# Distributed and Secure ML using Self-tallying Multi-party Aggregation

Yunhui Long*, Tanmay Gangwani*, Haris Mughees and Carl Gunter
(* equal contribution)

*NeurIPS'18 workshop on Privacy Preserving Machine Learning (PPML)*

# Motivating Example -- Cumulative Voting


BLOCKCHAIN

Pizza: 10
Salad: 0

Pizza: 7
Salad: 3

Pizza: -100
Salad: 110

Goals:

- ❑ Calculate Homomorphic Vector Addition
- ❑ Protect privacy
- ❑ Generate and Validate Proofs for Input Validity

## Challenges

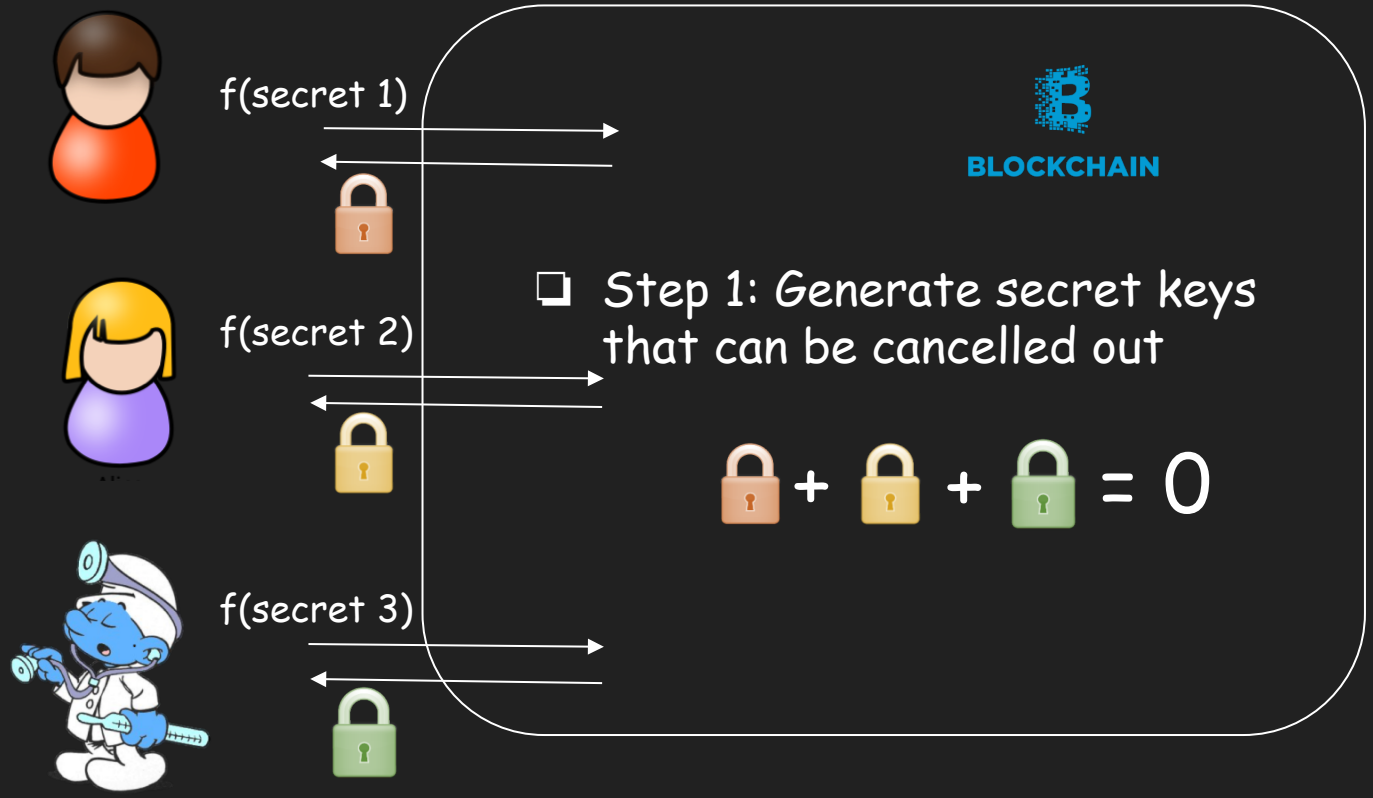
- No trusted third party
- No private channel
- Participating Parties can be malicious

## Our Approaches

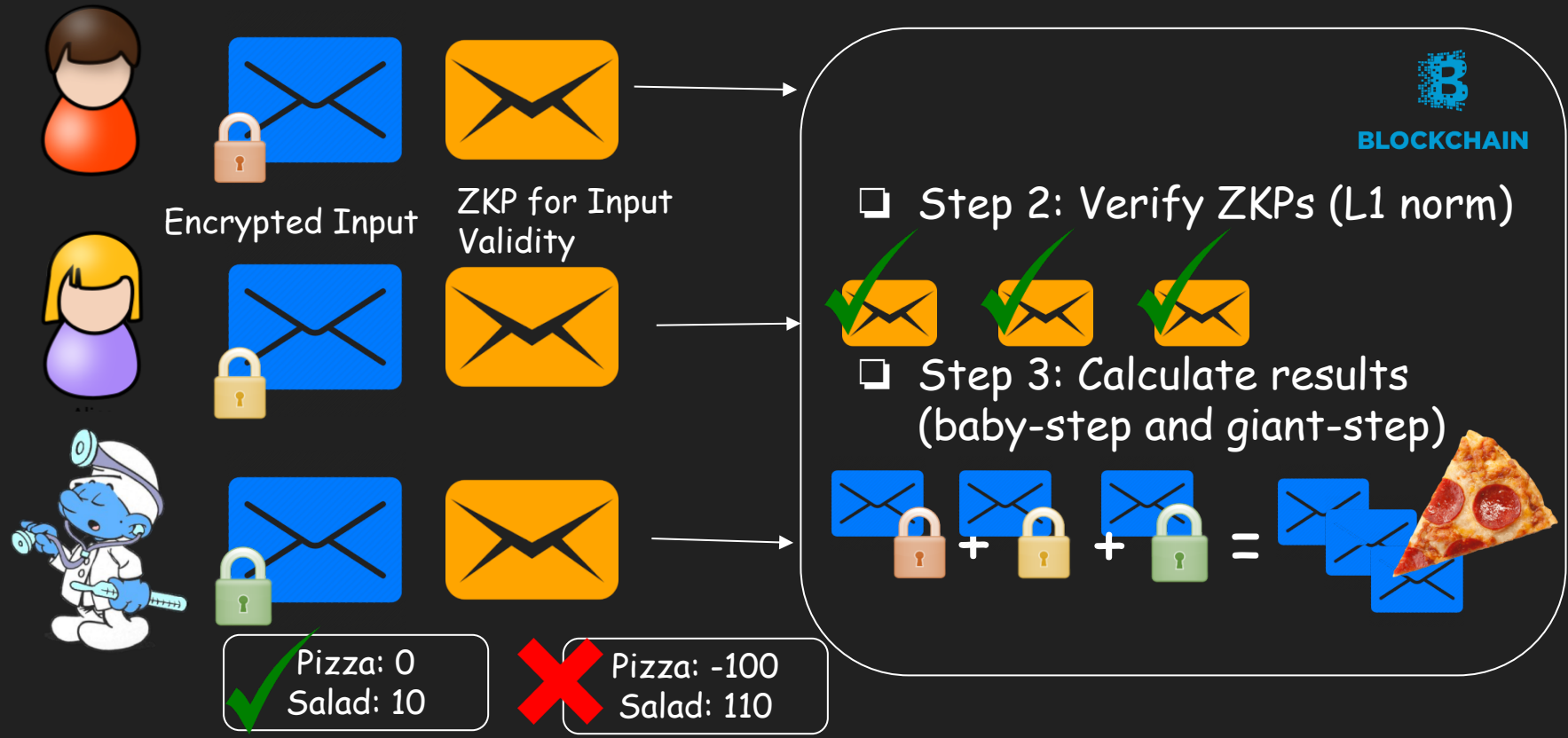
- Blockchain
- Additive Homomorphic Encryptions [Hao et al.]
- ZKP for input validity
  - L1 - norm range proof
  - L2 - norm range proof

[Hao et al.] Hao, Feng, Peter YA Ryan, and Piotr Zielinski. "Anonymous voting by two-round public discussion." IET Information Security 4.2 (2010): 62-67.

# Multi-party Vector Addition Protocol (Round 1)



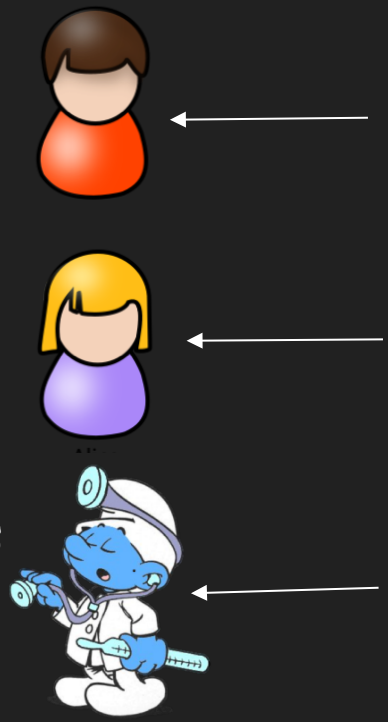
# Multi-party Vector Addition Protocol (Round 2)



# Self-tallying

Self-tallying:

- Verify ZKPs **offline**
- Compute the results **offline**
- **No** trusted talliers



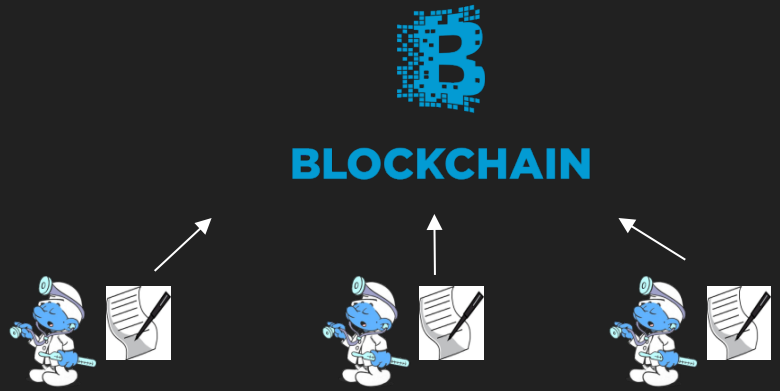
BLOCKCHAIN

- Store all the ZKPs

- Store all the encrypted inputs

# Beyond Cumulative Voting

- Machine Learning models
- Train on ALL data -> better accuracy
- Preserve confidential info.
- Get predictions (locally)




Naive Bayes( + + ) = Naive Bayes( + + )

# Beyond Cumulative Voting

- Machine Learning models
- Train on ALL data -> better accuracy
- Preserve confidential info.
- Get predictions (locally)

- Linear Regression
- Naive Bayes
- Decision Trees
- Matrix Ops. (SVD etc.)
- More...

Naive Bayes(  ) = Naive Bayes(  )

# One ZKP too many!

- ZKP L2 norm
  - Negative values ok
  - Composed of 4 ZKPs

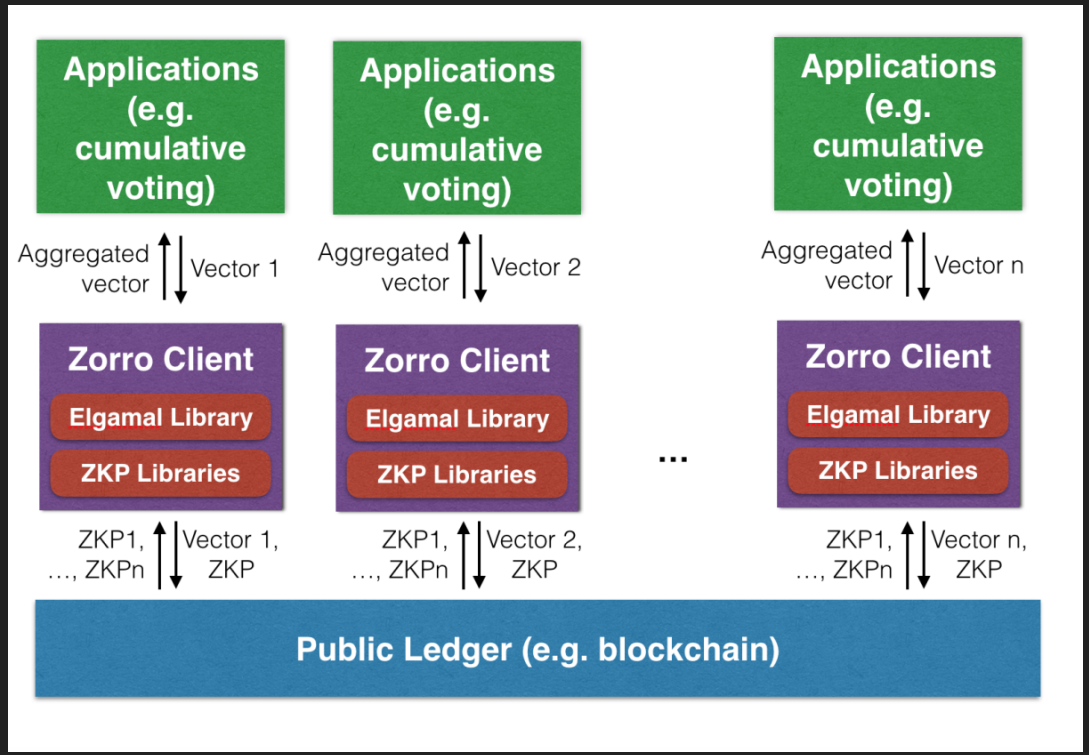
[-3 4 -5 7] + [-2 4 5 -9]

- ZKP L1 norm
  - Only positive values
  - Composed of many ZKPs

[3 4 5 7] + [2 4 5 9]

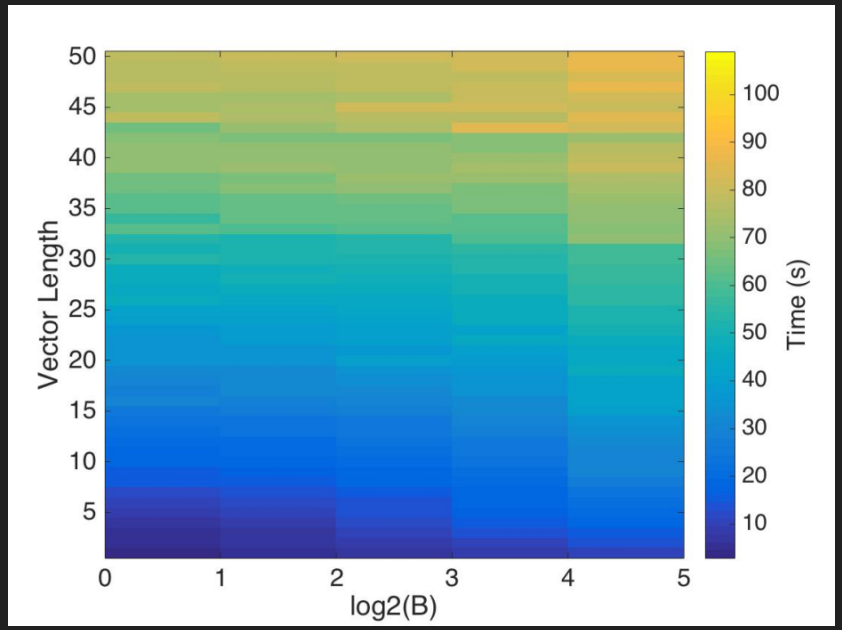
- Optimizing range proofs
  - Use base > 2
  - Gotta Batch'em All

# Implementation

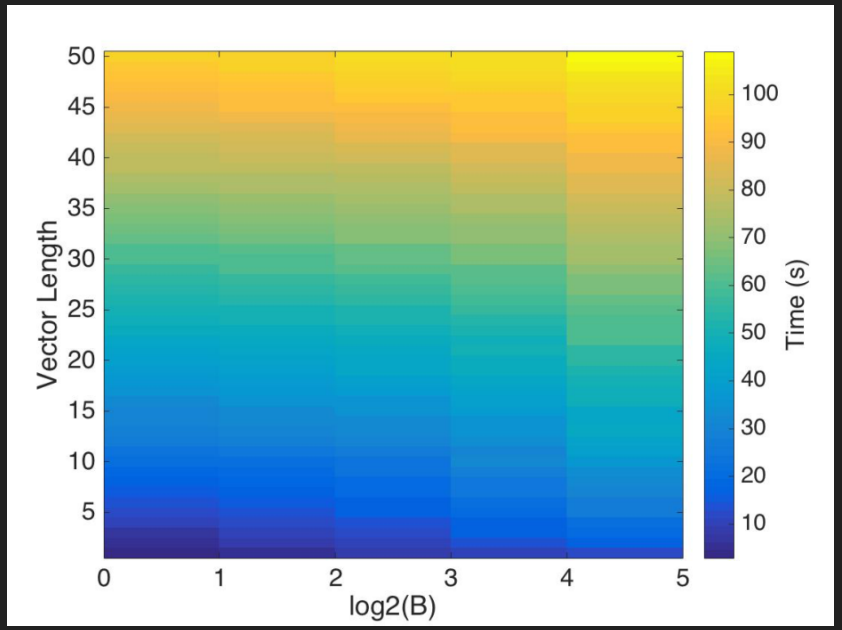



- ECC ElGamal
- Off-chain crypto
  - Generation and Verification
- Block-chain as white board
  - Proofs submission
  - Other public information

# ZKP Time Overheads

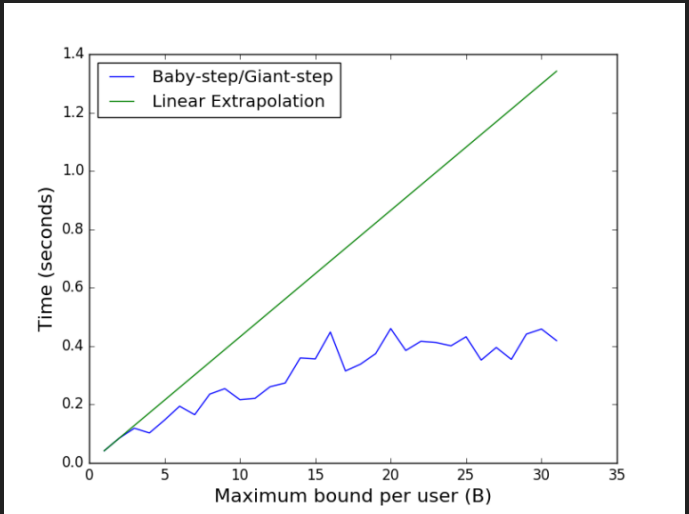



ZKP Generation Time per User

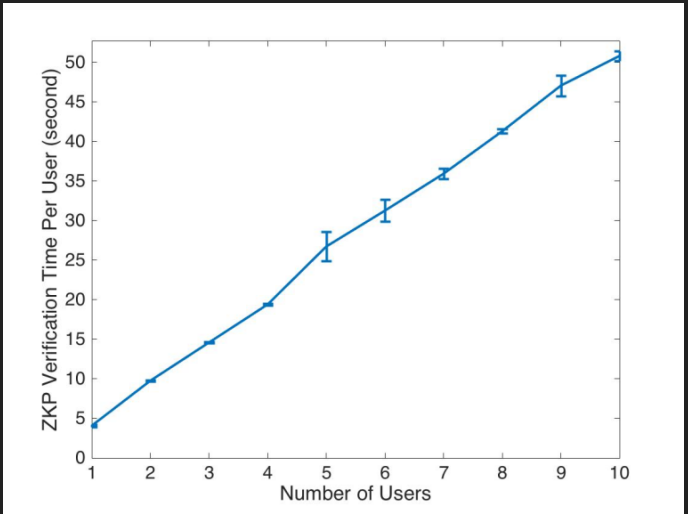



ZKP Verification time per User (n=1)

# Time Analysis




Time to Compute Discrete-log




Verification Time per User with Increasing Total Users

# Future Work

- Denial of Service Attacks
  - User fails to reveal the right ciphertext in the second round
  - Countermeasure 1: Identify the adversary, remove it from the protocol, and start a new round
  - Countermeasure 2: Punish the adversary by taking its collateral
  - More efficient countermeasures?
- Solve an open problem!
  - Discussion Forum Problem in cryptocurrency governance?
  - Multi-party Machine Learning?
    - Solved: Decision Tree, Naive Bayes, Matrix Factorization, Linear Regression
    - Challenges: SVM, Neural Network, LDA
- Evaluation of Alternative Methods (SGX, Generic Snarks)
- Combination with extra properties (e.g. Coercion resistance)
- Economic Feasibility

# Thank You!