TensorFlow
TensorFlow (2015)

- Originally developed at Google Brain
- Open-source library written in C/C++ to build and run neural network models
  - Plugs into CUDA/GPUs as a result
  - [http://tensorflow.org](http://tensorflow.org)
- Supports numerical computation using data flow graphs
  - Must be a directed, acyclic graph (DAG)
  - Mostly for convolutional neural networks
- Makes programming machine learning models easier
  - Bindings in Haskell, C++, Python, Go, R, Rust
  - Python most popular (Cython)
- Alternatives
  - PyTorch, Caffe
TensorFlow

- Multidimensional array data (Tensor)
- Flowing through directed, acyclic graph (Flow)
- TensorFlow maps graph across compute nodes via compiler
  - Tinker-toy building block paradigm similar to Dataflow
  - Complexity hidden from model developer
- Both high-level and low-level APIs in TensorFlow
  - Level of abstraction being raised to make programming easier
  - #1 repository in ML on github (5x > than next)
- Scales with # of processors to support both simple and complex models
- Can be installed on your own cluster or in cloud for portability
Two steps

- **Training**
  - Model developed via labeled data to generate weights in graph (weeks to months)
  - Training times vs. recall and accuracy rates
- **Inferencing**
  - Unlabeled data given to trained graph to give probability of a specific label
TensorFlow acceleration

- Can run on any computational engine
  - CPU
  - FPGA (Field Programmable Gate Arrays)
  - ASIC (Application Specific Integrated Circuit)
  - CPU => FPGA => ASIC
    - More general => Special purpose
    - High power consumption => Low power consumption
  - Similar to Bitcoin miners (dominated by ASICs)
- TPU: Tensor Processing Unit
  - ASIC for TensorFlow processing
  - 1st-Gen TPU
    - Focus on running the model
    - Designed for inference, not training
  - 2nd-Gen TPU (Cloud TPU)
    - Both inference and training
    - 180 Tflops, 64 GB high-bandwidth memory
    - TPU pods of 64 each (11.5 Petaflops)
    - Available on Google Cloud since Fall 2017
Related tools

- TensorFlow Board
  - Visualize graph
- TensorFlow Serving
  - Distribute models that have been trained for inferencing
  - Example: model can be placed behind a REST endpoint in the cloud (ML APIs)
- TensorFlow Lite (deploy models onto limited devices)
  - Fusion, compression, etc.
  - Model can be placed on any device (even Raspberry Pis)
CS 410/510: Cloud Machine Learning Engine
Motivation

- Traditional machine learning research
  - Researchers acquire data locally
  - Models run locally
  - Data copying requires lots of infrastructure that only well-endowed groups can purchase
  - Fixed compute not shared
  - Security issues
Democratizing Machine Learning

- [https://www.youtube.com/watch?v=Rgggdddl018](https://www.youtube.com/watch?v=Rgggdddl018) (3/2017)
  - Put it in the cloud
  - Make it easy to use
- Two parts
  - Democratize access to pre-trained models
    - See prior ML APIs
  - Democratize training
    - Bring researchers to public collections of data (see BigQuery datasets)
    - Provide elastic compute and storage
      - Allow those without a cluster or fully-decked out GPUs to do ML
    - Centralize security
Cloud Machine Learning Engine

- Managed TensorFlow
  - Fully managed infrastructure for running TensorFlow workloads
  - NoOps machine learning service
  - Access to cloud storage/data warehouses
Labs
ML Lab #1

- Go to https://playground.tensorflow.org (~20 min)
- Build and train a neural network to detect spiral pattern
ML Lab #2

- TensorFlow for Poets locally (21 min)
  - https://codelabs.developers.google.com/codelabs/tensorflow-for-poets
ML Lab #3 (Option #1)

- TensorFlow and deep learning, without a PhD (149 min)
  - https://codelabs.developers.google.com/codelabs/cloud-tensorflow-mnist/
ML Lab #3 (Option #3)

- Serverless Machine Learning (143 min)
  - [https://codelabs.developers.google.com/codelabs/dataengineering-machine-learning](https://codelabs.developers.google.com/codelabs/dataengineering-machine-learning)
  - 1-2-3-6
  - 7 (TensorBoard)
  - 8 is broken (Cloud MLE)
ML Lab #3 (Option #2)

• Carry out ML with TensorFlow (150 min)
  • Datalab, BigQuery, TensorFlow, to perform ML on Taxicab demand
• Create ML dataset with BigQuery
  • [https://codelabs.developers.google.com/codelabs/cpb100-bigquery-dataset](https://codelabs.developers.google.com/codelabs/cpb100-bigquery-dataset)
  • Launch Cloud Datalab VM, create taxi dataset in BigQuery, open Jupyter notebook
  • Repeat commands from github link to generate results in your own Datalab notebook
  • Continue all the way through the github walkthrough to the Machine Learning for Tensorflow
• Then, run the ML for Tensorflow section
  • Note that the following lab is the second half of the above exercise
    [https://codelabs.developers.google.com/codelabs/cpb100-tensorflow](https://codelabs.developers.google.com/codelabs/cpb100-tensorflow)
SurveyMonkey survey

- [https://www.surveymonkey.com/r/6V5BWBR](https://www.surveymonkey.com/r/6V5BWBR)
Extra slides
Misc. Google Cloud

- **AI**
  - Convolutional networks tied into re-inforcement learning
    - Need to scale
  - $10bn (turned into $30b) investment for ML infrastructure
    - TPU clusters (data centers are data campuses)
    - Water cooling, heating towers, triple redundancy, water facility, power facility
      - AI controls the AI to control power consumption
      - Shifting workload around campus to reduce peak power in one location
  - IP on outside, open-source GRPC w/ microsecond latency on inside (600 Gpbs)
    - Looking to quantum entanglement
    - Need sneakernet to do 100PetaBytes otherwise 124 days
    - Working on quantum entangled communication to exceed speed of light
http://playground.tensorflow.org/#activation=relu&batchSize=10&dataset=spiral&regDataset=reg-gauss&learningRate=0.1&regularizationRate=0&noise=0&networkShape=7,7,2&seed=0.27041&showTestData=true&discretize=true&percTrainData=50&x=true&y=true&xTimesY=true&xSquared=true&ySquared=true&cosX=false&sinX=true&cosY=false&sinY=true&collectStats=false&problem=classification&initZero=false&hideText=false