MLPerf

A Benchmark for Machine Learning from an Academic/Industry Cooperative

Researchers from:
Baidu, Google, Harvard, Stanford, and UC Berkeley

Contributors (<u>Presenting</u>)

Baidu: Siddharth Goyal, Sharan Narang, Greg Diamos

Google: Karmel Allison, Victor Bittorf, Kathy Wu, Cliff Young, Peter Mattson

Harvard: <u>Udit Gupta</u>, Lillian Pentecost, Brandon Reagen, Gu-Yeon Wei

Stanford: <u>Cody Coleman</u>, Daniel Kang, Deepak Narayanan, Peter Bailis, Matei Zaharia

University of California, Berkeley: Ion Stoica, David Patterson

MLPerf

Quick Computer Benchmark History

Benchmark	*Metric*	*When*
Gibson Instruction Mix (Frequency of instructions)	MIPS: Million Instructions Per Second	1970
Whetstone, Dhrystone (Synthetic programs)	Whetstones, Dhrystones per second	1976,1984
Puzzle, Quicksort (Toy programs)	MIPS	1981
Linpack, Livermore Loops (Kernels)	MFLOPS: Million Floating-Point Operations Per Second	1976,1986

MLPerf

SPEC: System Performance Evaluation Cooperative

- Application level benchmarking (enable via high-level languages and portability of UNIX OS)
- Cross-platform benchmarking and evaluation
- Industry and academia to join at reasonable cost
- Standard in marketplace, papers, and textbooks

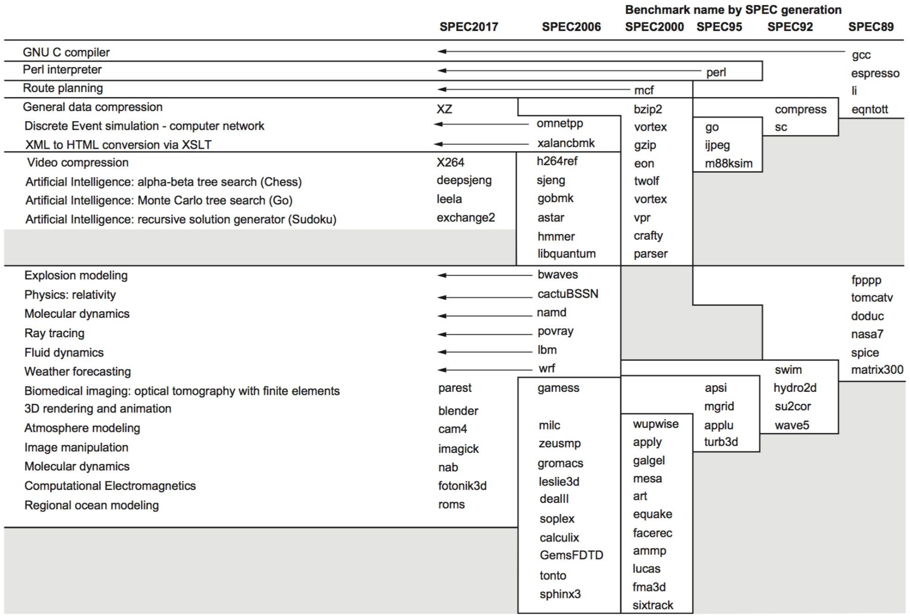


Fig 1.17,
Computer Architecture: A Quantitative Approach, 6th Edition, 2018

MLPerf

Companies:

AMD	Baidu	Google	Intel	Sambanova	Wave Computing

Researchers from these educational institutions:

Harvard University	Stanford University	University of California, Berkeley	University of Minnesota	University of Toronto

Goals for MLPerf

1. Accelerate progress in ML via fair and useful measurement
2. Encourage innovation across state-of-the-art ML systems
3. Serve both industrial and research communities
4. Enforce replicability to ensure reliable results
5. Keep benchmark effort affordable so all can play

Difficulties of ML Benchmarking

1. Diversity in deep learning models used
    a. Problem domain
    b. Models
    c. Datasets
2. Pace of field
    a. State-of-the-art models evolve every few months
3. Lack of evaluation metric
    a. Accuracy
    b. Time to train, latency of inference
4. Multi-disciplinary field
    a. Algorithms, Systems,, Hardware

MLPerf

Outline

- **Model diversity**
- Agile benchmark development
- Evaluation metrics
- Open and closed divisions
- Contributing to MLPerf

Fathom suite showed breadth in ML benchmarking

- Seq2Seq
- MemNet
- Speech
- Autoenc
- Residual
- VGG
- AlexNet
- DeepQ

- Collection of 8 diverse learning models
- Clear, tested implementations in TensorFlow
- Training and inference modes provided
- Provided broad view and coverage
- Models have drastically changed and greatly advanced since 2015

Benchmarks Considered for MLPerf

Area	*Vision*	*Language*	*Audio*	*Commerce*	*Action / RL*	*Other*
Problem	Image Classification Object Detection / Segmentation Face ID HealthCare (Radiology) Video Detection Self-Driving	Translation Language Model Word Embedding	Speech Recognition Text-to-Speech Question Answering Keyword Spotting Language Modeling Chatbots Speaker ID Graph embeddings Content ID	Rating Recommendations Sentiment Analysis Next-action Healthcare (EHR) Fraud detection Anomaly detection Time series prediction Large scale regression	Games Go Robotics Health Care Bioinformatics	GANs 3D point clouds Word embeddings
Datasets	ImageNet COCO	WMT English-German	LibriSpeech SQuAD LM-Benchmark	MovieLens-20M Amazon IMDB	Atari Go Chess Grasping	
Models	ResNet-50 TF Object Detection Detectron	Transformer OpenNMT	Deep Speech 2 SQuAD Explorer	Neural Collaborative Filtering CNNs	DQN PPO	
Accuracy Metrics	COCO mAP Prediction accuracy	BLEU	WER Perplexity	Prediction accuracy	Prediction accuracy Win/Loss	

MLPerf

MLPerf benchmarks (version 0.5)

Area	Benchmark	Dataset	Model	Reference Implementation
Vision	Image classification	ImageNet	ResNet	TensorFlow
	Object detection	COCO	Mask R-CNN	Caffe 2
Language/ Audio	Translation	WMT Eng-Germ	Transformer	TensorFlow
	Speech recognition	LibriSpeech	Deep Speech 2	PyTorch
Commerce	Recommendation	MovieLens-20M	NCF	PyTorch
	Sentiment Analysis	IMDB	Seq-CNN	PaddlePaddle
Action	Reinforcement Learning	Go	Mini-go	TensorFlow

- Balance benchmarks that represent
    - Industry workloads
    - Coverage of different areas and characteristics

MLPerf

Outline

- Model diversity
- **Agile benchmark development**
- Evaluation metrics
- Open and closed divisions
- Contributing to MLPerf

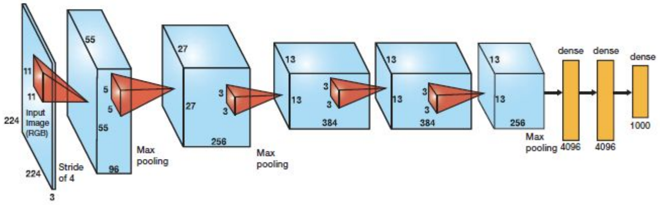
Agile Benchmark Development

- Rapidly iterate the benchmark suite:
    - Remain relevant in the very fast moving ML field

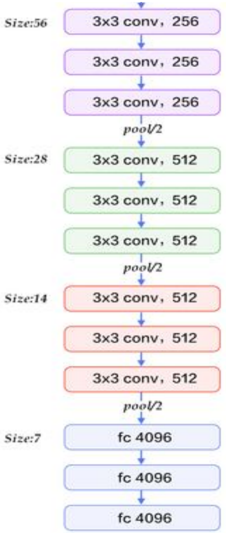
Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field

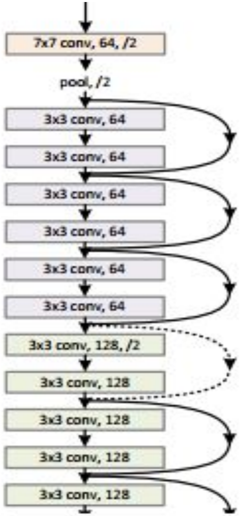
AlexNet (2012)



VGG16 (2014)



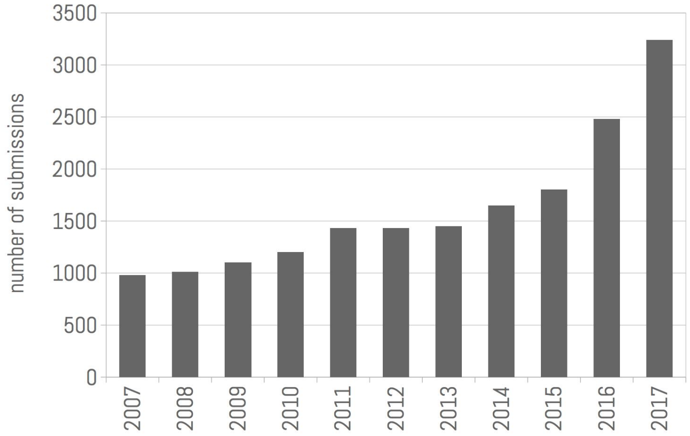
ResNet (2015)



MLPerf

Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field



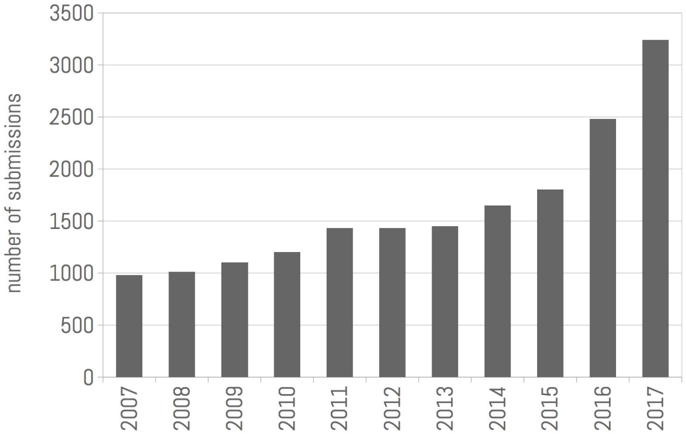
From Samy Bengio's opening remarks at NIPS 2017

MLPerf

Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field

NIPS 2017 had **3240 submissions**
NIPS 2018 had **~4900 submissions**



From Samy Bengio's opening remarks at NIPS 2017

MLPerf

Agile Benchmark Development

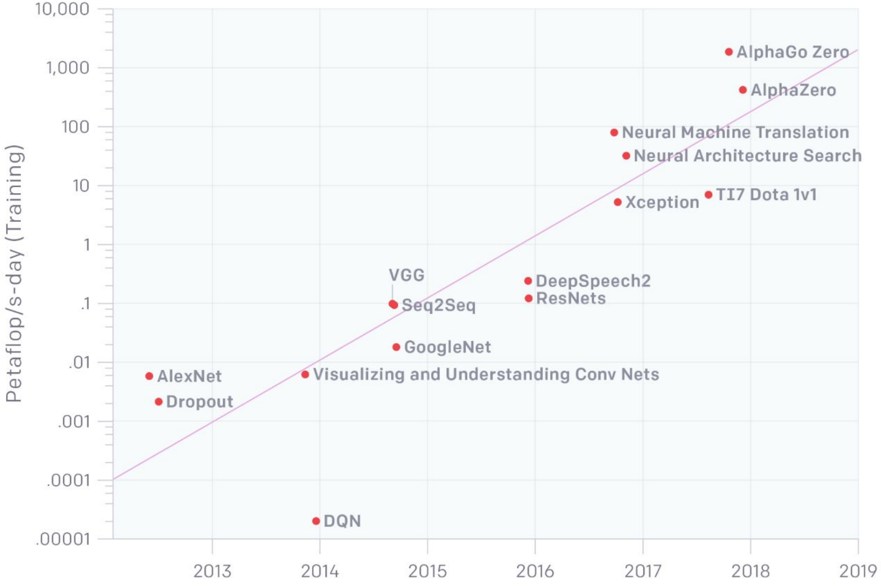
- Rapidly iterate the benchmark suite:
    - Remain relevant in the very fast moving ML field
    - Scale problems to match faster hardware

MLPerf

Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field
  - Scale problems to match faster hardware

A **300,000x Increase in Compute** since 2012



From OpenAI Blog "AI and Compute"

Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field
  - Scale problems to match faster hardware
  - Correct inevitable mistakes in the formulation

Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field
  - Scale problems to match faster hardware
  - Correct inevitable mistakes in the formulation
- At least initially, revise annually? MLPerf18, MLPerf19, …

Agile Benchmark Development

- Rapidly iterate the benchmark suite:
  - Remain relevant in the very fast moving ML field
  - Scale problems to match faster hardware
  - Correct inevitable mistakes in the formulation
- At least initially, revise annually? MLPerf18, MLPerf19, ...
- Like SPEC, have quarterly deadlines and then publish results for that quarter via searchable database

Outline

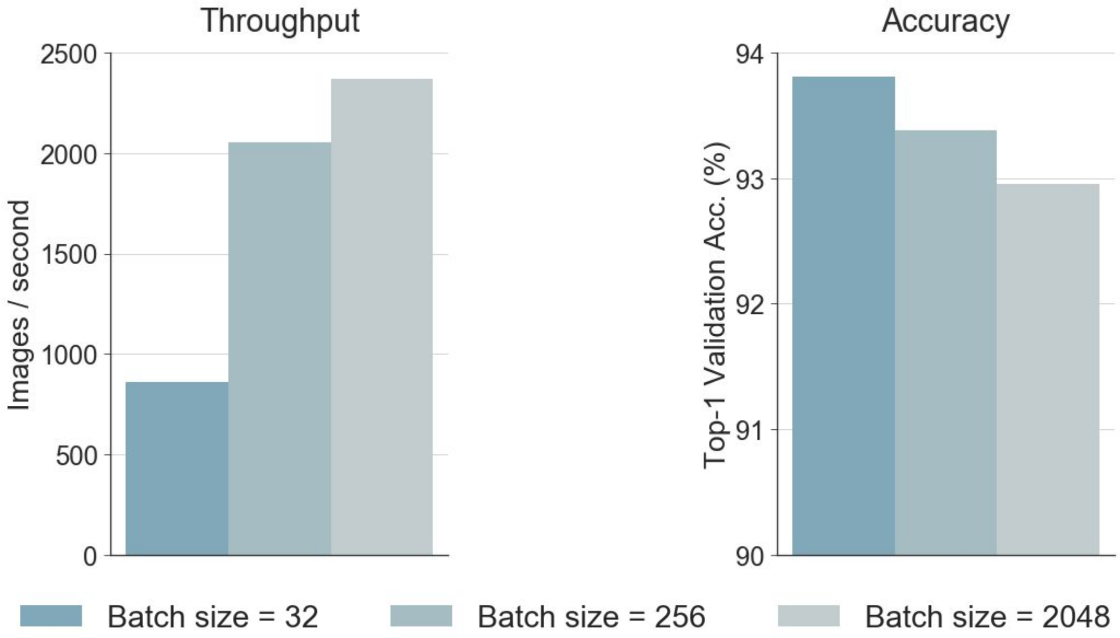
- Model diversity
- Agile benchmark development
- **Evaluation metrics**
- Open and closed divisions
- Contributing to MLPerf

MLPerf

Metrics Should Capture Performance and Quality

- **Performance**: how fast is a model for training, inference?
  - Focus of benchmarks like DeepBench, Fathom
- **Quality**: how good are a model's predictions?
  - Focus of benchmarks like ImageNet, MS COCO

Performance and Quality **aren't always correlated**



End-to-end training of a ResNet56 CIFAR10 model on a Nvidia P100 machine with 512 GB of memory and 28 CPU cores, using TensorFlow 1.2 compiled from source with CUDA 8.0 and CuDNN 5.1.

Metrics Should Capture Performance and Quality

- **Performance**: how fast is a model for training, inference?
- **Quality**: how good are a model's predictions?

Important for benchmark to capture
both performance and quality

2017-18: Stanford DAWNBench *http://dawn.cs.stanford.edu/benchmark/*

Measures Performance (Time, Cost) to Fixed Quality Target

DAWNBench

An End-to-End Deep Learning Benchmark and Competition

Training Time

All Submissions

Objective: Time taken to train an image classification model to a top-5 validation accuracy of 93% or greater on ImageNet.

Rank	Time to 93% Accuracy	Model	Hardware	Framework
1 Apr 2018	0:30:43	ResNet50 *Google* source	Half of a TPUv2 Pod	TensorFlow 1.8.0-rc1

MLPerf

MLPerf metric: **Training time** to reach quality target + cost **or** power

- Quality target is *specific for each benchmark* and *close to state-of-the-art*
  - Updated w/ each release to keep up with the state-of-the-art
  - Median of 5 runs
- Time includes preprocessing and validation
- Reference implementations that achieve quality target

In addition, *either*:

- *Cost* of public cloud resources *(no spot/preemptible instances)*
- *Power utilization* for on-premise hardware

MLPerf

Summary result combines benchmark metrics

Why?

- Provide a concise indicator of “general purpose ML” performance
- Encourage the field to move in a common direction, ultimately leading to greater performance across the board

How? *For participants that submit to each benchmark category:*

- For each benchmark task, normalize the time result to the reference implementation on baseline hardware
- Summary score computed via geometric mean of results

MLPerf

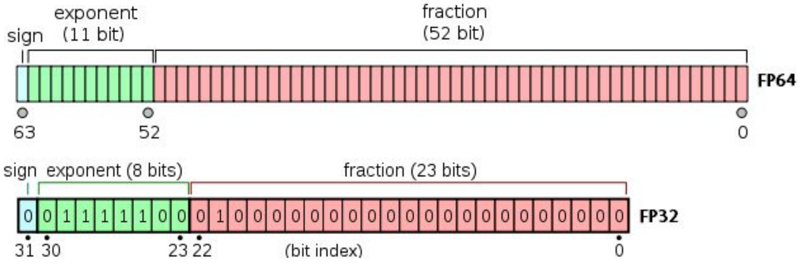
Outline

- Model diversity
- Agile benchmark development
- Evaluation metrics
- **Open and closed divisions**
- Contributing to MLPerf

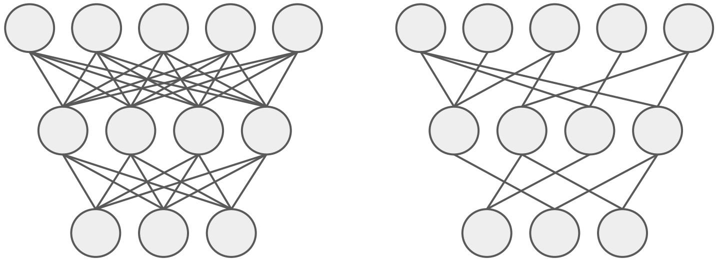
Goal: Encourage Innovation and fair comparison

Goal: **Encourage Innovation** and fair comparison

- ML algorithms are under active development



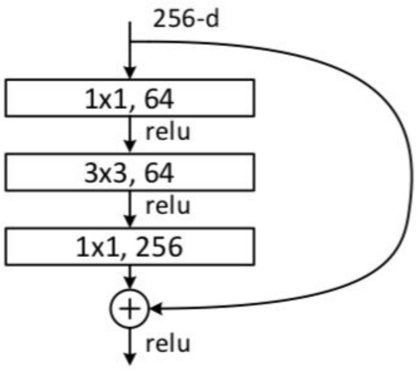
low precision



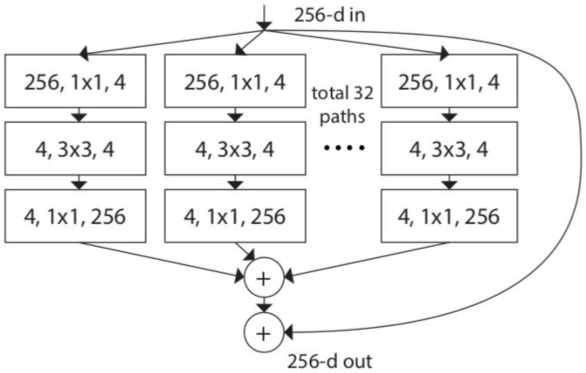
sparsity

Goal: **Encourage Innovation** and fair comparison

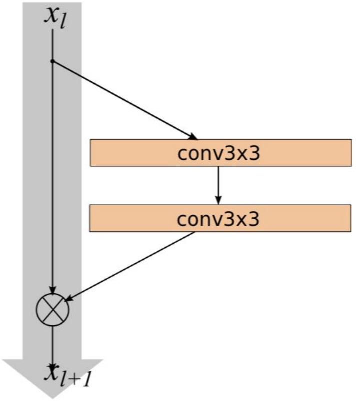
- ML algorithms are under active development
- Many models with different trade-offs



ResNet



ResNeXt

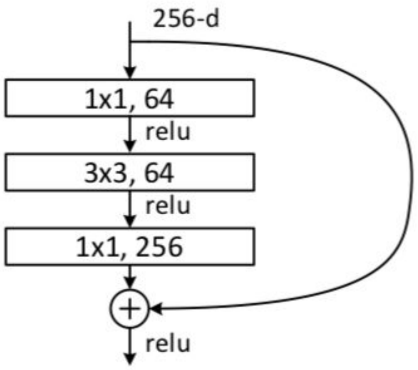


Wide ResNet

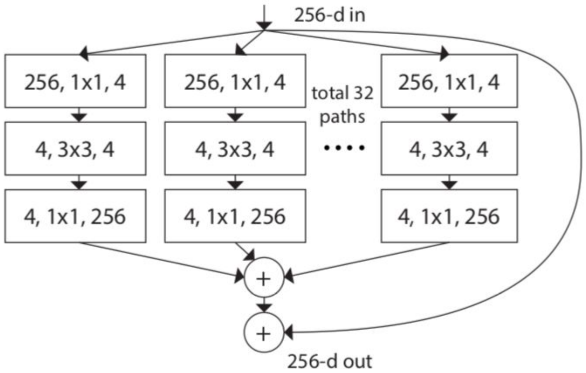
MLPerf

Goal: Encourage Innovation and **fair comparison**

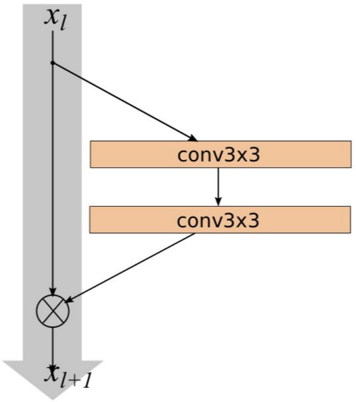
- ML algorithms are under active development
- Many models with different trade-offs



ResNet



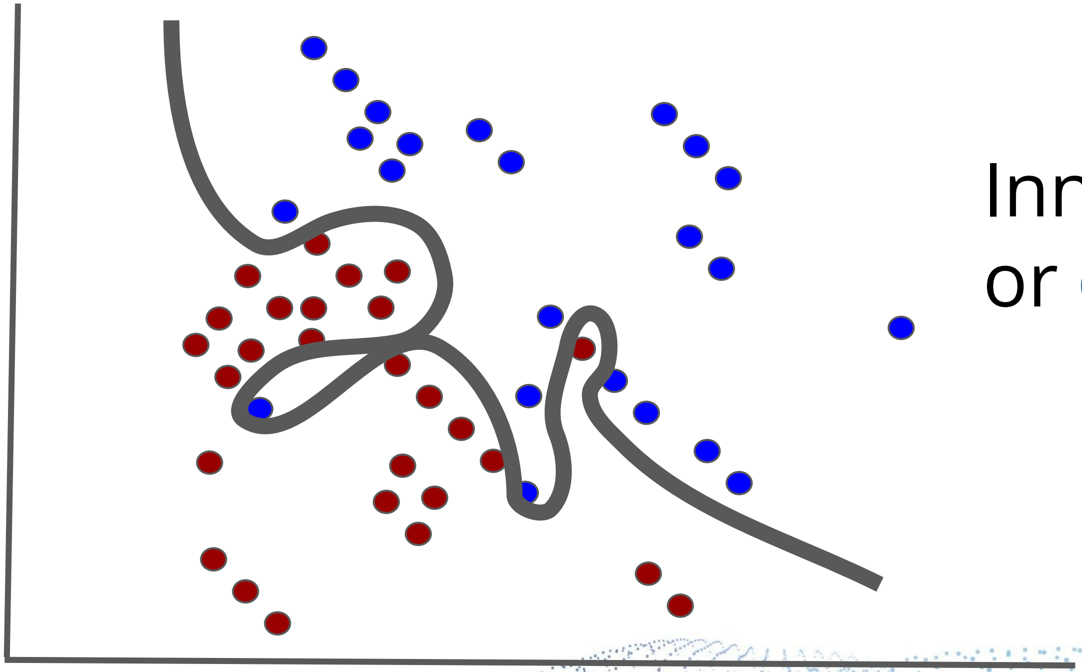
ResNeXt



Wide ResNet

MLPerf

Goal: Encourage Innovation and **fair comparison**



Innovative algorithm or **overfitting**

MLPerf

Open/Closed Divisions + Replication

- **Closed** division requires using the specified model
  - Limits overfitting
  - Enables apples-to-apples comparison
  - Simplifies work for HW groups

- **Open** division allows using any model
  - Encourages innovation
  - Ensures closed division does not stagnate

Outline

- Model diversity
- Agile benchmark development
- Evaluation metrics
- Open and closed divisions
- **Contributing to MLPerf**

Plan: move fast, become independent standard

- Start as small cooperative to quickly publish good benchmark suite soon
- Invite every like-minded group who shares the goals of MLPerf:
    - Big companies
    - Startups
    - Universities
- Current version “0.5”. For 1.0, transfer to independent org.

Ways to support and be involved

- **Github**: reference code
- **Submissions**: data points
- **Google group**: discussion of the benchmark and changes
- **Meetings**: community building and focused discussion towards action
- **Working groups**: targeted groups to flesh out specific areas
    - Inference
    - Reinforcement learning
    - Summary score
    - Measuring power and cost

More at **MLPerf.org**, or contact **info@mlperf.org**