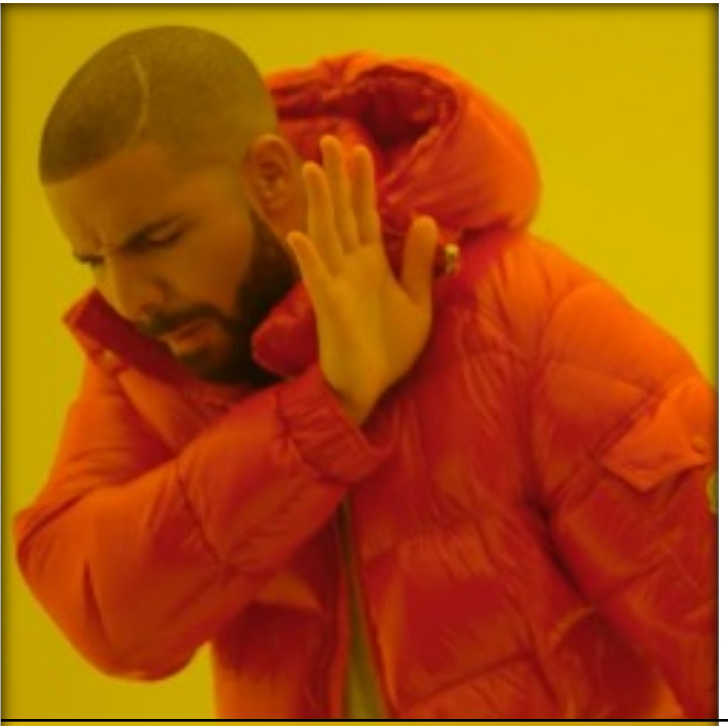
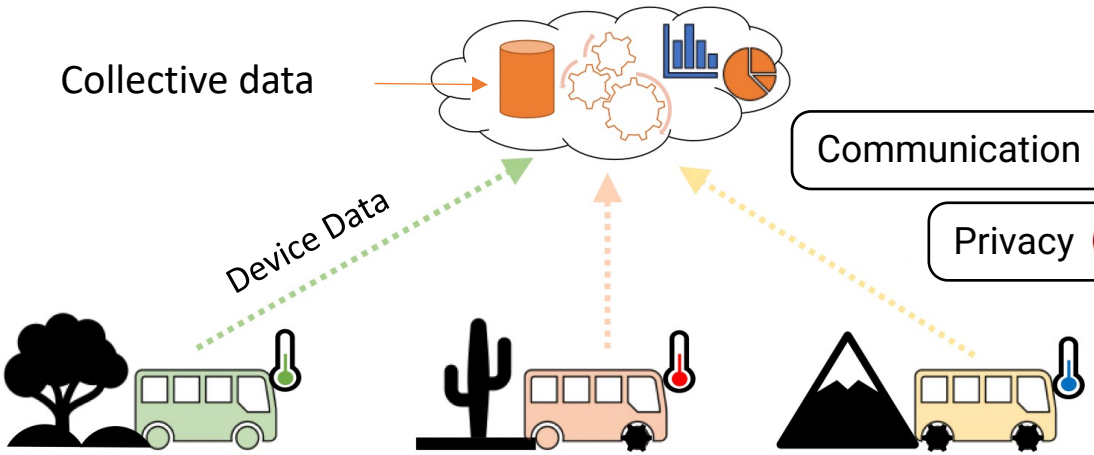
Federated Learning On a Cluster

Akshay Kamath
Parima Mehta

Group 17
ECE 6115



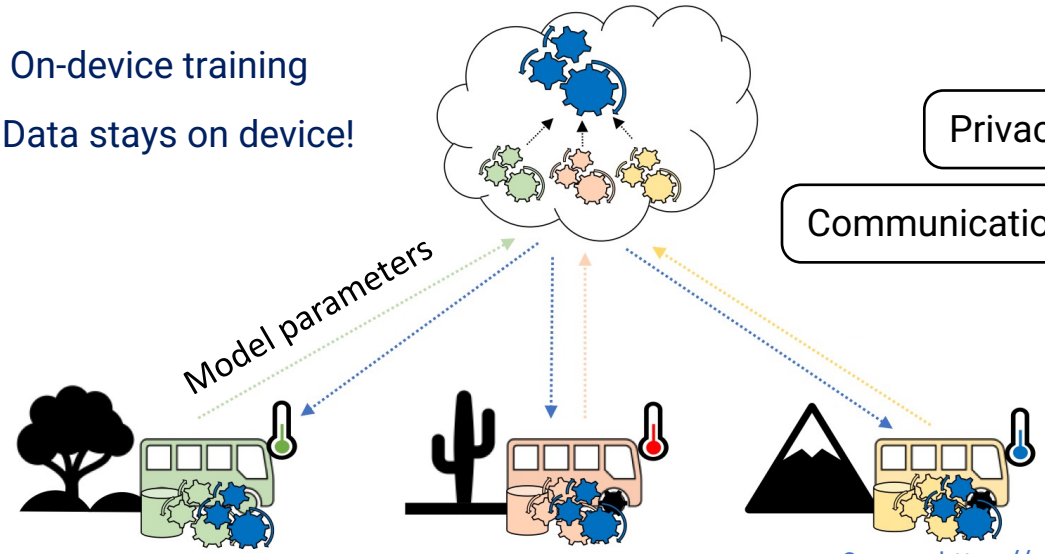
Centralized Learning



Federated Learning

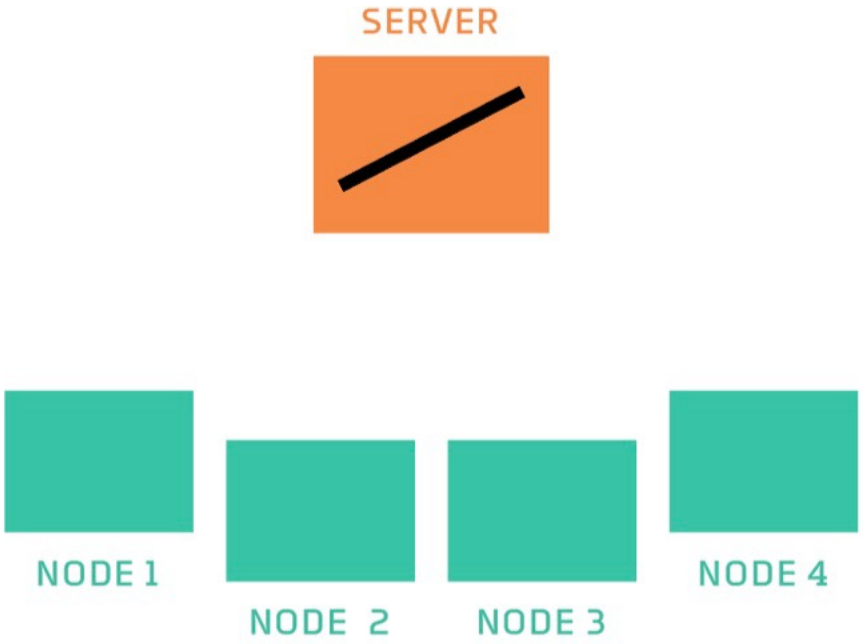
On-device training

Data stays on device!



Source: https://ekkono.ai

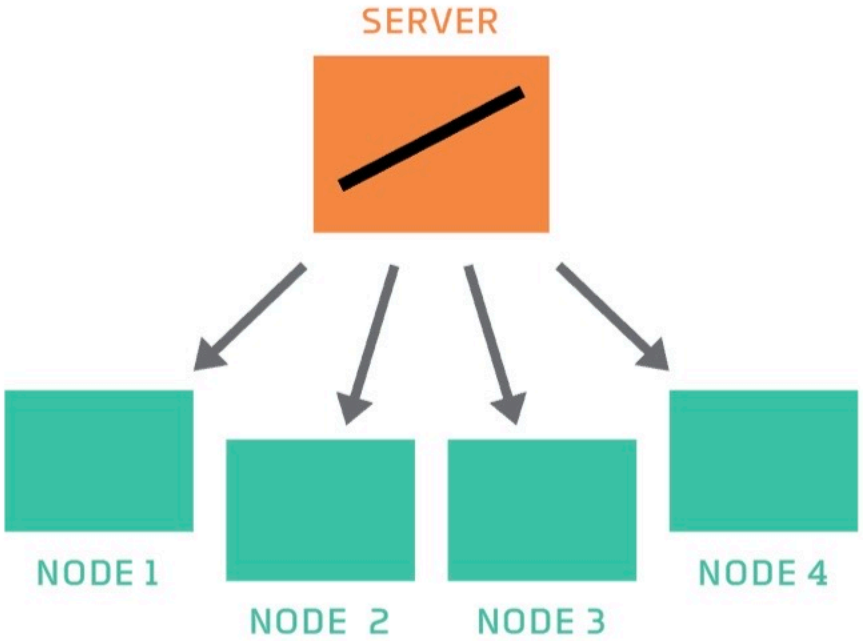
Federated Learning



Untrained model on the server
with randomly initialized parameters

Image source: https://mike.place/talks/fl/#21

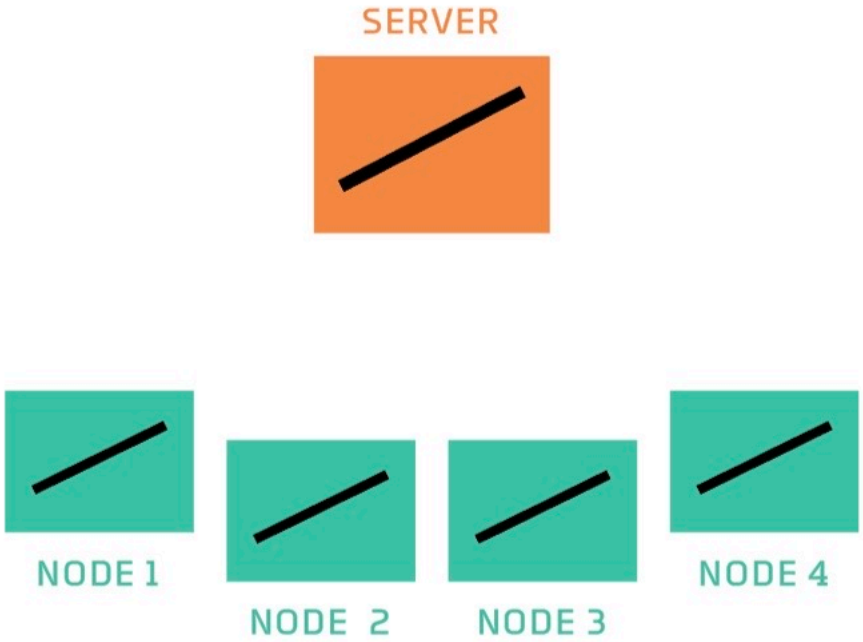
Federated Learning



Send a copy of the model
to all nodes

Image source: https://mike.place/talks/fl/#21

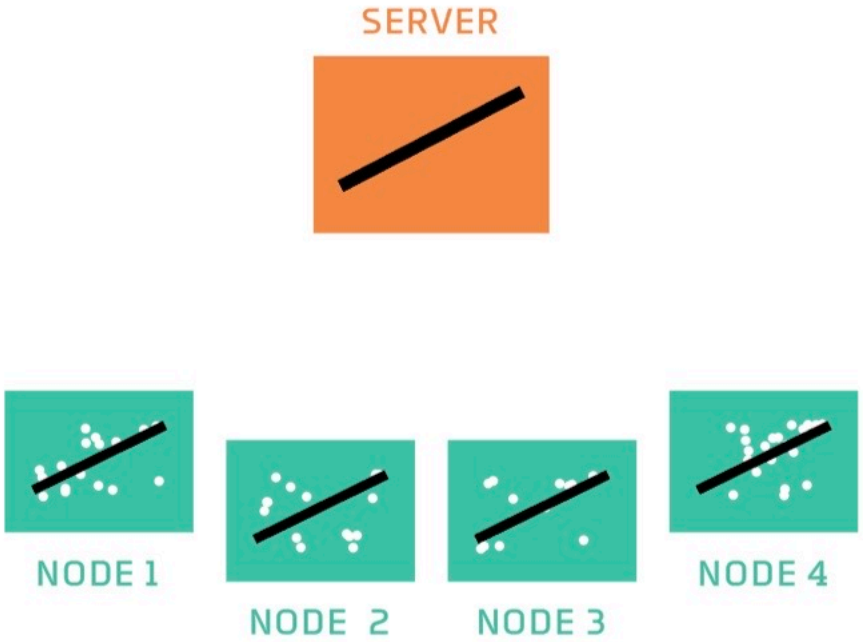
Federated Learning



Server instructs nodes to train their model instance

Image source: https://mike.place/talks/fl/#21

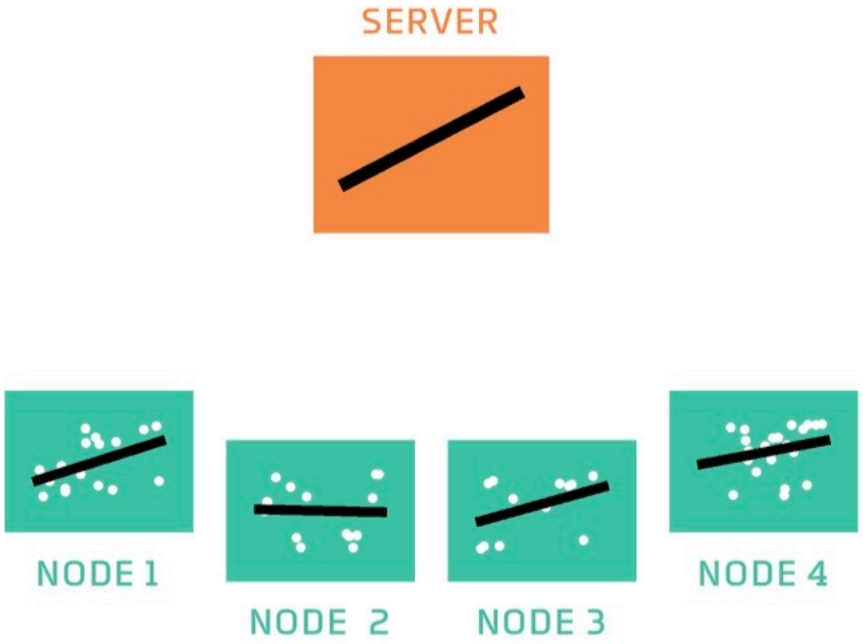
Federated Learning



Nodes acquire or load
some training data

Image source: https://mike.place/talks/fl/#21

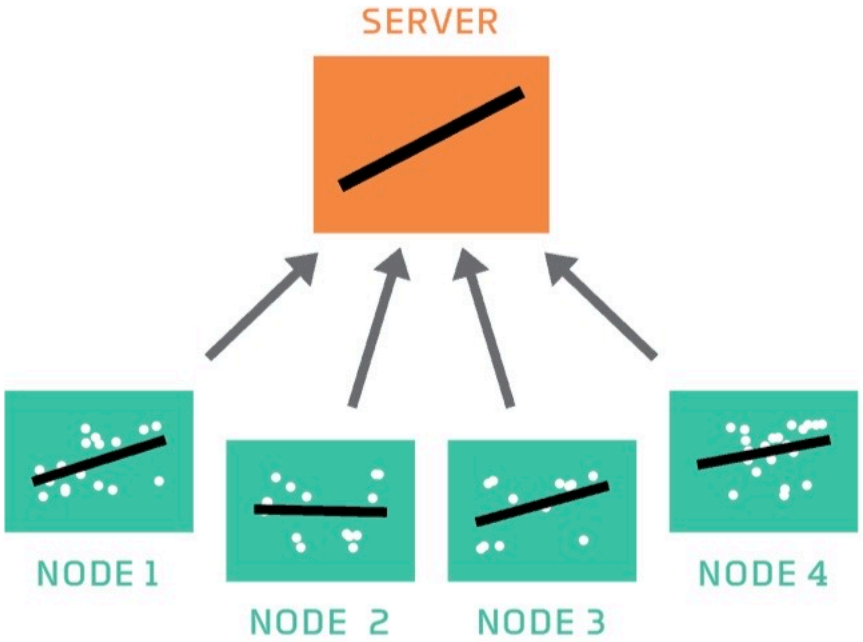
Federated Learning



Nodes train their model incrementally

Image source: https://mike.place/talks/fl/#21

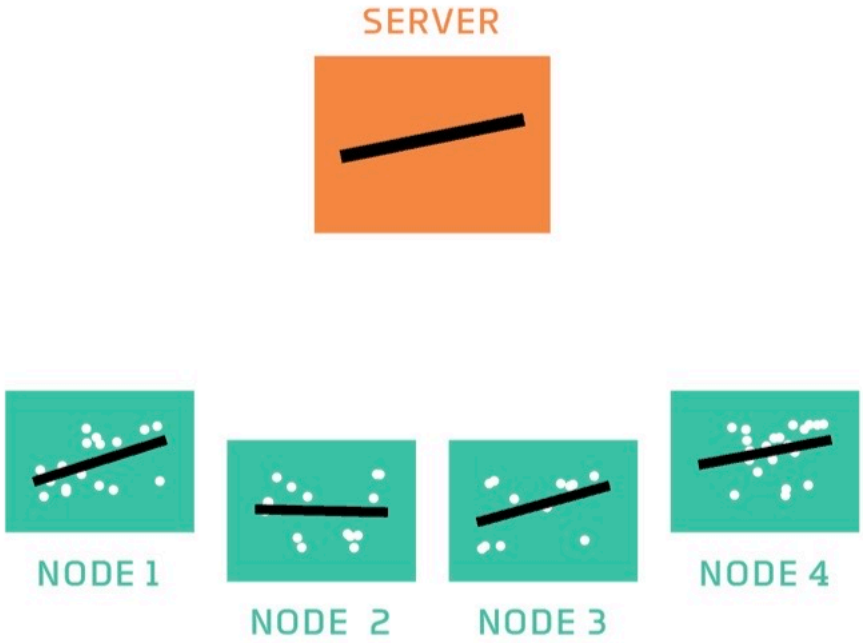
Federated Learning



Each node sends a copy of
their model back to the server

Image source: https://mike.place/talks/fl/#21

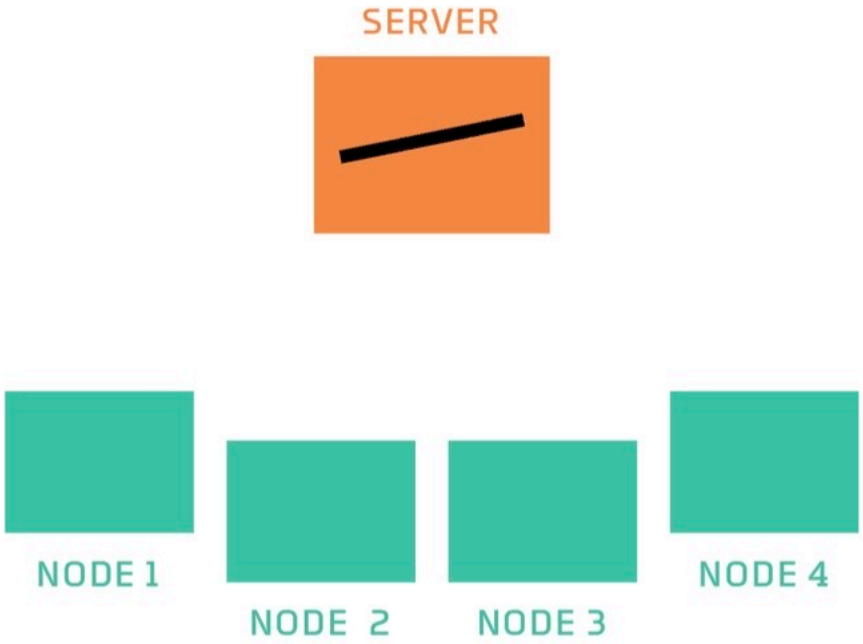
Federated Averaging!



Server averages the models
received from all nodes

Image source: https://mike.place/talks/fl/#21

Federated Round Complete

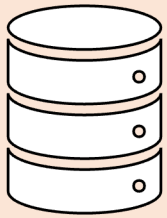


Repeat as required.

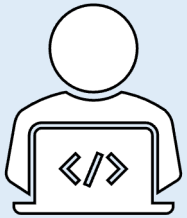
Image source: https://mike.place/talks/fl/#21

FLOC Phases

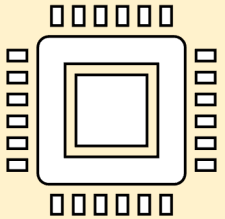
1. **Setting up cluster of Pynq-Z2 Boards**
   - Demonstration of distributed computation



2. **Simulating Federated Learning (FL) model**
   - Federated MNIST digit recognition with MLP
   - Analysis of algorithm/communication cost



3. **Deploying the FL model on the cluster**
   - Analysis of hardware/computation cost



Pynq-Z2 Cluster Setup

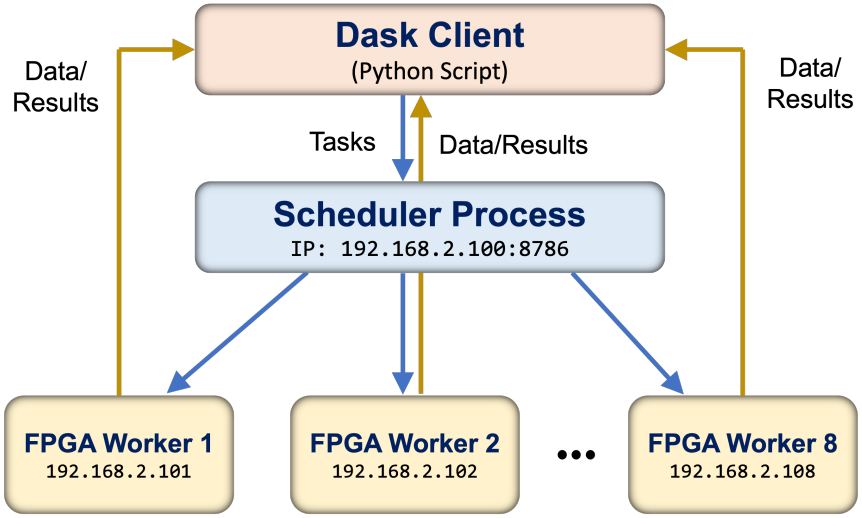
Pynq Board and Framework

- 32-bit ARM core (Processing System)
- Xilinx FPGA (Programmable Logic)
- Jupyter access over web browser

Dask.distributed Python Library

- Distributed computation APIs
- Lightweight, dynamic task scheduler
- Supports complex workflows

Cluster with 8 Pynq-Z2 boards, router, switch and powered USB hub



Pynq-Dask Cluster Architecture

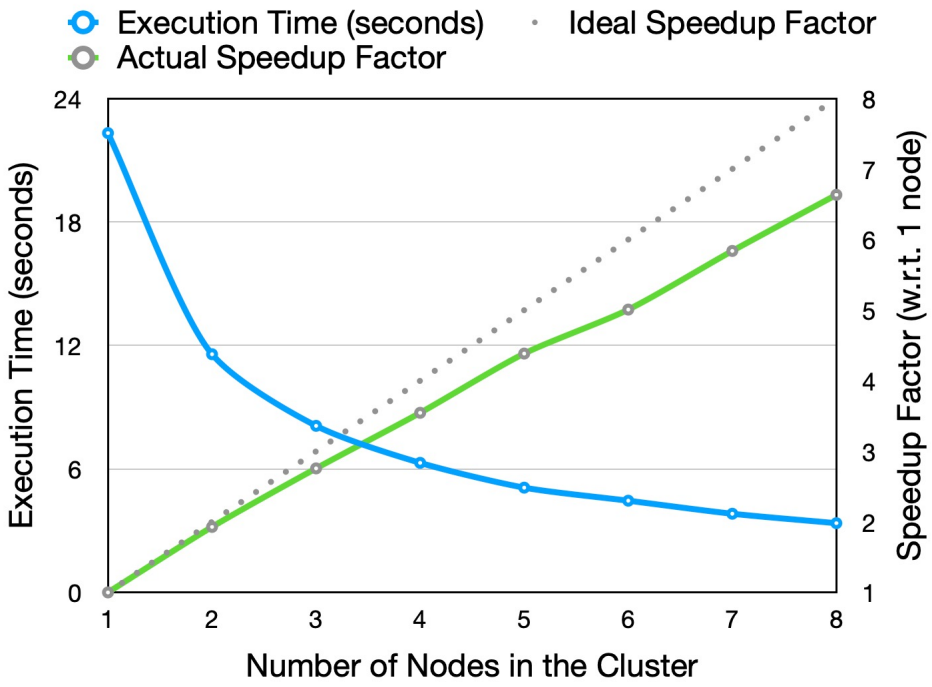
Distributed Computation Demo

Dask divides computation task equally among connected nodes.

$$N = 100\ million$$

$$X = [Randomly\ initialized\ array\ with\ N\ elements]$$

$$Y = \sum_{i=0}^{N-1} e^{X[i]}$$



Deviation from ideal speedup due to **network overhead!**

Federated Learning Simulation

- 🎯 Analyze communication cost associated with a FL model
- 🔍 Compare performance against a centralized model

Model

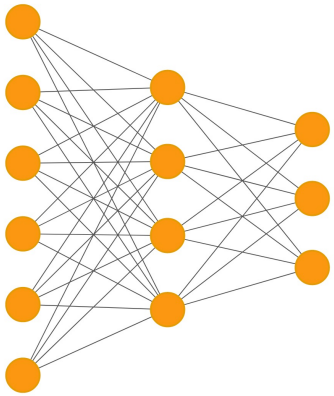
- 2-layer MLP classification model
- 784 x 100 x 10
- Activations: ReLU and Softmax

Dataset

- MNIST (10 labels)
  - 50k train, 10k test
- IID training data (Ideal)
  - Equal data/label
  - Even distribution of labels/node

Cluster Configuration

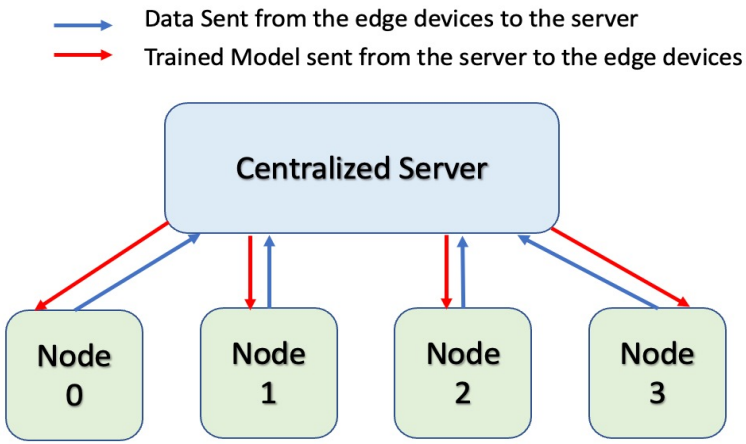
- 1 central server (main model)
- 2, 4, and 8 clients for decentralized, federated training

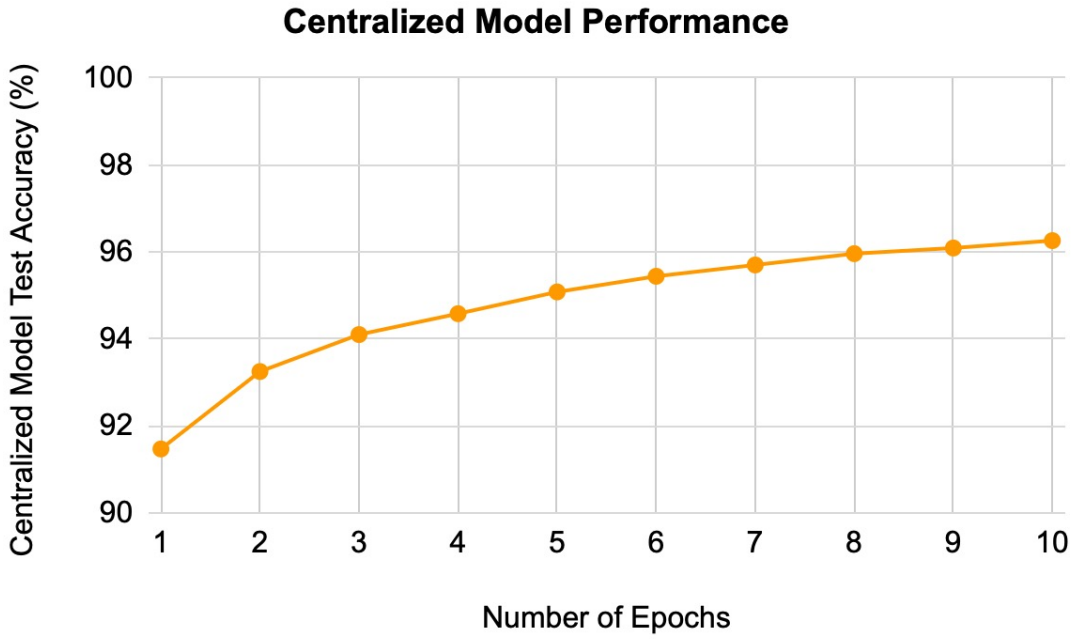


http://alexlenail.me/NN-SVG/index.html

Baseline: Centralized Model Training

- Training Configuration:
  - Total training data: **44960** images
  - Number of epochs: **10**
  - Batch size: **100**
- Total test data: **10000** images

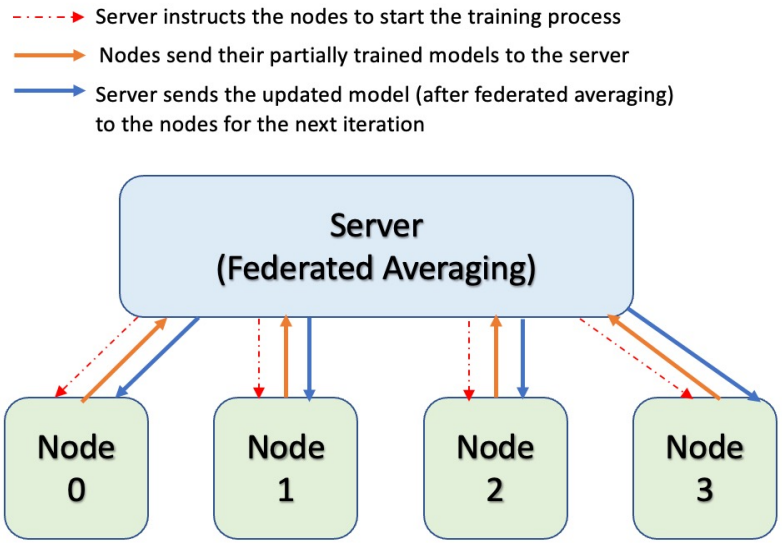




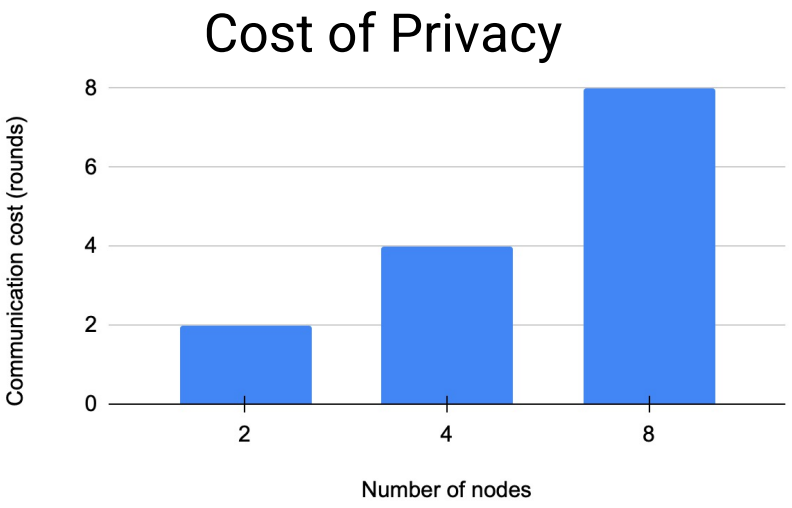
Baseline test accuracy after 10 epochs:

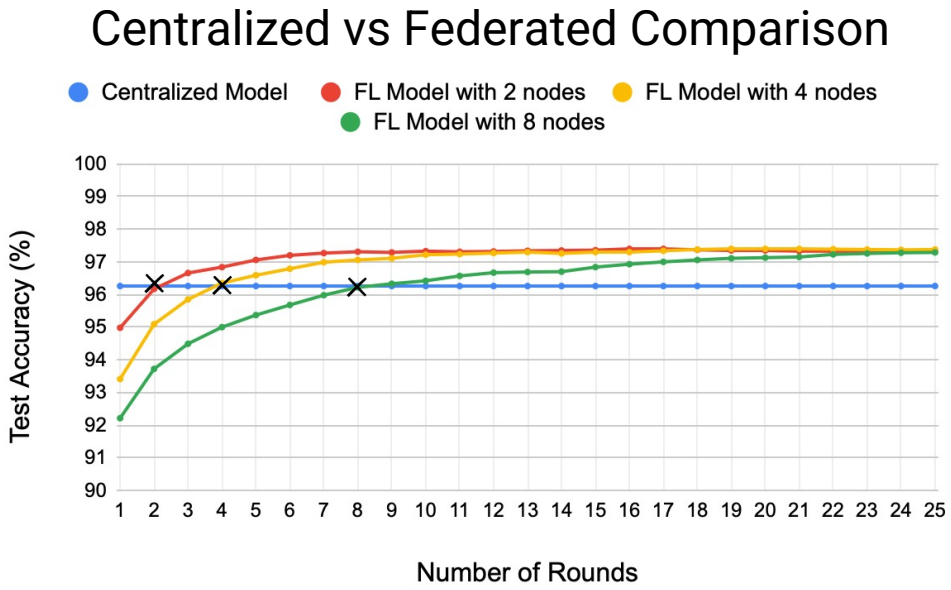
96.26%

Performance Analysis of FL Model



#Nodes	Train/label/node
1	4496
2	2248
4	1124
8	562

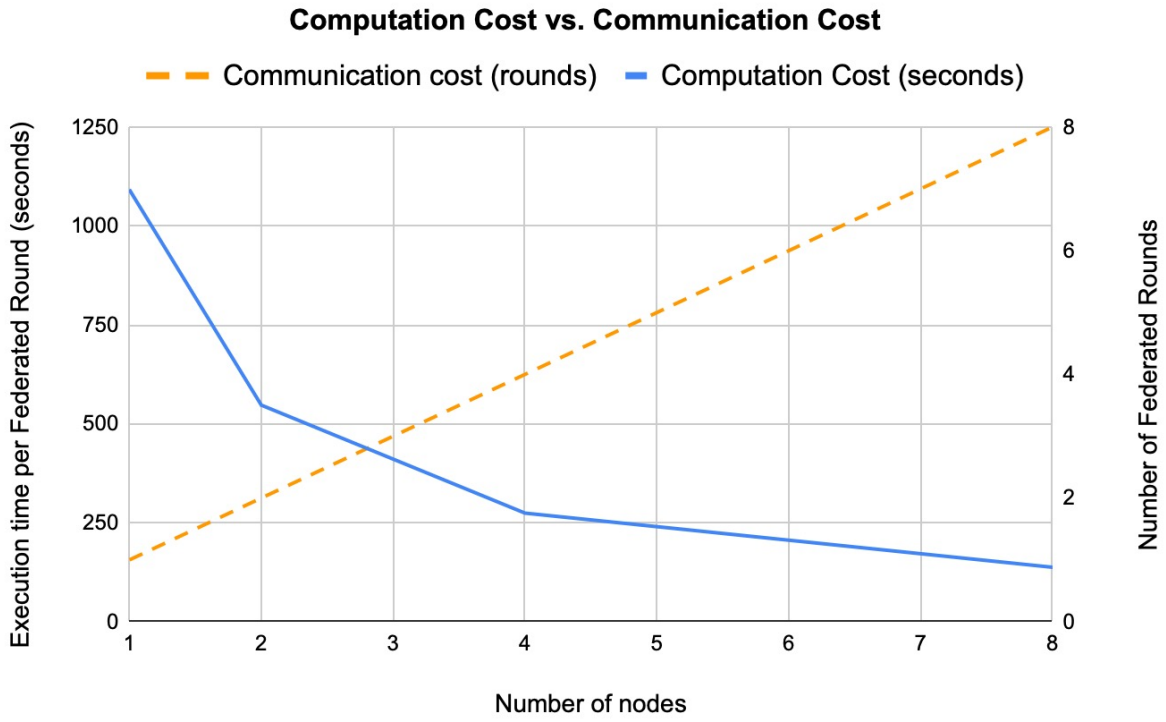




**Training and test configuration same as the centralized model for fair evaluation*

Deploying FL Model on Pynq Cluster

- FL model deployed on a homogeneous cluster with 2, 4, and 8 nodes.
- Processing System (PS) of Pynq-Z2 boards used for on-device training.



Number of nodes ↑

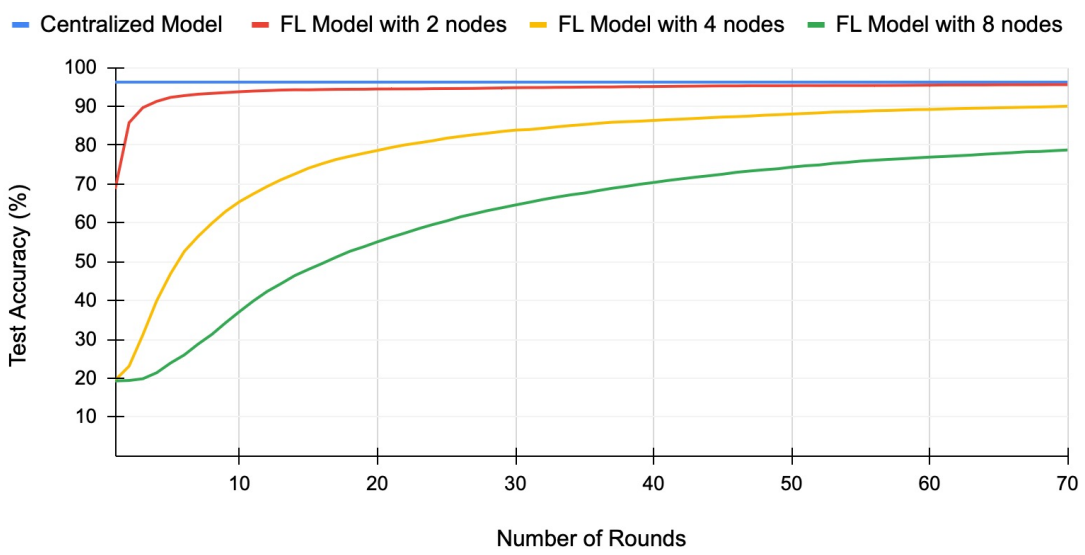
Hardware burden ↓

Network burden ↑

Practical Case: Non-IID Data

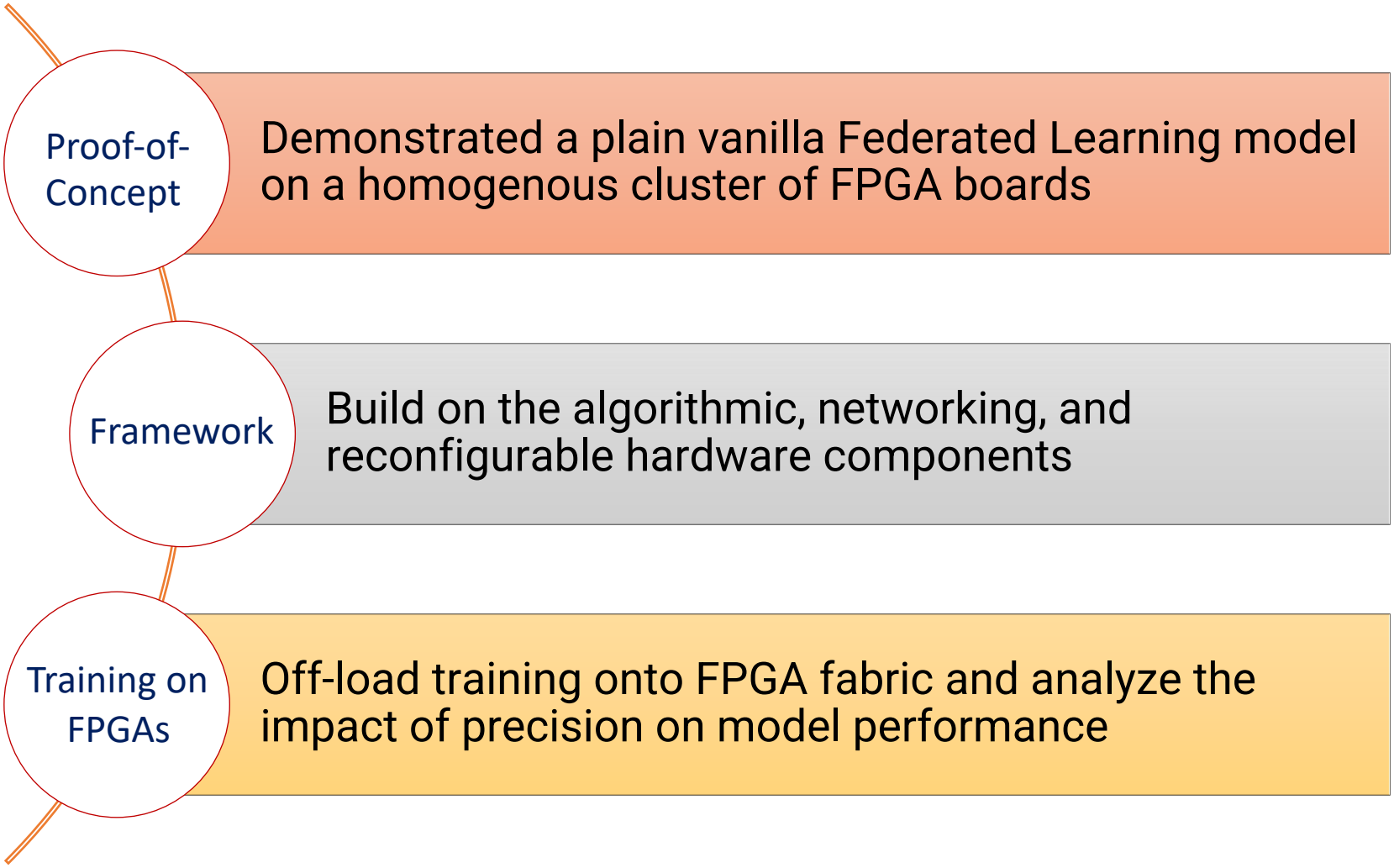
- Practical Scenario:
  - Unequal training data/node
  - **Unequal labels/node**
- No. of federated rounds **highly sensitive** to the data distribution per node
- Federated rounds required are significantly **higher** due to uneven data distribution

FL Model Analysis with non-IID Data



0 - 4	0 – 1 8 - 9	2 - 3 8 - 9	0 8 - 9	1 8 - 9	2 8 - 9	3 8 - 9
5 - 9	4 - 5 8 - 9	6 - 7 8 - 9	4 8 - 9	5 8 - 9	6 8 - 9	7 8 - 9
Label Distribution for 2 nodes			Label Distribution for 4 nodes			Label Distribution for 8 nodes

Summary and Future Work



Design Choices and Final Thoughts

Why Pynq, not Pi?

Multi-FPGA extension to FL relatively unexplored

Why Dask, you ask?

Powerful APIs abstract data packet handling

Our project: Simplified FL on homogeneous cluster

Practical challenges with FL:

Heterogeneity, Uneven data distribution

Malicious nodes, Node dropping

An Interesting Read!

https://federated.withgoogle.com


THANK
YOU!
BIG DOT OF HAPPINESS