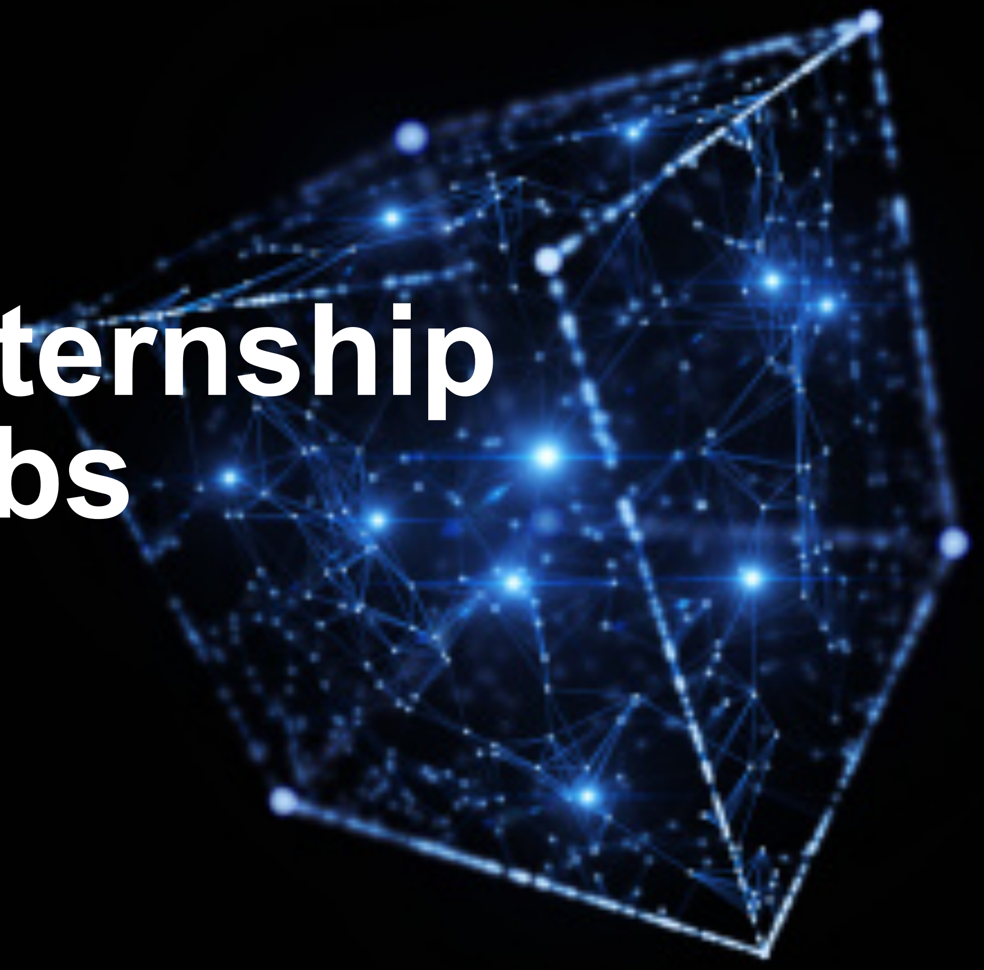
Analytics Research Internship at Hewlett Packard Labs



Stefanie Deo

Mentor: Mehran Kafai

September 12, 2016

First, another opportunity that came my way but didn't pan out: Data Science Internship at Intuit

- I joined local Meetup groups for women who code/women in data science:

pyladies San Francisco

R- ladies

Bay Area Women in Machine Learning & Data Science

- Through Pyladies SF, I heard about an intro to data science class at Intuit (Oct 1 – Nov 12)
- I applied for the class and was accepted
- An Intuit recruiter got in touch with me after my final project presentation
- I got a phone interview, but wasn't selected (and that's okay!)

How I got an Research Internship at HPE’s Analytics Lab

- Old-fashioned way: I made a list of all the big tech companies and searched for jobs directly on each company’s website
- I started looking really early; I sent out my first resume in October
- I checked around every week, and then found this internship on HPE’s job postings page (I applied for it on January 14)
- In March, I got a request from HPE for a phone interview
- By that time, I had applied for **over 50 internships** (I had to keep a spreadsheet to organize my search!)
- One interview question was about prior statistical work (for that, I talked about my Math 261A project)
- Hardest question was about programming, but I was able to answer it on the second try because of what I learned in CS21A (Python Programming at Foothill College) and Math 167 (SAS Programming at SJSU)

Hewlett Packard
Labs

A little bit about the company

Hewlett Packard Enterprise ≠ hp

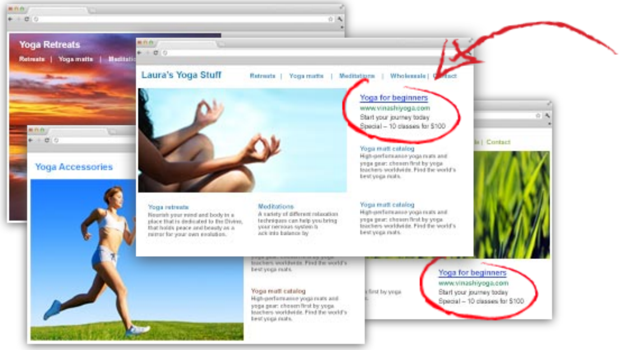
- HPE became a separate company after Hewlett Packard split into two companies last Nov 2015
- As the name states, they focus on enterprise products (servers, storage, networking, consulting, and software for big data analysis and security)
- Hewlett Packard Labs (in Palo Alto) is their R&D division which comes up with new products

- HP Inc. makes printers, laptops, and other consumer goods

What I worked on:
A machine learning classification tool

Predictions from classification models can help us solve all sorts of problems

AD TARGETING: Which ad is a user most likely to click on?



Yoga Retreats

Retreats | Yoga matts | Meditatio

Laura's Yoga Stuff

Retreats | Yoga matts | Meditations | Wholesale | Contact

Yoga for beginners
www.vinashiyoga.com
Start your journey today
Special – 10 classes for $100

Yoga matt catalog
High-performance yoga mats and yoga gear: chosen first by yoga teachers worldwide. Find the world's best yoga mats.

Yoga retreats
Nourish your mind and body in a place that is dedicated to the Divine, that holds peace and beauty as a mirror for your own evolution.

Meditations
A variety of different relaxation techniques can help you bring your nervous system b ack into balance by

Yoga matt catalog
High-performance yoga mats and yoga gear: chosen first by yoga teachers worldwide. Find the world's best yoga mats.

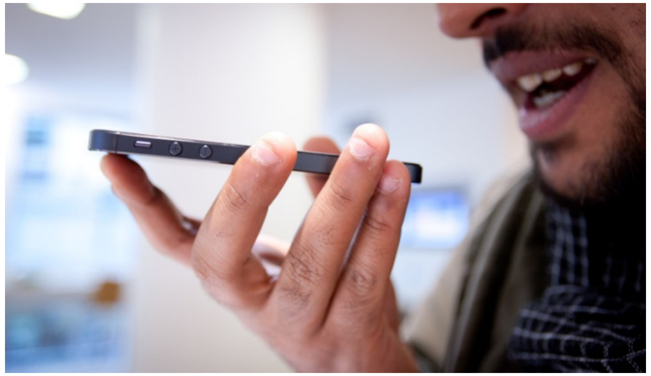
Yoga Accessories

Yoga matt catalog
High-performance yoga mats and yoga gear: chosen first by yoga teachers worldwide. Find the world's best yoga mats.

Contact

Yoga for beginners
www.vinashiyoga.com
Start your journey today
Special – 10 classes for $100

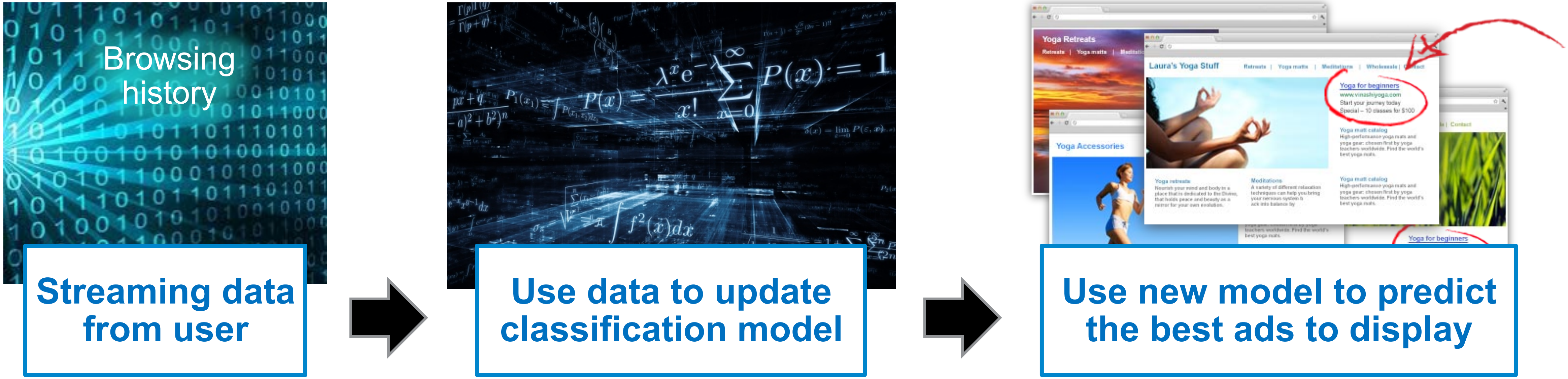
SPEECH RECOGNITION: Did the person say “Call Bobby” or “Call Barbie”?



MEDICAL DIAGNOSIS: Does a patient have Disease X?

With streaming data, however, stale data usually isn’t ideal.

Taking a closer look at a search engine trying to predict ad clicks:



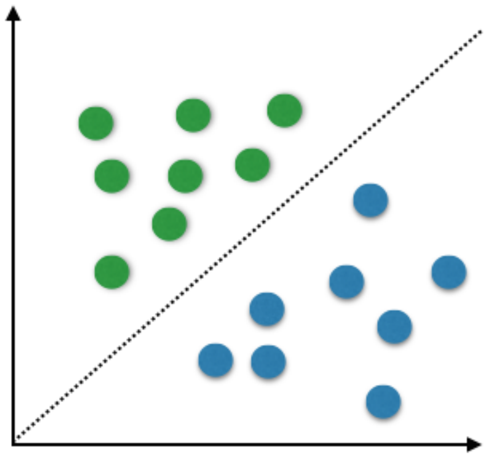
Streaming data from user → **Use data to update classification model** → **Use new model to predict the best ads to display**

…but retraining a model with the most recent data can be challenging when we have large datasets that are rapidly evolving

Using the most recent data requires the shortest possible training time

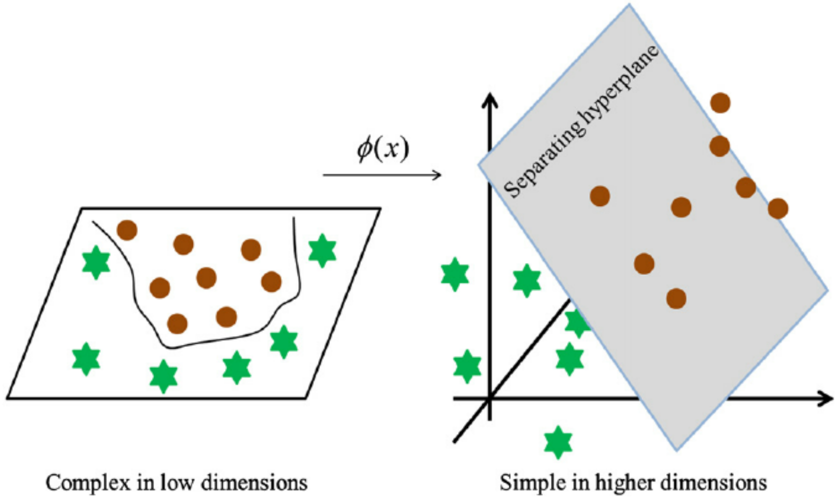
This is a tough problem, due to a historical trade-off between the two classification methods:

Linear classification



Fast, but only suits data that is already linearly separable

Kernel classification



Can handle non-linearly separable data, but is computationally expensive (and therefore slow)

Labs’ solution is CROlinear, a tool that is the best of both worlds

CRO feature map

- fast kernel transformation developed at Hewlett Packard Labs
- able to take data that is not linearly separable, and quickly make it linearly separable in a higher dimension

+

Accelerated Linear classification

- incremental
- multicore
- faster optimization algorithms

=

CROlinear

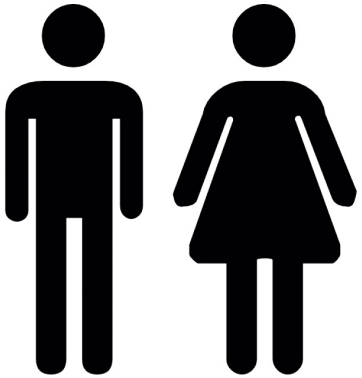
- fast
- can handle data that is not linearly separable

Hewlett Packard Enterprise

San José State UNIVERSITY

CROlinear Experiments (using logistic regression as the solver)

Datasets



Dataset: a9a (Adult census data)

Training Dataset: 32,561 instances

Number of classes: 2

Dataset: real-sim (Real vs Simulated document classification)

Training Dataset: 72,309 instances

Number of classes: 2

Datasets

Dataset: MNIST (handwritten digits)

Training set: 1M instances

Testing set: 100K instances

Number of classes: 10

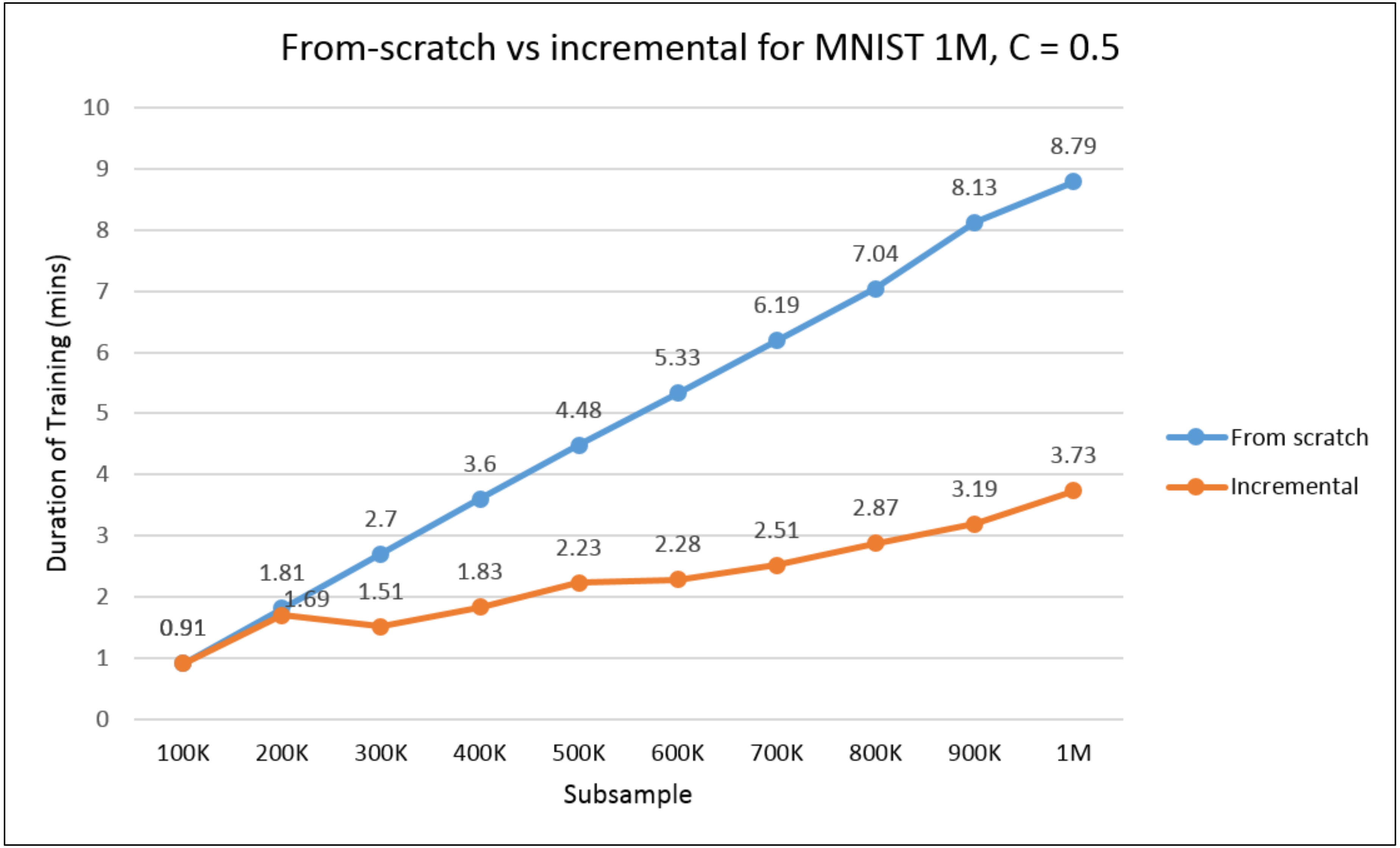
Dataset: TIMIT
(American English speech corpus)

Training set: 1,385,426 instances

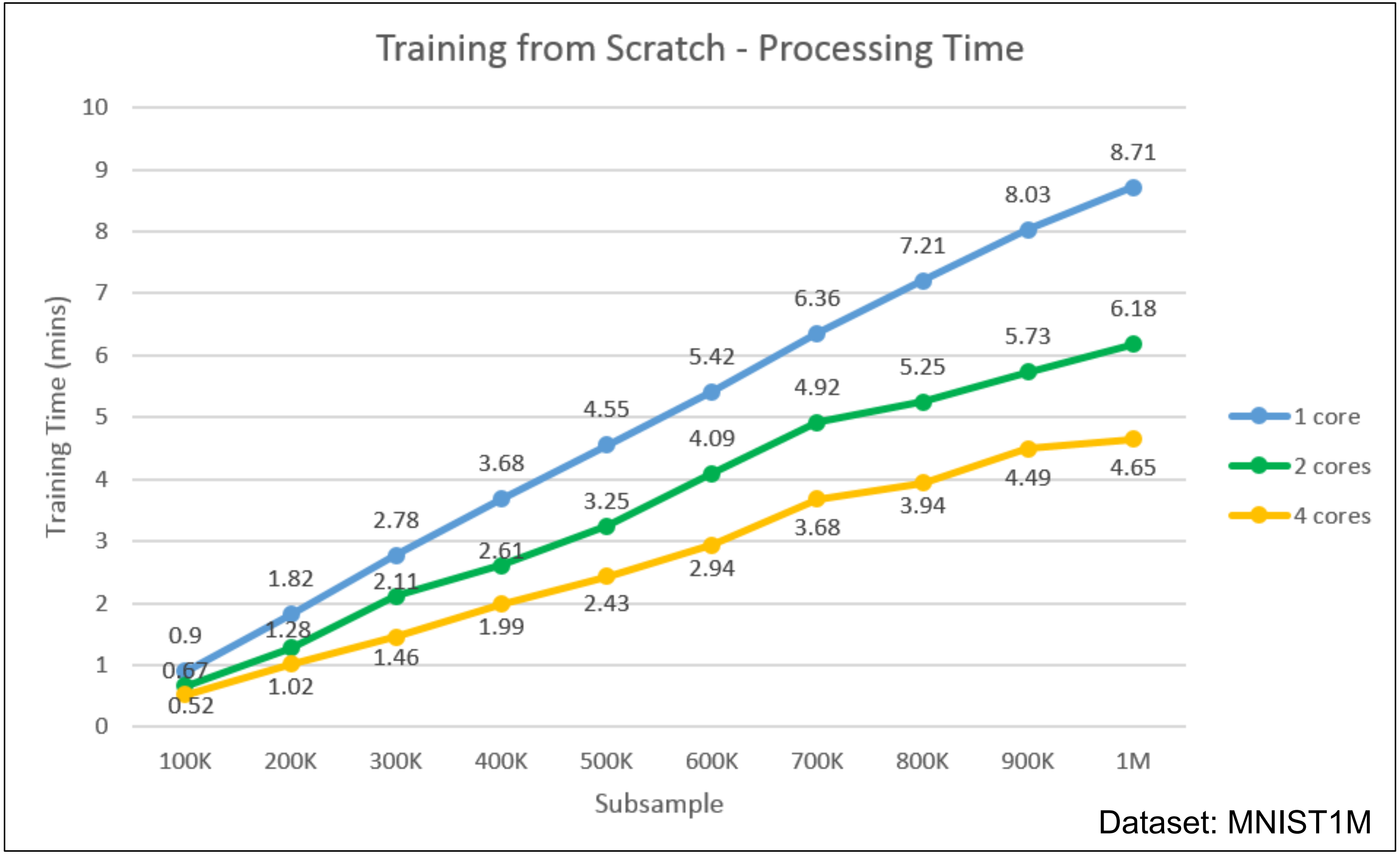
Testing set: 506,113 instances

Number of Classes: 39

Incremental training produces an updated model much faster than training from scratch

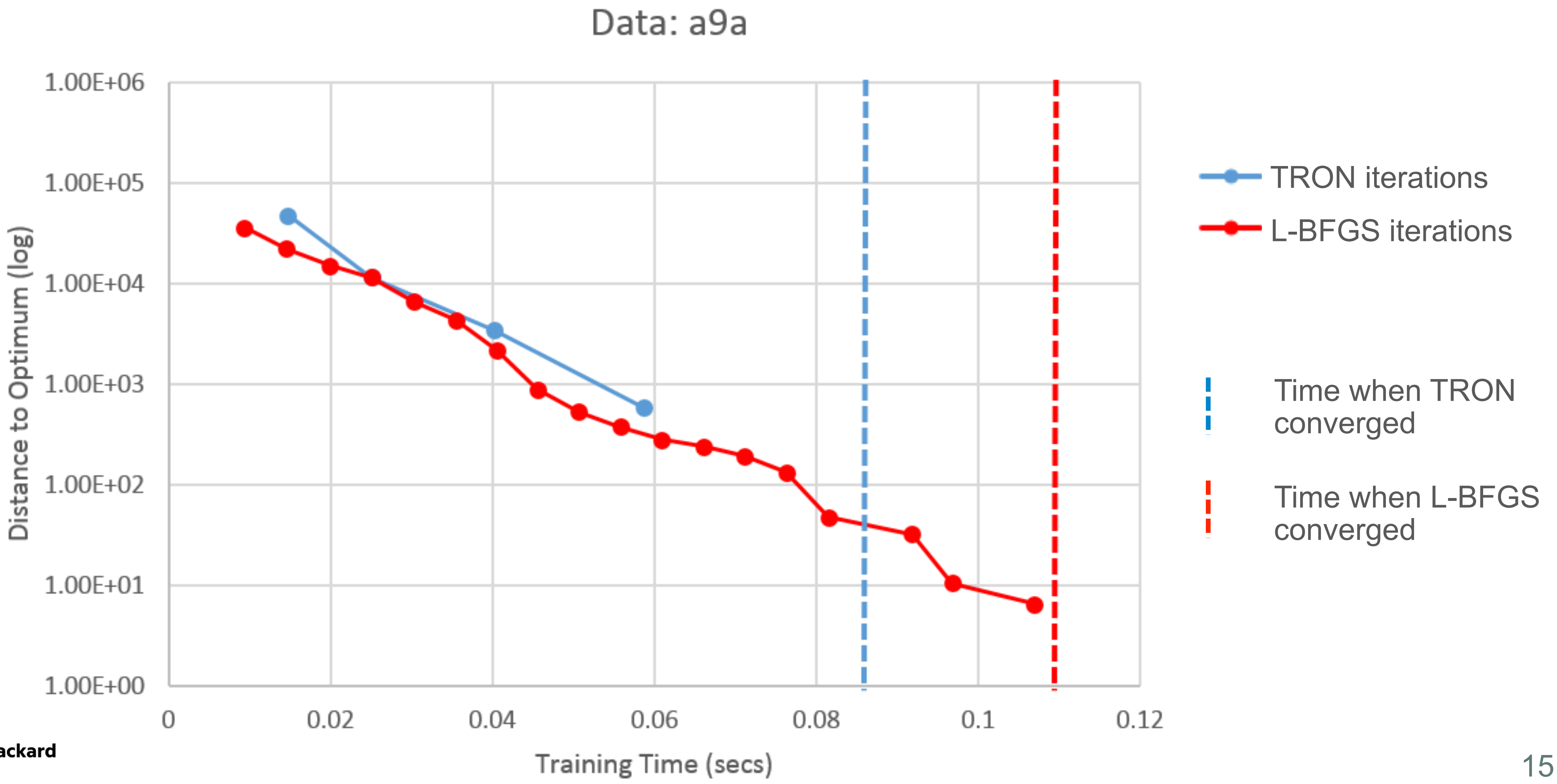


No surprise: training on more cores requires less processing time



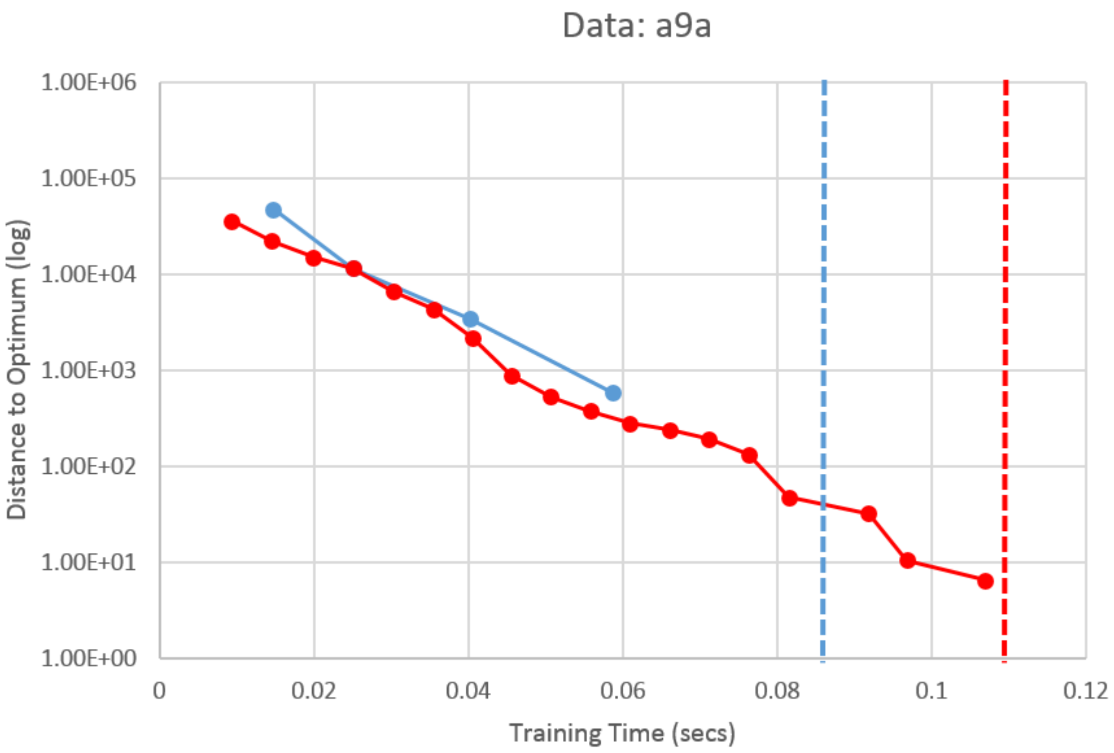
Dataset: MNIST1M

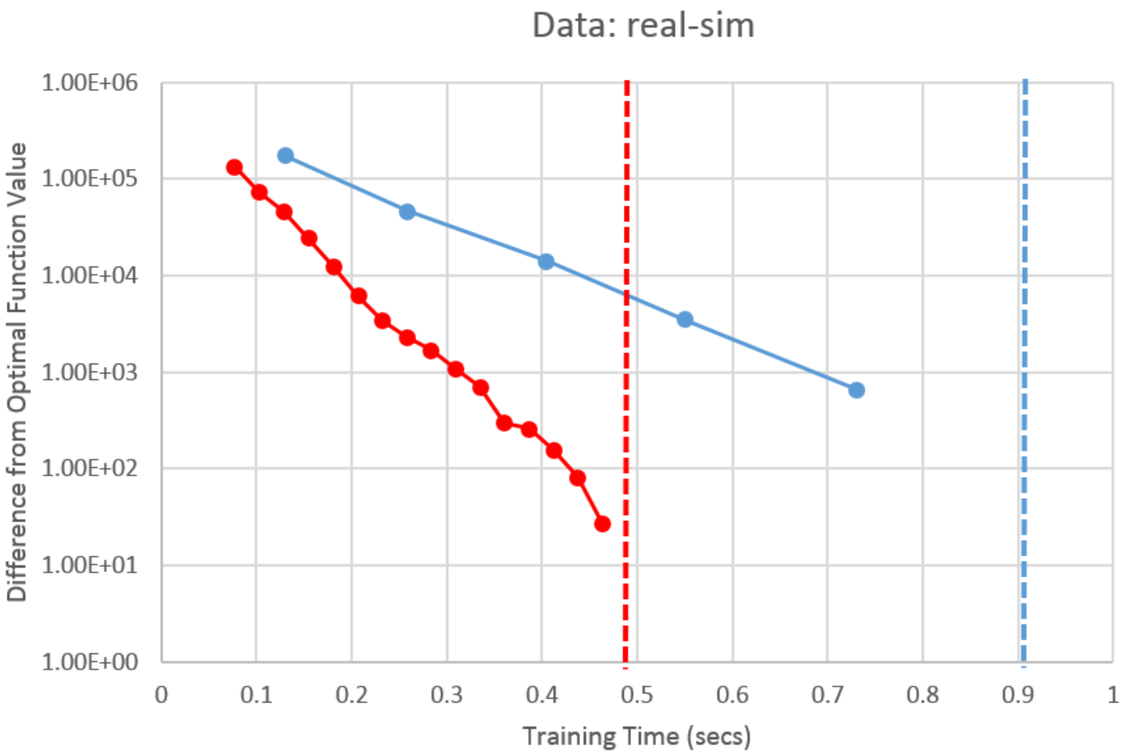
Optimization algorithms are just different ways of solving the same classification problem



Hewlett Packard Enterprise

Some algorithms are faster at solving certain problems (there is no one-size-fits-all when it comes to algorithms and datasets)





TRON iterations

L-BFGS iterations

Time when TRON converged

Time when L-BFGS converged

Hewlett Packard Enterprise

CROlinear is more accurate than a linear classifier, and much faster than a kernel classifier

MNIST 1M (single core)

	Linear classifier only	Kernel classifier only*	CROlinear
Training Time (mins)	8.79	754	31.45
Prediction Time (µs/ instance)	126	1,200	121
Accuracy	86.4%	96.3%	99.2%

*process auto killed

TIMIT 1.4M (100 cores)

	Linear classifier only	Kernel classifier only	CROlinear
Training Time (hrs)	1.9	52	3
Prediction Time (µs/instance)	4	140,000	4
Accuracy	50.7%	74.0%	72.2%

What I accomplished on this internship

- I read scientific papers (lots of them)
- I studied logistic regression in greater depth, since we only covered a little bit of it at the end of Math 261A
- I studied the inner workings of Liblinear, which is an open source software library for large-scale linear classification (Liblinear was the basis for the linear classifier of CROlinear)
- I learned Linux (necessary, since it's hard to compile Liblinear on Windows)
- I learned how to access a remote Linux server for running large jobs
- I wrote Python scripts to run experiments with Liblinear
- I learned a bit of C/C++ in order to make modifications to Liblinear, including:
  - Modifying the Python wrapper for a version of Liblinear that includes the two extensions for incremental training and multicore training
  - Incorporating the L-BFGS algorithm into Liblinear and creating a command-line option to use it
- I learned Git in order to share code with my team on HPE's internal Github repository

For more information on the CRO feature map,

see the paper by Kave Eshghi and Mehran Kafai:

http://alumni.cs.ucr.edu/~mkafai/papers/Paper_icde16.pdf

Hewlett Packard Enterprise

San José State UNIVERSITY

Thank you

I THINK MY NEST SMOKE ALARM IS GOING OFF. GOOGLE ADWORDS JUST PITCHED ME A FIRE EXTINGUISHER AND AN OFFER FOR TEMPORARY HOUSING.
TOM FISH BURNE

© marketoonist.com