ZK-friendly ML model explorations - Milestone 3

Saeyoon Oh

March 2024

This article outlines the progress of milestone 3 for acceleration program by Ethereum PSE team. Article first restates the goals made for the report, concluding with the progress made and future works.

1 Goals

The goal of milestone 3 is to provide source code and documentations of how one can make classification using decision tree given heart failure dataset. We provide source code to prove decision tree prediction using EZKL, and mimic proof generation to obtain resource required using Circom, and compare the number of constraints with neural network.

- Functionality: Provide source code to train and prove decision tree output using EZKL.
- Functionality: Provide approach to prove trained balanced decision tree using Circom.
- Analysis: Compare proving approach using EZKL and Circom, analyze the constraints that are required to prove decision tree and compare them with neural network.

2 Progress

The progress made are as follows.

2.1 Generate script for automatic Circom key generation

While previous steps required users to manually generate Circom keys, we added a script that allows automatic key generation using snarkjs. Users can appropriately set the constant depending on the circuit size the user is trying to prove. You can find the file here.

2.2 Proving decision tree using EZKL

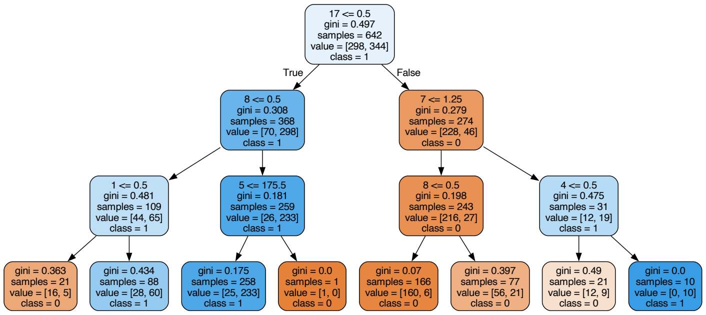
We first provide pipeline to train scikit learn decision tree and prove its output using EZKL. Since EZKL currently supports automatic proving scikit learn decision tree, this is straightforward. We provide end-to-end code for proving decision tree in the repository. The results are as follows:

- setting_time: 0.016
- calibration_time: 0.343
- compile_time: 0.002
- get_srs_time: 0.003
- witness_generation_time: 0.189
- setup_time: 4.975
- proof_generation_time: 49.212
- verification_time: 0.124
- total: 276
- correct: 230
- zk_total: 276.0
- zk_correct: 230.0
- Accuracy: 83.33
- ZK Accuracy: 83.33

We also note that for EZKL we need to use larger max_logrows (13) compared to max_logrows used for neural network verification. Although we set max depth of decision tree to 3, which forbids the tree to grow in complex manner, it yields accuracy of 83.33% which is significantly larger than scores we observed using multilayer-perceptrons. Yet, decision tree shows much longer proof generation time compared to multilayer-perceptrons. (The proving time for MLP were at most times under 1 second.) We conjecture that while neural networks can be seen as series of mathematical operations, decision tree requires a bit more program-like logic, which makes it a bit harder and longer to prove.

2.3 Proving trained decision tree using Circom

The first thing we attempted was to prove any kind of decision tree using Circom. We tried using previously implemented source codes such as ZK-DTP or ZKDT_release but was not able to resolve dependency issue nor make appropriate changes due to deprecated packages. Therefore, we used ZKDT repository to prove decision tree. Yet, current implementation only allows proving certain kind of decision tree. They need to be balanced, and each tree level needs to make decision using the same feature. On the other hand, if we look at the deciison tree in Figure 2.3 generated by scikit learn decision tree classifier, this does not use the same features in each level. Also, while the decision tree is balanced in this case, it is not necessarily always this case; they are a lot of the times not balanced. Therefore, we mimic a decision tree so that it can be proven using the ZKDT source code. This allows us to check the time for proving and number of constraints and compare the results with EZKL. ZKDT also contains deprecated Circom codes, so we provide another repository forked from ZKDT to enable proving. We also provide script to prove decision tree. Using the same sized decision tree, we calculate the time for proving. The results are as follows:



]

- proof generation time: 1.142
- verification time: 0.208

In terms of number of constraints, MLP model Circom circuit contained 8320 constraints, while decision tree contained 29602 constraints. In terms of number of constraints, MLP was superior. But we also note that the time gap was marginal (under 1 second). Detailed results for MLP Circom circuits is contained in progree report 1. The result can be reproduced by following the modified ZKDT repository.

2.4 Conclusion

For EZKL, decision tree provided higher accuracy while being more time inefficient. For Circom, decision tree shows higher number of constraints while being marginally slower. Therefore if neural networks were to show higher accuracy, they can be seen as a better option for building ZK-ML applications. While this may be alleviated by increasing the size of MLP, we leave that as future work. We also note that this result may not apply generally to all other tasks. We believe that the dataset we are using is a good fit for decision tree, where features do not have to be interacted implicitly. Therefore we believe different results can be made for other datasets that require machine learning models to learn the relationship between features.

3 Future Works

For milestone 4, we plan to do the followings.

- Cleanup and refactor source code for better user experience.
- Build proof generation pipeline for k-means clustering using EZKL and possibly by Leo using library built for k-means clustering using leo.
- Analyze the constraints and time complexity for generating proof for k-means clustering, comparing with decision tree and neural networks.