackers suggest the same local gradient drawn close to the correct gradient Lf1, which is calculated by cosine similarity. This makes the DML model less accurate.

4.2 Criteria for judging:

The error rate for our aggregation rule is as follows:

Under a Gaussian mechanism, these are the test error results for the test dataset of the Wisconsin breast cancer dataset with different DP levels, =4 and =2, but the same =105. All experiments except the baseline are run with 30% bad attackers to show how each aggregation rule can stop bad attacks. Each figure in the top row shows a Gaussian attack resilience result. For each cosine similarity (distance) value, the other numbers in the rows below show how well it can defend against a Low distance attack. So, we can show that our error-based process is strong enough to withstand two attacks.

Effect of Budget Composition on Privacy:

The results of our experiment show how these ways of writing affect utility. As we talked about in Section II, we need to use a composition theorem to get more utility while keeping the same level of privacy. Above this, no other composition in a differentially private SGD has a lower bound than the moments accountant method.

But we wouldn't use moments accountant for the Wisconsin breast cancer dataset because Theorem 1) says that we can't set a sampling probability of q1/16 for 1 when D=450 and there are more than two participants. So, for the Wisconsin breast cancer dataset, a fairly large amount of noise is added. With a Gaussian mechanism and strong composition under 400 epochs, we add =200 for (4,105) DP and =316 for (2,105) DP.

In this case, our experiment shows that, like the baseline, the majority-based aggregation rules have a very high error rate. Experiments, on the other hand, show that our error-based method works well in the same settings, even if a lot of noise is added to ensure a high level of privacy. This means that our proposed model is useful even if the examples in a dataset are small.

The results of how much time our model takes up

In particular, our model has both DML accuracy and system efficiency in a number of different ways. We compare the time it takes to complete a transaction in our proposed model and in LearningChain. We don't use multi-Krum because, other than in the aggregation process, it doesn't give us any DML models that we can use. We test with PoW [22] for LearningChain and PoET [25], [44] for our model, and we keep track of the results for an average of 10 times.

How long it takes to figure out three aggregation rules:

Our error-based aggregation rule uses less time than multi-Krum and LearningChain for many different participants in a large-scale distributed network. Multi-Krum, in particular, can't be used in a distributed network because it takes a long time to calculate when there are a lot of participants. If we use PoET in our model for 20 participants, our model has pretty low transaction latency compared to LearningChain, which is based on a permissionless blockchain. Because of this, our model saves 16 times as much time on computations as the previous blockchain.

EPOCH:

An epoch is a term used in machine learning to describe how many times the machine learning algorithm has gone through the whole training dataset. Most of the time, data sets are grouped into batches (especially when the amount of data is very large)

When the size of the dataset is d, the number of epochs is e, the number of iterations is I and the size of each batch is b, the general relationship would be d*e = i*b.

Error Rate: The error of the method is the difference between what is expected and what is actually seen. If the target values are grouped into categories, the error is given as an error rate. This is how often the prediction turns out to be wrong.

4.3 Outcome:

The Summary of Validation will be shown based on the results of testing our proposed model, which does a better job of protecting privacy while machine learning for blockchain networks.

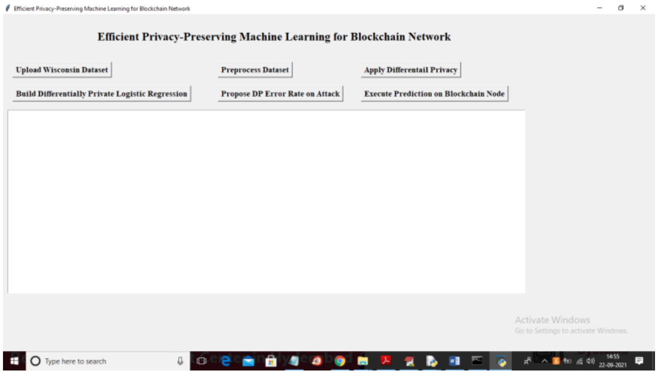
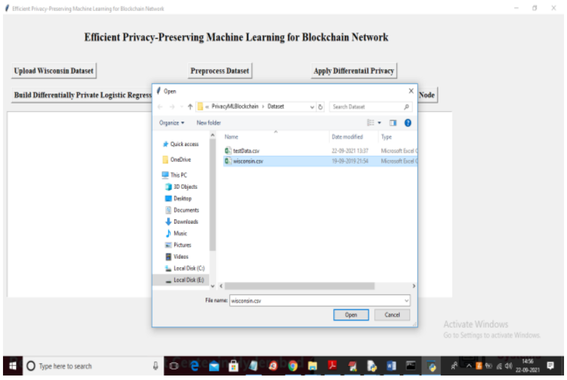

Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differentail Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
Activate Windows
Go to Settings to activate Windows.
Type here to search
14:55
22-09-2021

Fig.5. Upload Wisconsin Dataset’

In above diagram click on ‘Upload Wisconsin Dataset’ button to upload dataset and to get below diagram

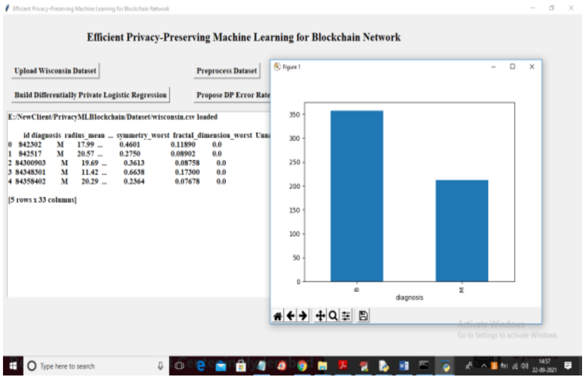


Name	Date modified	Type
testData.csv	22-09-2021 13:37	Microsoft Excel C
wisconsin.csv	19-09-2019 21:54	Microsoft Excel C

File name: wisconsin.csv

Fig.6uploading ‘wisconsin.csv’

In above diagram selecting and uploading ‘wisconsin.csv’ dataset and then click on ‘Open’ button to load dataset and to get below diagram


Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Build Differentially Private Logistic Regression
Propose DP Error Rate
Figure 1
350
300
250
200
150
100
50
0
B
M
diagnosis
Type here to search
14:57
22-09-2021

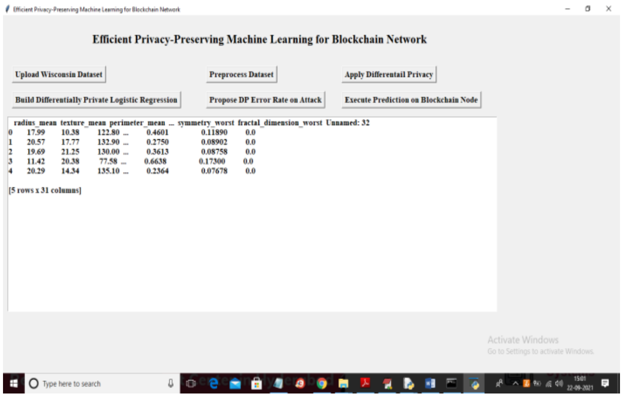
```
E:/NewClient/PrivacyMLBlockchain/Dataset/wisconsin.csv loaded

      id diagnosis  radius_mean  ...  symmetry_worst  fractal_dimension_worst  Unna
0   842302         M        17.99  ...          0.4601                  0.11890       0.0
1   842517         M        20.57  ...          0.2750                  0.08902       0.0
2 84300903         M        19.69  ...          0.3613                  0.08758       0.0
3 84348301         M        11.42  ...          0.6638                  0.17300       0.0
4 84358402         M        20.29  ...          0.2364                  0.07678       0.0

[5 rows x 33 columns]
```

Fig.7. dataset loaded

In above diagram dataset loaded and I am displaying few records from it and in dataset we can see some non-numeric values are there and DML will not accept such values so we need to process dataset by replacing non-numeric values with numeric ID where M will replace with 1 and B replace with 0. In above graph x-axis represents B and M classes and y-axis represents total number of records in that class. Now close above graph and then click on 'Preprocess Dataset' button to process dataset

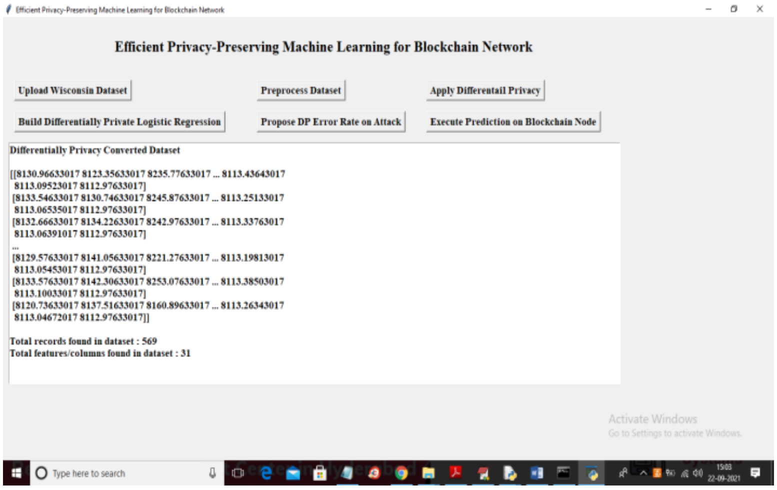

Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differentail Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
Type here to search
15:01
22-09-2021

```
  radius_mean  texture_mean  perimeter_mean  ...  symmetry_worst  fractal_dimension_worst  Unnamed: 32
0       17.99         10.38          122.80  ...          0.4601                  0.11890          0.0
1       20.57         17.77          132.90  ...          0.2750                  0.08902          0.0
2       19.69         21.25          130.00  ...          0.3613                  0.08758          0.0
3       11.42         20.38           77.58  ...          0.6638                  0.17300          0.0
4       20.29         14.34          135.10  ...          0.2364                  0.07678          0.0

[5 rows x 31 columns]
```

Fig.8. Apply Differential Privacy'

In above diagram all values are converted to numeric data and all data is in original plain format and now click on 'Apply Differential Privacy' button to anonymised dataset and to get below diagram. After anonymization you can see difference between above diagram values and below diagram values


Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differentail Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
Type here to search

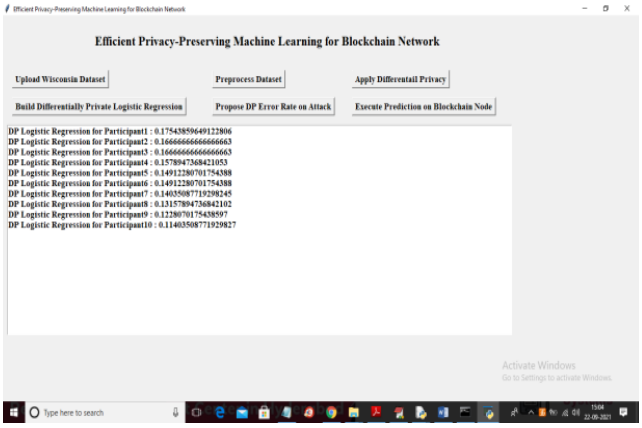
```
Differentially Privacy Converted Dataset

[[8130.96633017 8123.35633017 8235.77633017 ... 8113.43643017
  8113.09523017 8112.97633017]
 [8133.54633017 8130.74633017 8245.87633017 ... 8113.25133017
  8113.06535017 8112.97633017]
 [8132.66633017 8134.22633017 8242.97633017 ... 8113.33763017
  8113.06391017 8112.97633017]
 ...
 [8129.57633017 8141.05633017 8221.27633017 ... 8113.19813017
  8113.05453017 8112.97633017]
 [8133.57633017 8142.30633017 8253.07633017 ... 8113.38503017
  8113.10033017 8112.97633017]
 [8120.73633017 8137.51633017 8160.89633017 ... 8113.26343017
  8113.04672017 8112.97633017]]

Total records found in dataset : 569
Total features/columns found in dataset : 31
```

Fig.9. entire dataset values

In above diagram we can see entire dataset values are differentially anonymised and now click on ‘Build Differentially Private Logistic Regression’ button to build DML model with privacy


Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differentail Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
Type here to search

```
DP Logistic Regression for Participant1 : 0.17543859649122806
DP Logistic Regression for Participant2 : 0.16666666666666663
DP Logistic Regression for Participant3 : 0.16666666666666663
DP Logistic Regression for Participant4 : 0.1578947368421053
DP Logistic Regression for Participant5 : 0.14912280701754388
DP Logistic Regression for Participant6 : 0.14912280701754388
DP Logistic Regression for Participant7 : 0.14035087719298245
DP Logistic Regression for Participant8 : 0.13157894736842102
DP Logistic Regression for Participant9 : 0.12280701754385597
DP Logistic Regression for Participant10 : 0.11403508771929827
```

Fig.10. DP Logistic Regression

In above diagram DP Logistic Regression DML model generated for all 10 participants and for each participants we calculate error rate. Error rate represents percentage of number of records wrongly predicted from test records or attack records. Now click on ‘Propose DP Error Rate on Attack’ button to get error rate graph of all 10 participants.

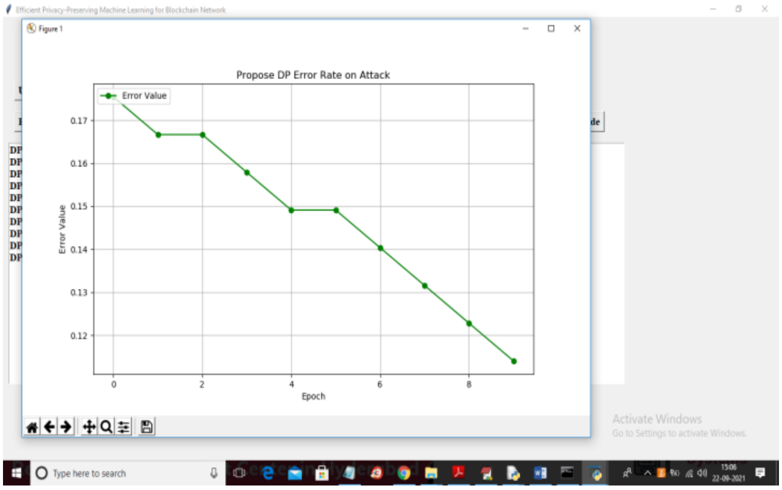
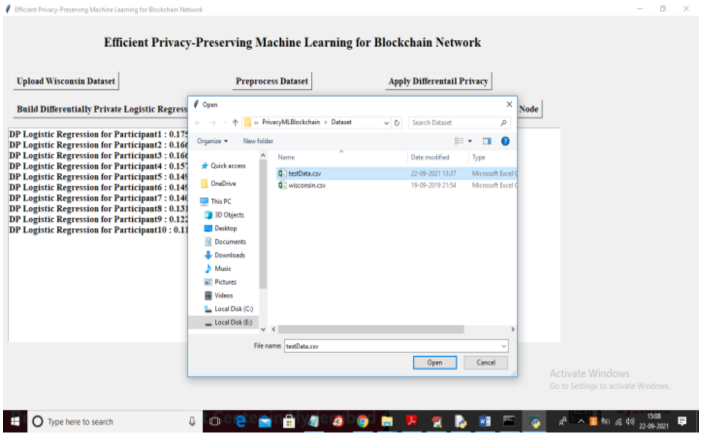


Fig.11. EPOCH vs error rate

In above graph x-axis represents EPOCH or number of trainings for DML and y-axis represents error rate and in above graph we can see with increasing participant's error rate reduced as DML is getting updated with all participants dataset. Now close above graph and then click on 'Execute Prediction on Blockchain Node' button to upload test data and then Blockchain DML will predict cancer from that test records



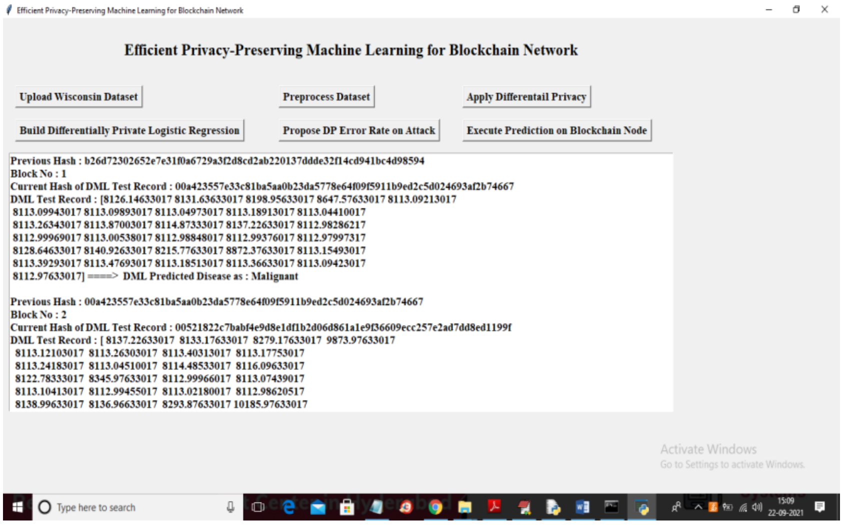
```
DP Logistic Regression for Participant1 : 0.17
DP Logistic Regression for Participant2 : 0.16
DP Logistic Regression for Participant3 : 0.16
DP Logistic Regression for Participant4 : 0.15
DP Logistic Regression for Participant5 : 0.14
DP Logistic Regression for Participant6 : 0.14
DP Logistic Regression for Participant7 : 0.14
DP Logistic Regression for Participant8 : 0.13
DP Logistic Regression for Participant9 : 0.12
DP Logistic Regression for Participant10 : 0.11
```

Name	Date modified	Type
testData.csv	22-09-2021 13:37	Microsoft Excel C
wisconsin.csv	19-09-2019 21:54	Microsoft Excel C

File name: testData.csv

Fig.12. testData.csv

In above diagramwe selected and uploaded 'testData.csv' file and then click on 'Open' button to get below output

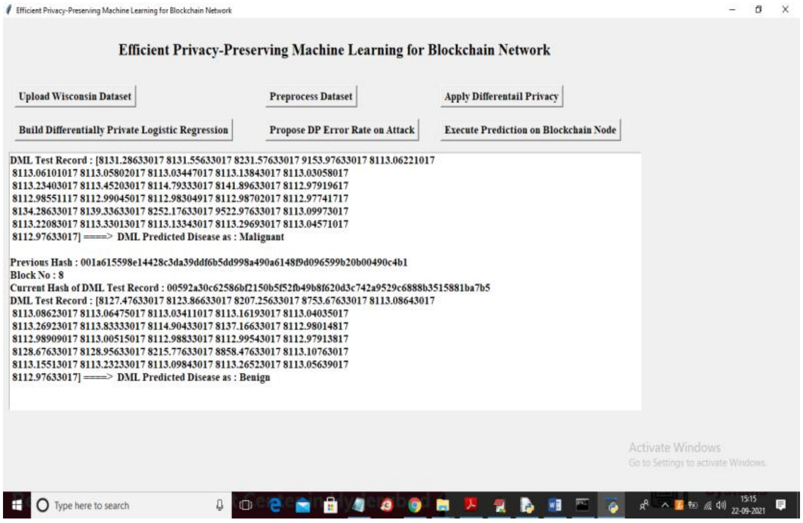

Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differentail Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
Activate Windows
Go to Settings to activate Windows.
Type here to search
15:09
22-09-2021

```
Previous Hash : b26d72302652e7e31f0a6729a3f2d8cd2ab220137ddde32f14cd941bc4d98594
Block No : 1
Current Hash of DML Test Record : 00a423557e33c81ba5aa0b23da5778e64f09f5911b9ed2c5d024693af2b74667
DML Test Record : [8126.14633017 8131.63633017 8198.95633017 8647.57633017 8113.09213017
 8113.09943017 8113.09893017 8113.04973017 8113.18913017 8113.04410017
 8113.26343017 8113.87003017 8114.87333017 8137.22633017 8112.98286217
 8112.99969017 8113.00538017 8112.98848017 8112.99376017 8112.97997317
 8128.64633017 8140.92633017 8215.77633017 8872.37633017 8113.15493017
 8113.39293017 8113.47693017 8113.18513017 8113.36633017 8113.09423017
 8112.97633017] ====> DML Predicted Disease as : Malignant

Previous Hash : 00a423557e33c81ba5aa0b23da5778e64f09f5911b9ed2c5d024693af2b74667
Block No : 2
Current Hash of DML Test Record : 00521822c7babf4e9d8e1df1b2d06d861a1e9f36609ecc257e2ad7dd8ed1199f
DML Test Record : [ 8137.22633017  8133.17633017  8279.17633017  9873.97633017
  8113.12103017  8113.26303017  8113.40313017  8113.17753017
  8113.24183017  8113.04510017  8114.48533017  8116.09633017
  8122.78333017  8345.97633017  8112.99966017  8113.07439017
  8113.10413017  8112.99455017  8113.02180017  8112.98620517
  8138.99633017  8136.96633017  8293.87633017 10185.97633017
```

Fig.13. Blockchain hashcode

In above diagram in first line we can see Blockchain hashcode of genesis block and then we can see Block_no where test data is store and in next line we can see hashcode of current test record and then we can see anonymised test record and the after arrow symbol we can see predicted record as Malignant or Benign. In above diagram Blockchain will verify previous hashcode of current record with last record. For example in above diagram Block_No2 previous hash is matching with Block_No1 current hash and verification will be successful and if change detected then attack is occurred. You scroll down above text area to view result of all records like below diagram


Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differentail Privacy
Build Differentially Private Logistic Regression
Propose DP Error Rate on Attack
Execute Prediction on Blockchain Node
Activate Windows
Go to Settings to activate Windows.
Type here to search
15:15
22-09-2021

```
DML Test Record : [8131.28633017 8131.55633017 8231.57633017 9153.97633017 8113.06221017
 8113.06101017 8113.05802017 8113.03447017 8113.13843017 8113.03058017
 8113.23403017 8113.45203017 8114.79333017 8141.89633017 8112.97919617
 8112.98551117 8112.99045017 8112.98304917 8112.98702017 8112.97741717
 8134.28633017 8139.33633017 8252.17633017 9522.97633017 8113.09973017
 8113.22083017 8113.33013017 8113.13343017 8113.29693017 8113.04571017
 8112.97633017] ====> DML Predicted Disease as : Malignant

Previous Hash : 001a615598e14428c3da39ddf6b5dd998a490a6148f9d096599b20b00490c4b1
Block No : 8
Current Hash of DML Test Record : 00592a30c62586bf2150b5f52fb49b8f620d3c742a9529c6888b3515881ba7b5
DML Test Record : [8127.47633017 8123.86633017 8207.25633017 8753.67633017 8113.08643017
 8113.08623017 8113.06475017 8113.03411017 8113.16193017 8113.04035017
 8113.26923017 8113.83333017 8114.90433017 8137.16633017 8112.98014817
 8112.98909017 8113.00515017 8112.98833017 8112.99543017 8112.97913817
 8128.67633017 8128.95633017 8215.77633017 8858.47633017 8113.10763017
 8113.15513017 8113.23233017 8113.09843017 8113.26523017 8113.05639017
 8112.97633017] ====> DML Predicted Disease as : Benign
```

Fig.14. result of all records

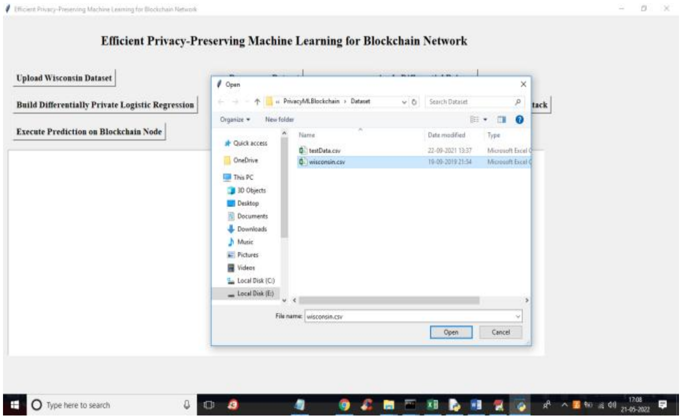
Extension Outcomes:

In this paper, we used a traditional machine learning algorithm called Logistic Regression to predict attacks from a privacy dataset and then calculated the prediction error rate. Prediction error is the percentage of wrong predictions, so the less error there is, the more accurate or right the prediction is.

So, as an extension, we're using the Random Forest algorithm with a group of trees. Random Forest is a popular algorithm for machine learning that uses the supervised learning method. It can be used for both Classification problems and Regression problems in ML. It is based on the idea of ensemble learning, which is a way to solve a hard problem and improve the model's performance by putting together multiple classifiers.

As the name suggests, "Random Forest is a classifier that uses a number of decision trees on different subsets of a given dataset and takes the average to improve the accuracy of that dataset's predictions." Instead of relying on just one decision tree, the random forest takes the prediction from each tree and predicts the final output based on what the majority of the trees say.

Due to the number of trees, the rate of wrong predictions will go down.

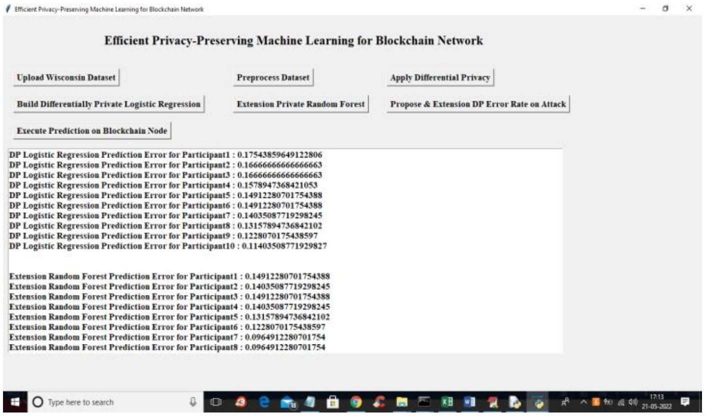

Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Build Differentially Private Logistic Regression
Execute Prediction on Blockchain Node
tack
Open
PrivacyMLBlockchain › Dataset
Search Dataset
Organize
New folder
Quick access
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
Open
Cancel
Type here to search
17:08
21-05-2022

Name	Date modified	Type
testData.csv	22-09-2021 13:37	Microsoft Excel C
wisconsin.csv	19-09-2019 21:54	Microsoft Excel C

File name: wisconsin.csv

Fig.15. selecting and uploading dataset file

In above diagram selecting and uploading dataset file and then click each button and then check error rate of propose logistic regression and random forest like below diagram


Efficient Privacy-Preserving Machine Learning for Blockchain Network
Efficient Privacy-Preserving Machine Learning for Blockchain Network
Upload Wisconsin Dataset
Preprocess Dataset
Apply Differential Privacy
Build Differentially Private Logistic Regression
Extension Private Random Forest
Propose & Extension DP Error Rate on Attack
Execute Prediction on Blockchain Node
Type here to search

```
DP Logistic Regression Prediction Error for Participant1 : 0.17543859649122806
DP Logistic Regression Prediction Error for Participant2 : 0.16666666666666663
DP Logistic Regression Prediction Error for Participant3 : 0.16666666666666663
DP Logistic Regression Prediction Error for Participant4 : 0.1578947368421053
DP Logistic Regression Prediction Error for Participant5 : 0.14912280701754388
DP Logistic Regression Prediction Error for Participant6 : 0.14912280701754388
DP Logistic Regression Prediction Error for Participant7 : 0.14035087719298245
DP Logistic Regression Prediction Error for Participant8 : 0.13157894736842102
DP Logistic Regression Prediction Error for Participant9 : 0.1228070175438597
DP Logistic Regression Prediction Error for Participant10 : 0.11403508771929827


Extension Random Forest Prediction Error for Participant1 : 0.14912280701754388
Extension Random Forest Prediction Error for Participant2 : 0.14035087719298245
Extension Random Forest Prediction Error for Participant3 : 0.14912280701754388
Extension Random Forest Prediction Error for Participant4 : 0.14035087719298245
Extension Random Forest Prediction Error for Participant5 : 0.13157894736842102
Extension Random Forest Prediction Error for Participant6 : 0.1228070175438597
Extension Random Forest Prediction Error for Participant7 : 0.0964912280701754
Extension Random Forest Prediction Error for Participant8 : 0.0964912280701754
```

Fig.16. logistic regression

In above diagram with propose logistic regression for participant 1 we got error rate as 0.17 and with extension we got error rate for same participant as 0.14 so by applying extension random forest we can further reduce error rate to enhance prediction and now click on 'Propose & Extension DP Error rate on Attack' button to get below graph

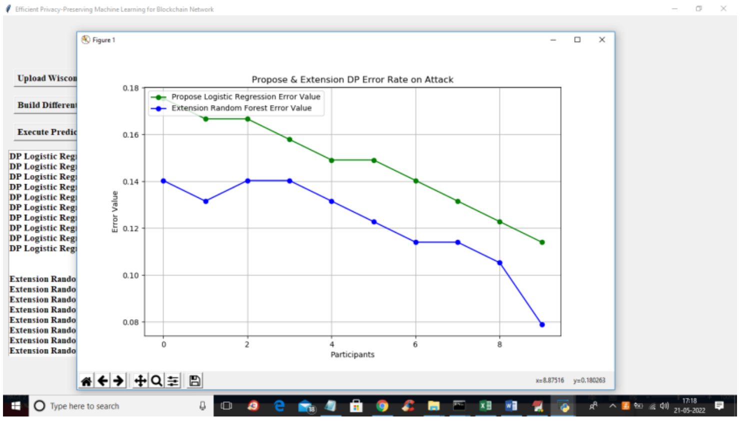


Fig.17. participants vs error rate

In above graph x-axis represents participants and y-axis represents error rate for that participant and blue line refers extension and green line refers propose logistic regression. From above two comparison we can see that extension got less error rate compare to propose

CONCLUSIONS

In this paper, we made a DML that protects privacy on a blockchain with permissions and DP. In particular, we suggested a new DML model based on the simulate-order-execute architecture of

Hyperledger Fabric, which is one of the most popular, practical, and usable permissioned blockchain systems. As the main primitive, we came up with an error-based aggregation rule, which is a new way to figure out global weights based on errors. In two different types of attacks on the aggregation process in the DML model, our experiments show that our error-based aggregation rule is more useful than majority-based aggregation rules. In a differentially private situation, our error-based aggregation rule is especially useful. Our model saves more time than permissionless blockchain-based DML systems and takes less time than other aggregation rules. The proposed model, which can be controlled by chain code functions, will be interesting to implement and divide into modules in the future. Also, in the future, work will be done to apply the modularized model to the latest version of the open-source Hyperledger Fabric, which can be used easily with private data by channels and the DML network.

Bibilography

[1] Linux Foundation. Hyperledger Announcements. Accessed: Oct. 8, 2018. [Online]. Available: https://www.hyperledger.org/announcements

[2] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, and S. Muralidharan, ‘‘Hyperledger Fabric: A distributed operating system for permissioned blockchains,’’ in Proc. 13th EuroSys Conf., 2018, p. 30.

[3] F. Benhamouda, S. Halevi, and T. Halevi, ‘‘Supporting private data on Hyperledger Fabric with secure multiparty computation,’’ IBM J. Res. Develop., vol. 63, pp. 3-1–3-8, Mar./May 2019. doi: 10.1147/JRD.2019.2913621.

[4] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, ‘‘Blockchain technology: Beyond bitcoin,’’ Appl. Innov. Rev., vol. 2, nos. 6–10, Jun. 2016. [Online]. Available: https://j2-capital.com/wpcontent/uploads/2017/11/AIR-2016-Blockchain.pdf

[5] N. Hynes, D. Dao, D. Yan, R. Cheng, and D. Song, ‘‘A demonstration of sterling: A privacy-preserving data marketplace,’’ in Proc. VLDB Endowment, vol. 11, no. 12, pp. 2086–2089, Aug. 2018.

[6] A Guide to GDPR Data Privacy Requirements. Accessed: Aug. 8, 2019. [Online]. Available: https://gdpr.eu/data-privacy/

[7] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, ‘‘Federated learning: Strategies for improving communication efficiency,’’ 2017, arXiv:1610.05492. [Online]. Available: https://arxiv.org/abs/1610.05492

[8] J. Hamm, Y. Cao, and M. Belkin, ‘‘Learning privately from multiparty data,’’ in Proc. Int. Conf. Mach. Learn., New York, NY, USA, 2016, pp. 555–563.

[9] A. Rajkumar and S. Agarwal, ‘‘A differentially private stochastic gradient descent algorithm for multiparty classification,’’ in Proc. Int. Conf. Artif. Intell. Statist., La Palma, Canary Islands, 2012, pp. 933–941.

[10] T.-T. Kuo and L. Ohno-Machado, ‘‘ModelChain: Decentralized privacypreserving healthcare predictive modeling framework on private blockchain networks,’’ 2018, arXiv:1802.01746. [Online]. Available: https://arxiv.org/abs/1802.01746

[11] A. Bellet, R. Guerraoui, M. Taziki, and M. Tommasi, ‘‘Personalized and private peer-to-peer machine learning,’’ in Proc. 21st Int. Conf. Artif. Intell. Statist., 2018, pp. 1–20.

[12] X. Chen, J. Ji, C. Luo, W. Liao, and P. Li, ‘‘When machine learning meets blockchain: A decentralized, privacy-preserving and secure design,’’ in Proc. Int. Conf. Bigdata, Seattle, WA, USA, 2018, pp. 1177–1186.

[13] M. Fredrikson, S. Jha, and T. Ristenpart, ‘‘Model inversion attacks that exploit confidence information and basic countermeasures,’’ in Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur., Denver, CO, USA, 2015, pp. 1322–1333.

[14] C. Dwork, F. McSherry, K. Nissim, and A. Smith, ‘‘Calibrating noise to sensitivity in private data analysis,’’ in Proc. Theory Cryptogr. Conf., New York, NY, USA, 2006, pp. 265–284.

[15] C. Dwork and A. Roth, ‘‘Basic techniques and composition theorems,’’ in The Algorithmic Foundations of Differential Privacy. Cambridge, MA, USA: Univ. Havard Press, 2014. doi: 10.1561/0400000042.

[16] P. Blanchard, R. Guerraoui, E. M. El Mhamdi, and J. Stainer, ‘‘Machine learning with adversaries: Byzantine tolerant gradient descent,’’ in Proc. Adv. Neural Inf. Proc. Syst., Long Beach, CA, USA, 2017, pp. 119–129.

[17] C. Xie, O. Koyejo, and I. Gupta, ‘‘Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance,’’ 2019, arXiv:1805.10032. [Online]. Available: https://arxiv.org/abs/1805.10032

[18] L. Chen, H. Wang, and D. Papailiopoulos, ‘‘DRACO: Robust distributed training against adversaries,’’ presented at the 2nd SysML Conf., Stanford, CA, USA, Feb. 2018.

[19] C. Xie, O. Koyejo, and I. Gupta, ‘‘Generalized byzantine-tolerant SGD,’’ 2018, arXiv:1802.10116. [Online]. Available: https://arxiv.org/ abs/1802.10116

[20] Hyperledger Fabric Release-1.4. Accessed: Aug. 6, 2018. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-1.4/