
H2O
WORLD
2017

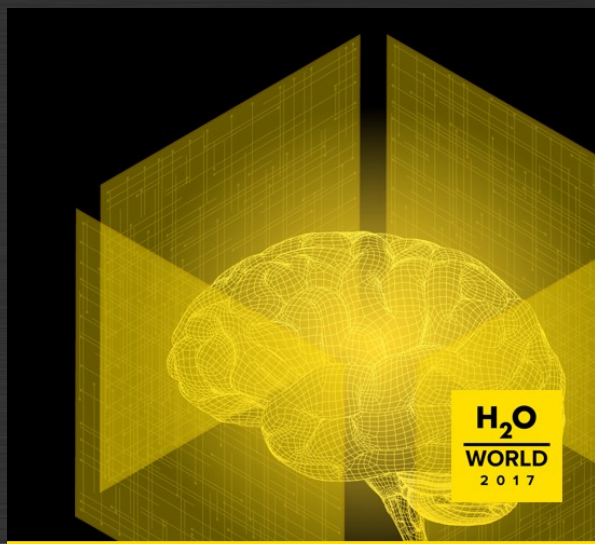
NVIDIA

H2O.ai

World's Fastest Machine Learning With GPUs

http://github.com/h2oai/h2o4gpu

Speaker: Jonathan C. McKinney



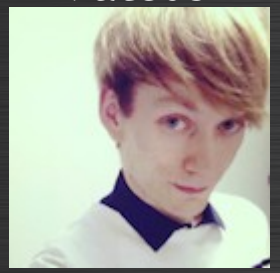
First-time Qwiklab Account Setup

- Just view: http://github.com/h2oai/h2o4gpu
  - Go to: examples/py/demos
- Hands-on: http://h2oai.qwiklab.com
  - Click on “JOIN”
  - Create a new account with a valid email address
  - You will receive a confirmation email
  - Click on the link in the confirmation email
  - Go back to http://h2oai.qwiklab.com and log in
  - Go to the Catalog on the left bar
  - Choose “Introduction to H2O4GPU”
  - Click Start Lab
  - Copy http link on left into browser
  - Enter token “h2o”

NVIDIA

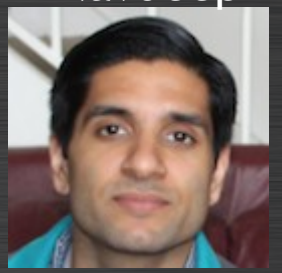
H2O.ai

Mateusz

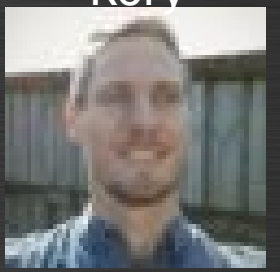


Erin

Navdeep



Rory



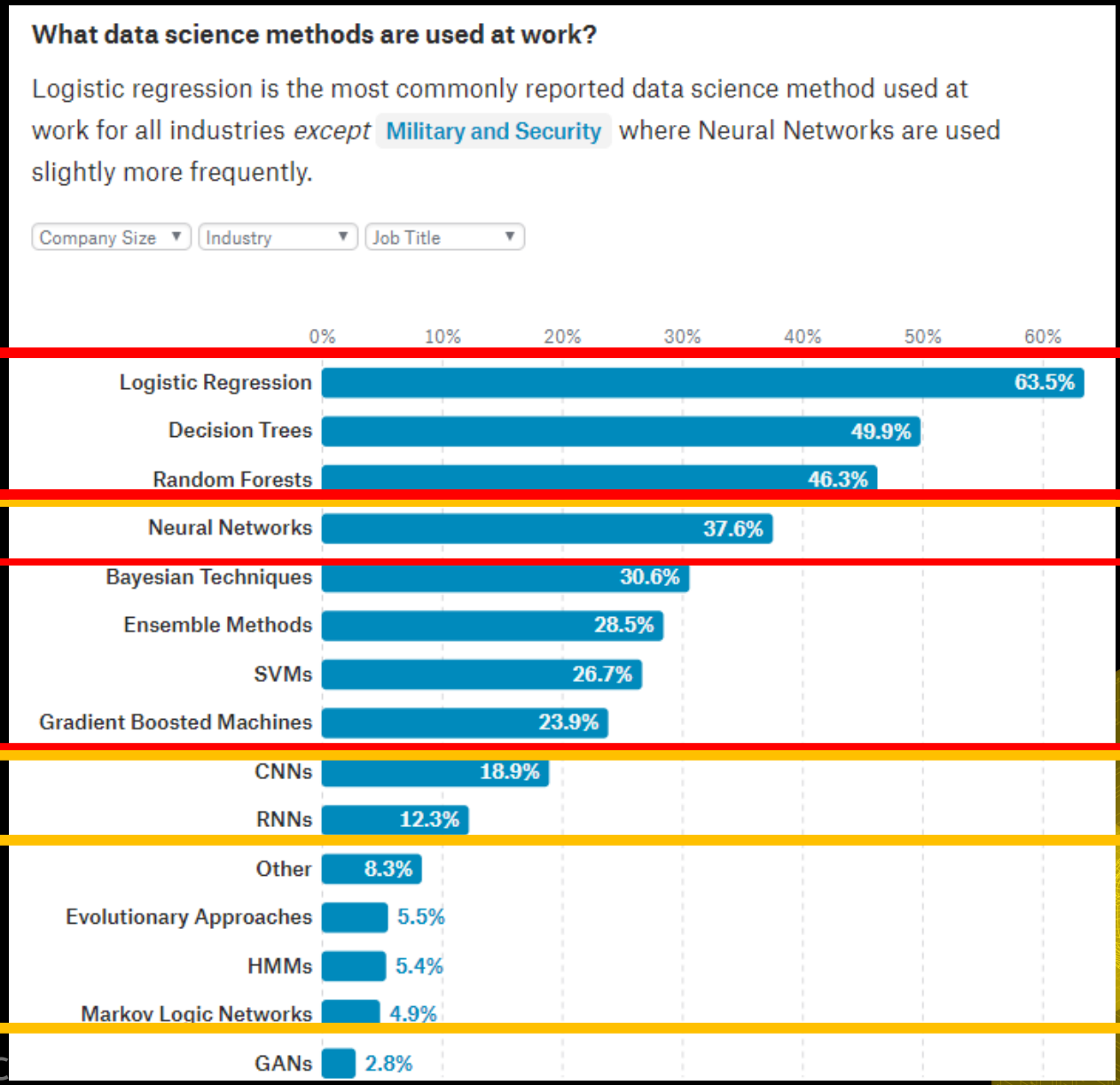
Karen

Arno

Jonathan

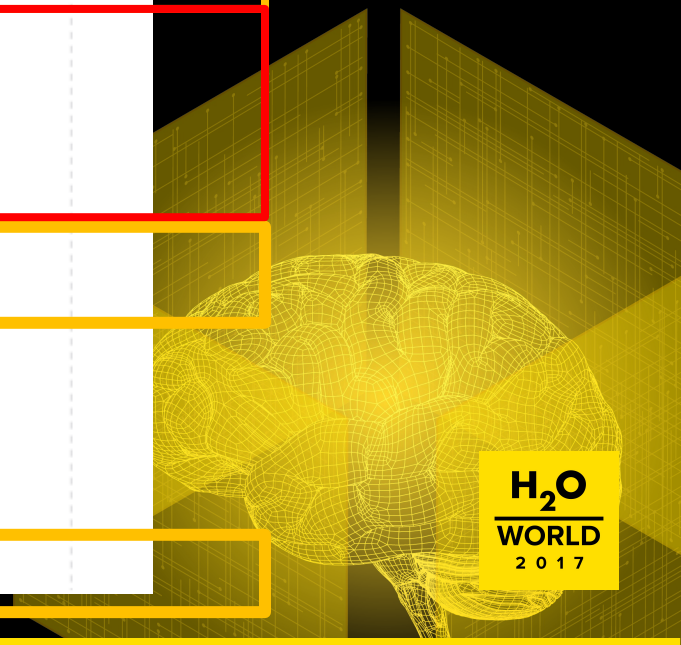
Steve

H2O4GPU TEAM



Deep Learning

Machine Learning

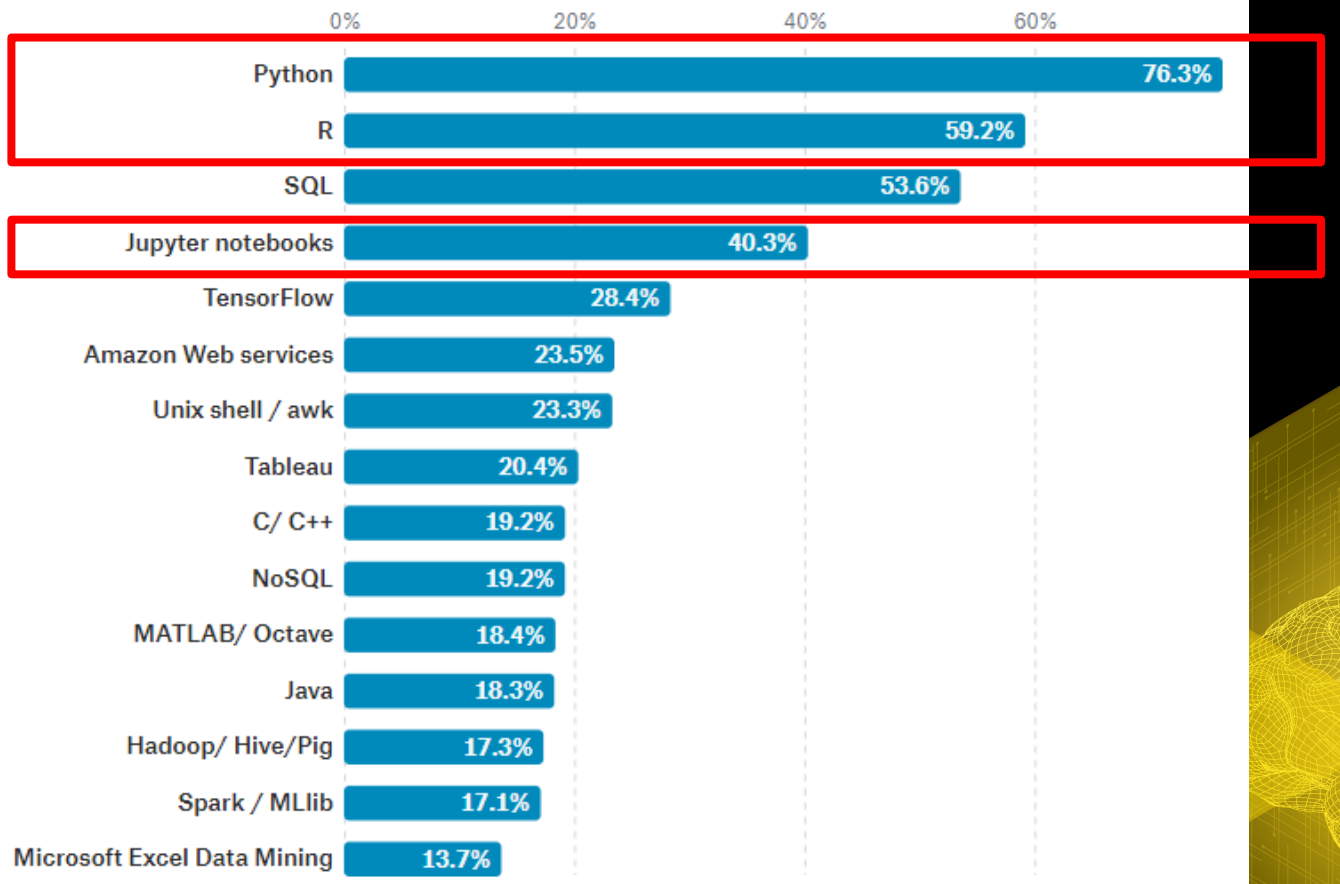


H2O WORLD 2017

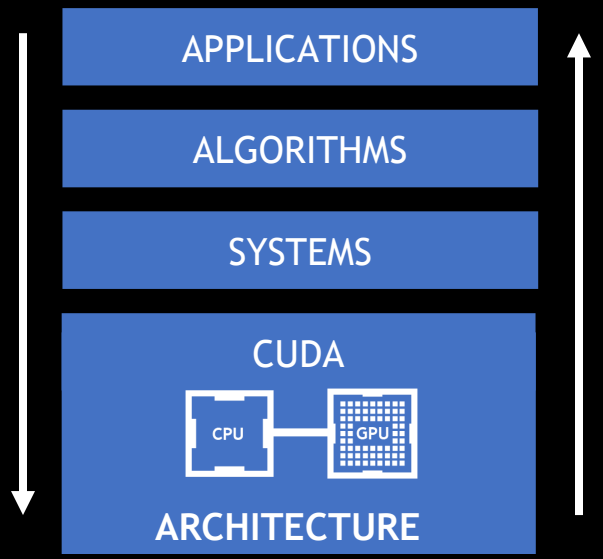
What tools are used at work?

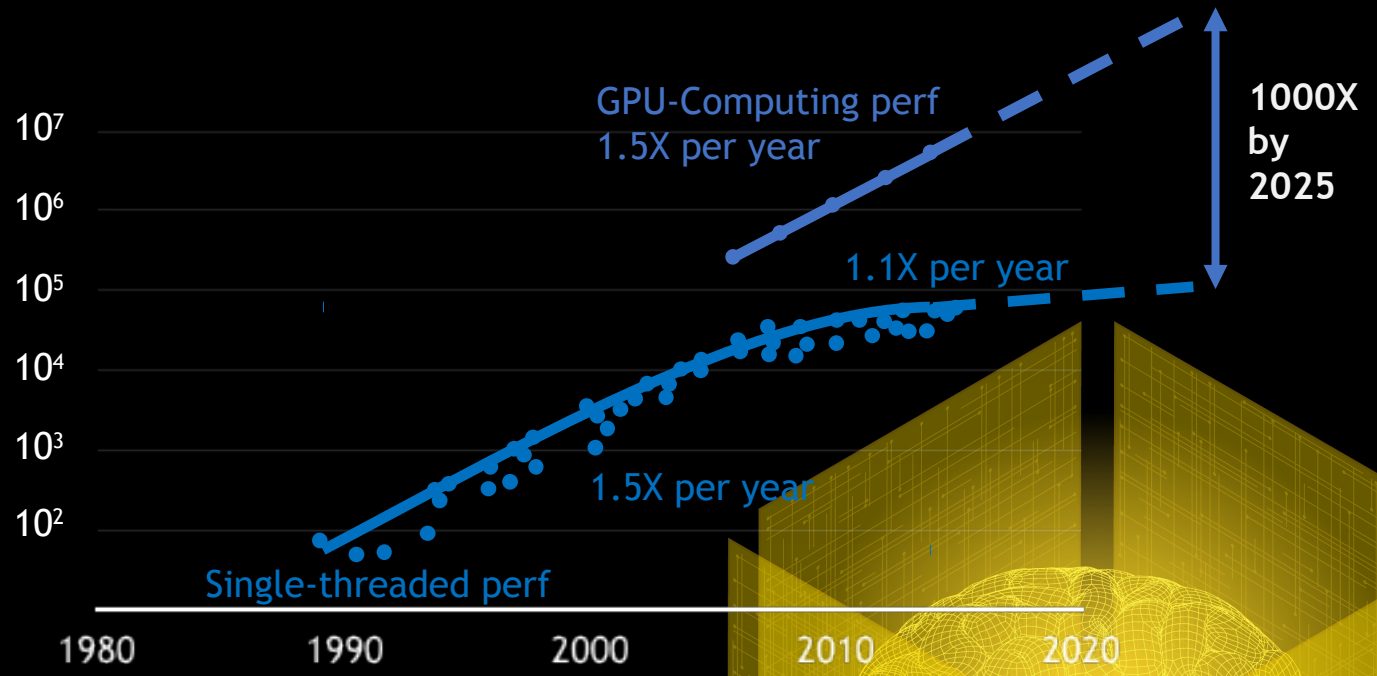
Python was the most commonly used data analysis tool across employed data scientists overall, but more Statisticians are still loyal to R.

Company Size ▼ | Industry ▼ | Job Title ▼



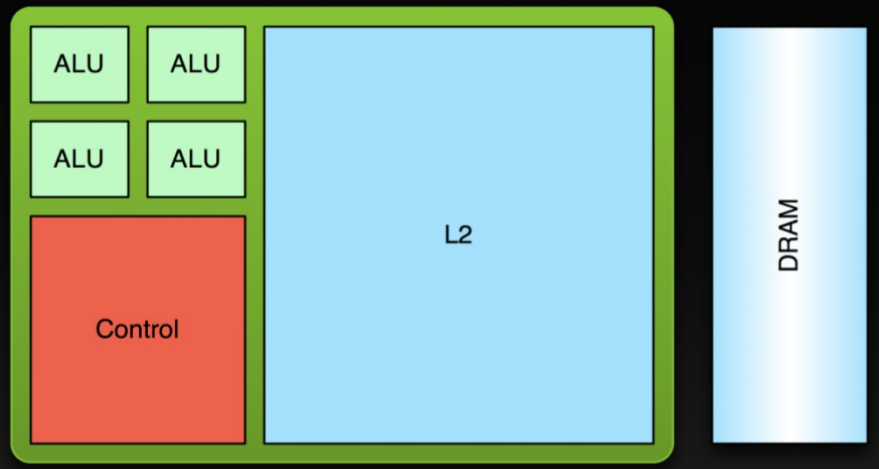
RISE OF GPU COMPUTING





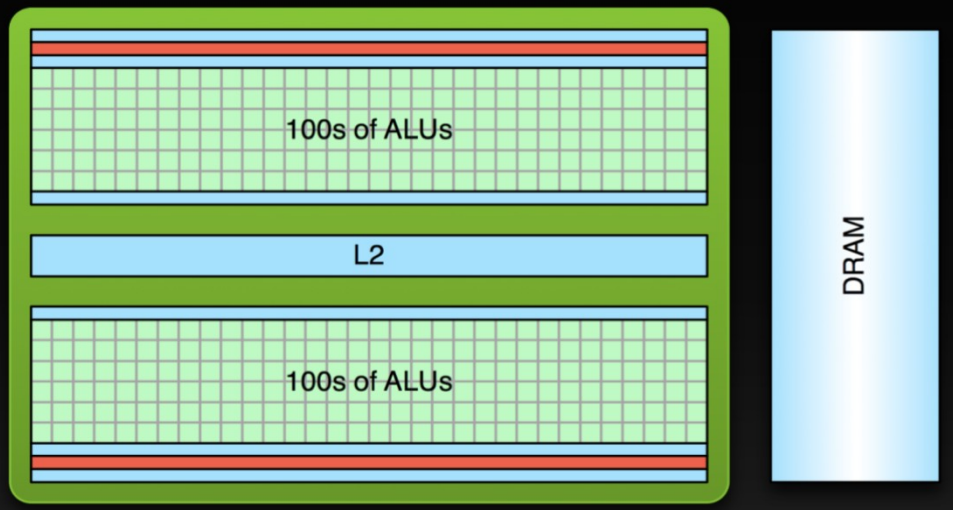
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten New plot and data collected for 2010-2015 by K. Rupp

Low Latency or High Throughput?



CPU

- Optimized for low-latency access to cached data sets
- Control logic for out-of-order and speculative execution



GPU

- Optimized for data-parallel, throughput computation
- Architecture tolerant of memory latency
- More transistors dedicated to computation

H2O4GPU

- Open-Source: **http://github.com/h2oai/h2o4gpu**
- Used within our own **Driverless AI Product** to boost performance **30X**
- Scikit-Learn Python API (and soon R API)
- All Scikit-Learn algorithms included
- Important algorithms ported to GPU

H2O4GPU Roadmap

Currently Available - Q3 (09-30-2017)
GLM (POGS)
Python API for training & scoring
GBM
Inference on GPU (GLM)
Random Forest
Inference on GPU (GBM)
k-Means Clustering

API Support
Python API for training & scoring
Sckit learn API compatibility

Q4 2017 - (12-31-2017)
k-Nearest Neighbors
PCA
SVD
Quantiles
Kalman Filters
Sort
Aggregator

API Support
R API for training & scoring
GOAI API support
Data.table

Performance & Scalability
Fastest single GPU performance
Multi GPU
Multi machine

2018-19
Kernel Methods
Recommendation Engines - Non-Negative Matrix Factorization
Recommendation Engines - Bayesian Neural Nets
MCMC Solver
Time Series
SVM
Text Analysis - TF-IDF
Text Analysis - Word2Vec
Text Analysis - Doc2Vec
Automatic K for K-means
H2O GLM - Lasso
Simulation Techniques
Sampling Techniques

Domain Specific Algorithms
Life Sciences
Financial Services
Underwriting
Sampling Techniques

H2O.ai

H2O.ai

0.0.1

TRAINING DATA

DATASET

default_of_credit_card_clients.csv

ROWS	COLUMNS	DROPPED	IGNORED
24k	25	0	1

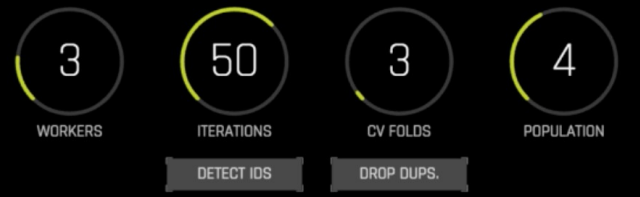
TARGET COLUMN

default payment next month

TYPE	NA	MEAN	STD DEV
int64	0	0.22	0.41

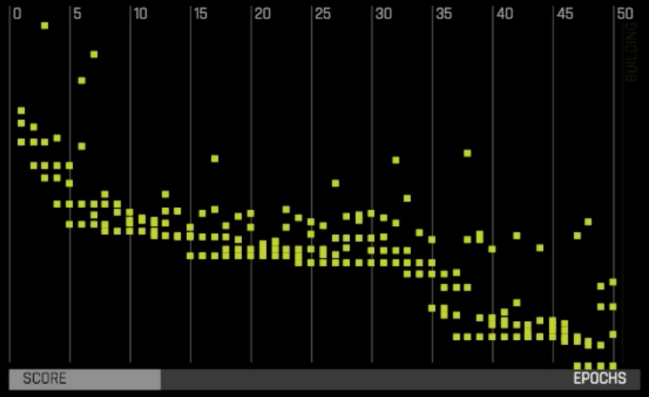
STATUS: RUNNING

EXPERIMENT SETTINGS



GPU STATS

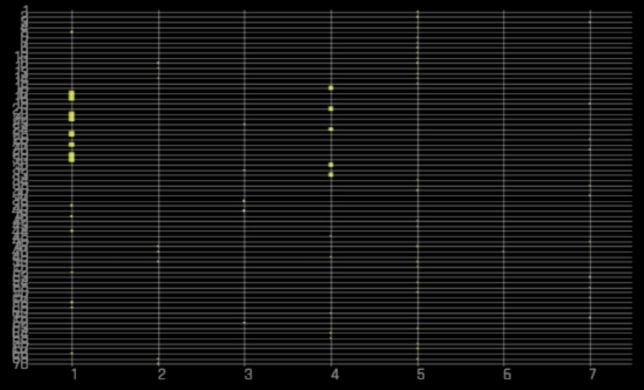
ITERATION SCORES



VARIABLE IMPORTANCE

58_Interaction_PAY_0#multiply#BILL_AMT1	14447.97
10_PAY_0	8213.74
41_Interaction_PAY_0#safe_divide#PAY_AMT5	5121.81
70_PAY_0	3287.98
30_Interaction_BILL_AMT2#safe_divide#LIMIT_BAL	3011.14
66_Interaction_PAY_AMT3#multiply#LIMIT_BAL	2323.32
56_Interaction_BILL_AMT2#multiply#LIMIT_BAL	1567.81
11_PAY_2	1403.90
71_Interaction_PAY_AMT6#subtract#PAY_AMT4	1248.57
12_PAY_3	1227.97
62_CV_CatNumEnc_PAY_5__PAY_AMT5_std	1127.80
56_Interaction_PAY_0#multiply#BILL_AMT2	1101.12
47_Interaction_PAY_4#multiply#LIMIT_BAL	1089.63
32_ClusterID_57	1075.07

FEATURE TRANSFORMATIONS



https://www.youtube.com/watch?v=KkvWX3FD7yI

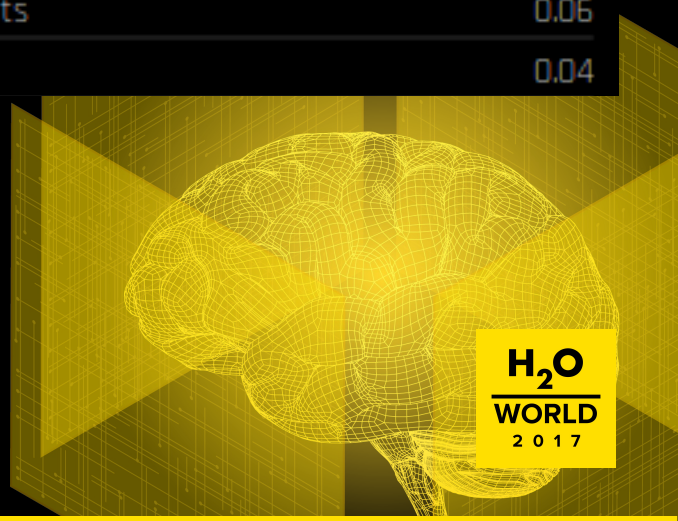
H2O WORLD 2017

VARIABLE IMPORTANCE

36_ClusterDist_8_PAY_0_PAY_2_PAY_AMT2_4	1.00
38_NumCatTE_PAY_0_PAY_2_PAY_6_0	0.52
39_NumCatTE_BILL_AMT2_PAY_0_PAY_5_0	0.28
21_PAY_0	0.28
36_ClusterDist_8_PAY_0_PAY_2_PAY_AMT2_7	0.22
4_CV_TE_PAY_0_0	0.19
12_BILL_AMT1	0.15
2_CV_TE_LIMIT_BAL_0	0.15
19_LIMIT_BAL	0.12
27_PAY_AMT1	0.12
29_PAY_AMT3	0.09
32_PAY_AMT6	0.09
28_PAY_AMT2	0.08
0_CV_TE_AGE_0	0.08

VARIABLE IMPORTANCE

537_Interaction_I_AQB_PrevQ1#subtract#EOP_prev1	1.00
533_TruncSVD_D_prev1_D_prev2_EOP_prev1_1	0.32
540_ClusterDist_6_BAL_prev6_D_prev1_D_prev2_I_AQB_Pre...	0.30
136_CV_TE_FINAL_WORTH_prev1_0	0.16
374_CR_AMB_Drop_Build_1	0.15
537_Interaction_D_prev1#subtract#EOP_prev1	0.11
407_EOP_prev1	0.10
537_Interaction_CR_AMB_Drop_Build_1#subtract#EOP_pr...	0.08
540_ClusterDist_6_BAL_prev6_D_prev1_D_prev2_I_AQB_Pre...	0.07
462_Percent_Change_in_Credits	0.06
278_BAL_prev6	0.04



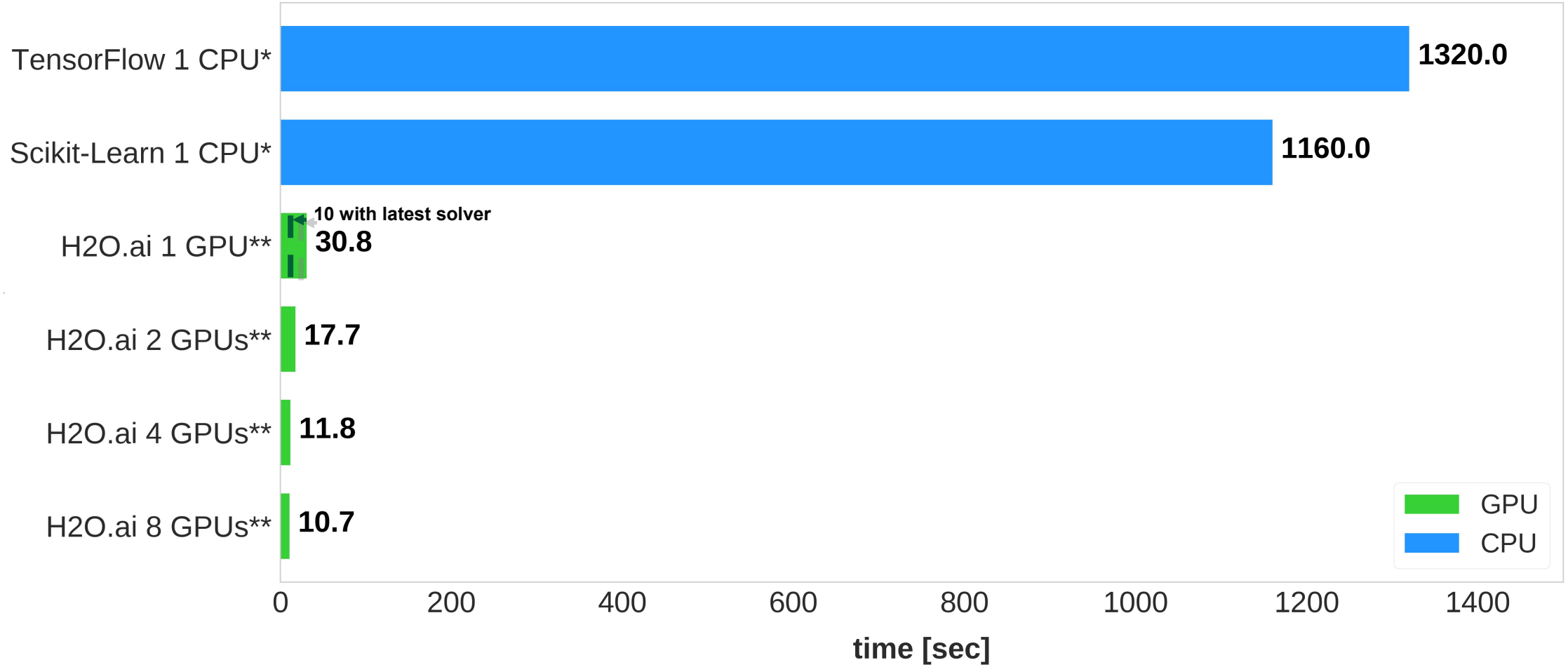
Model Accuracy & Speed

K-Means on H2O4GPU

- Based upon NVIDIA prototype of K-Means algorithm in CUDA
- Improvements to original implementation:
  - Significantly faster than scikit-learn implementation (50x)
  - Significantly faster than other GPU implementations (5x-10x)
  - Supports kmeans++/kmeans|| initialization
  - Supports multiple GPUs
  - Supports batching data if exceeds GPU memory

H2O.ai Machine Learning – k-Means Clustering

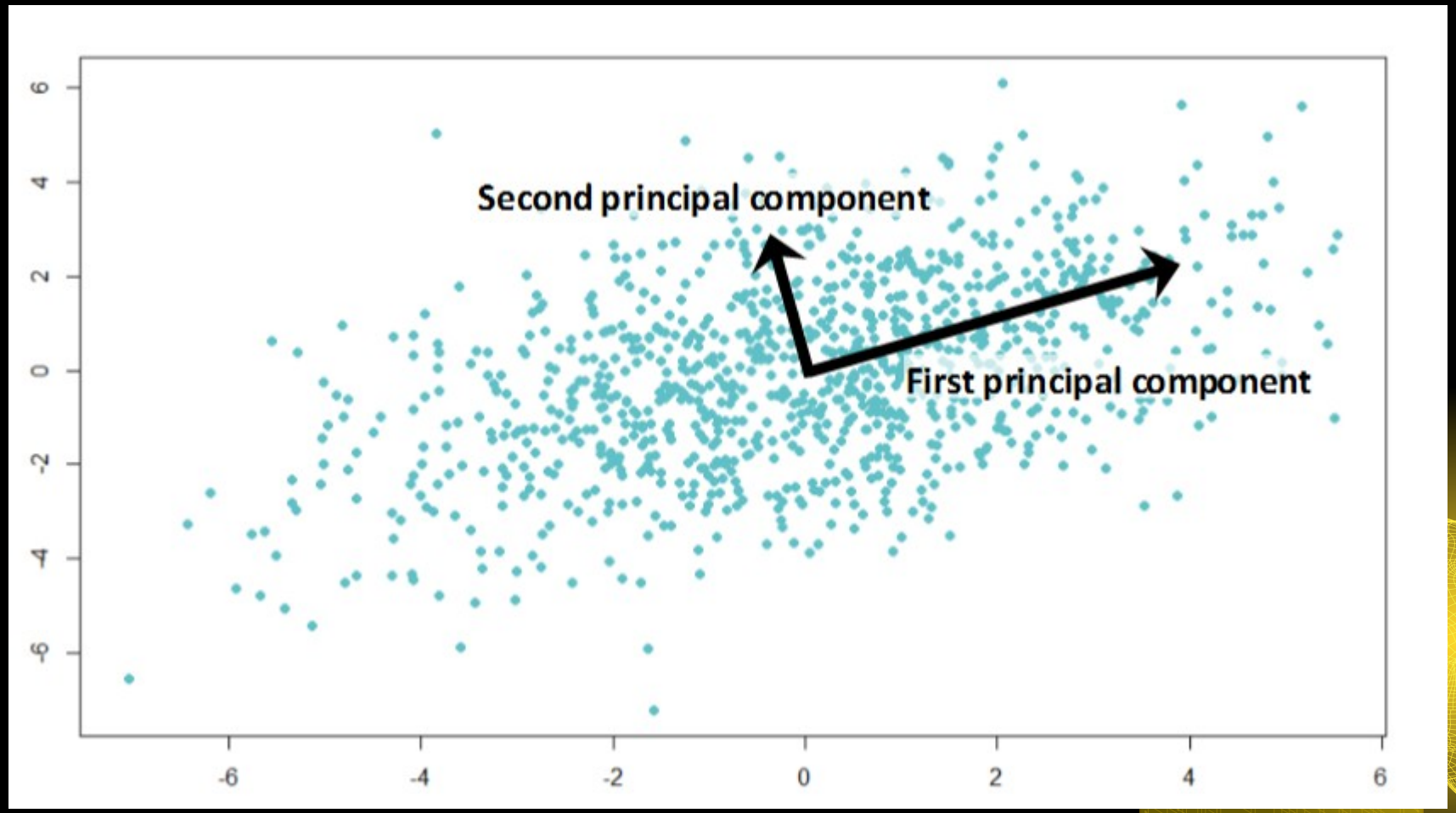
Time to run 1000 Lloyds iterations for k=1000 clusters

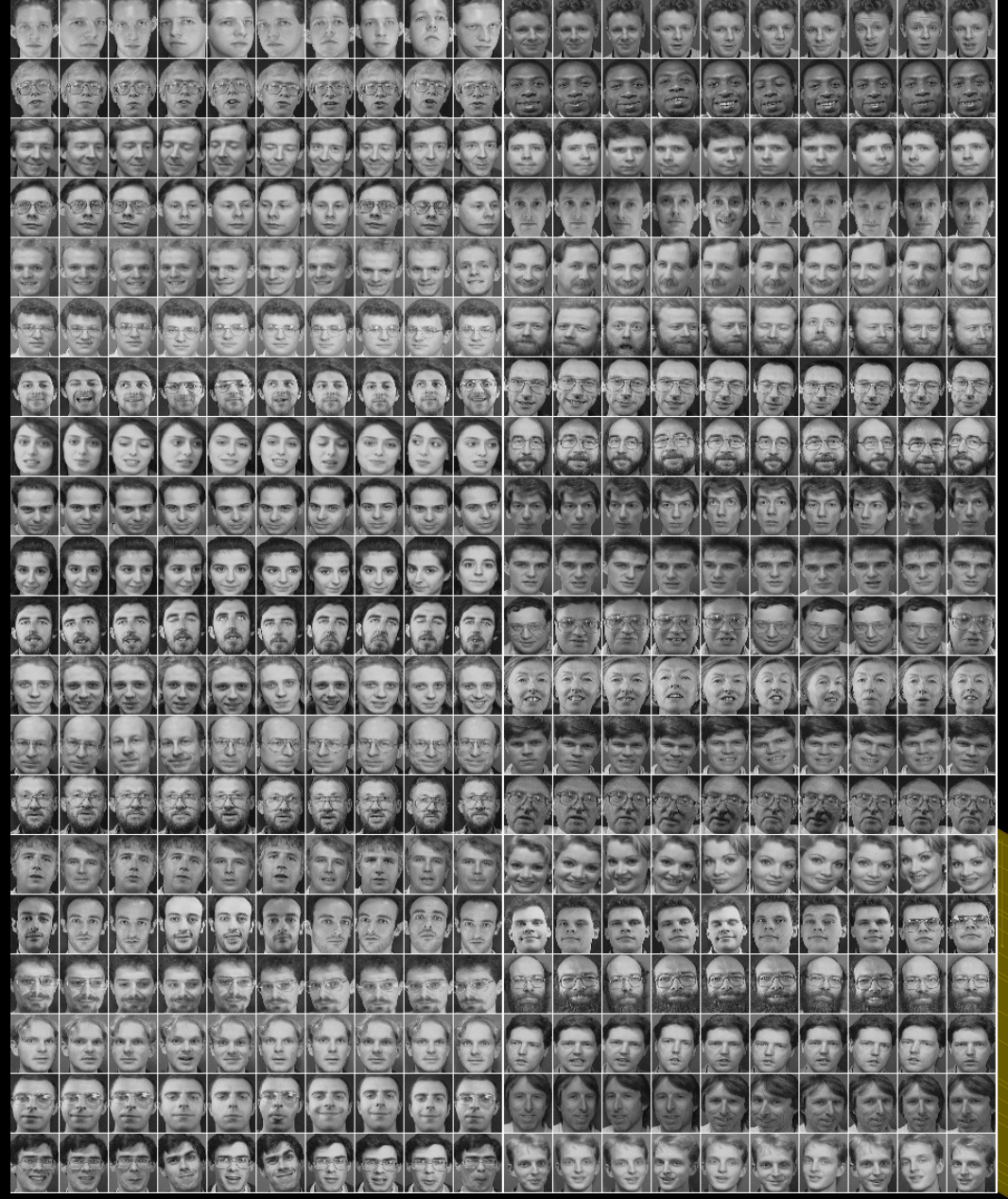


Kaggle Homesite Home Insurance Claims Predictions Dataset (261k rows, 298 cols)
k-Means Clustering (Lloyds), random initialization, 1000 centroids, 1000 iterations
Hardware: *Intel i7 5820k (6-core), **NVIDIA Tesla P100 (DGX-1)

http://github.com/h2oai/perf/

Principle Component Analysis (PCA)

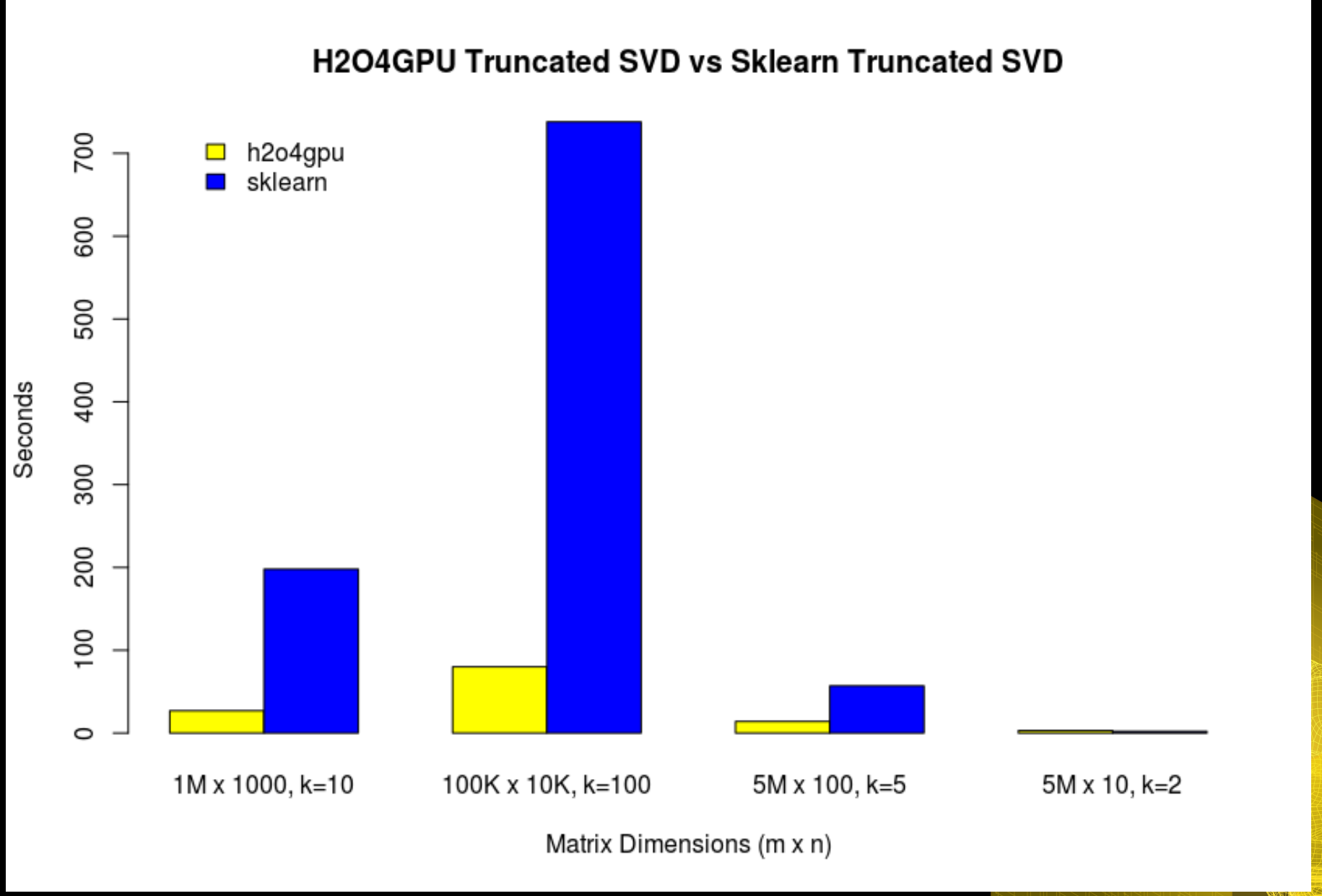





H2O
WORLD
2017

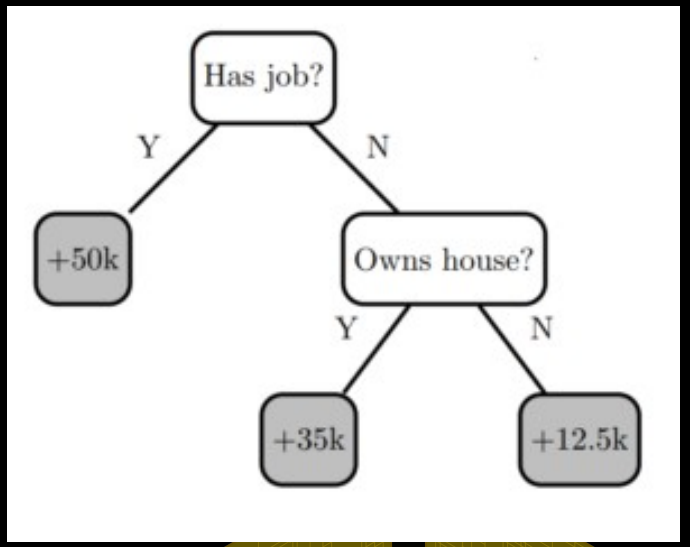
First centered Olivetti faces

Generate faces from PCA



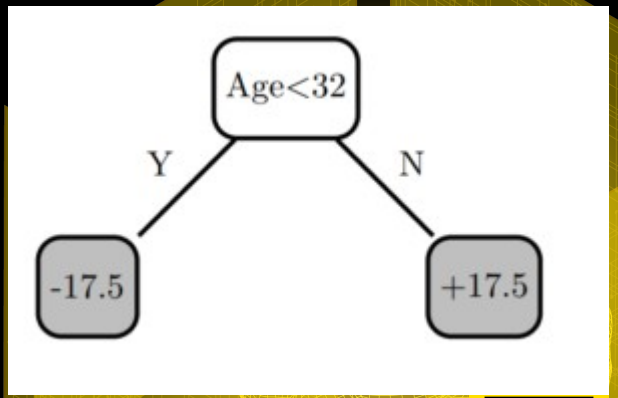
Gradient Boosting Machines in H2O4GPU

- Based upon XGBoost



- Raw floating point data -> Binned into Quantiles
- Quantiles are stored as compressed instead of floats

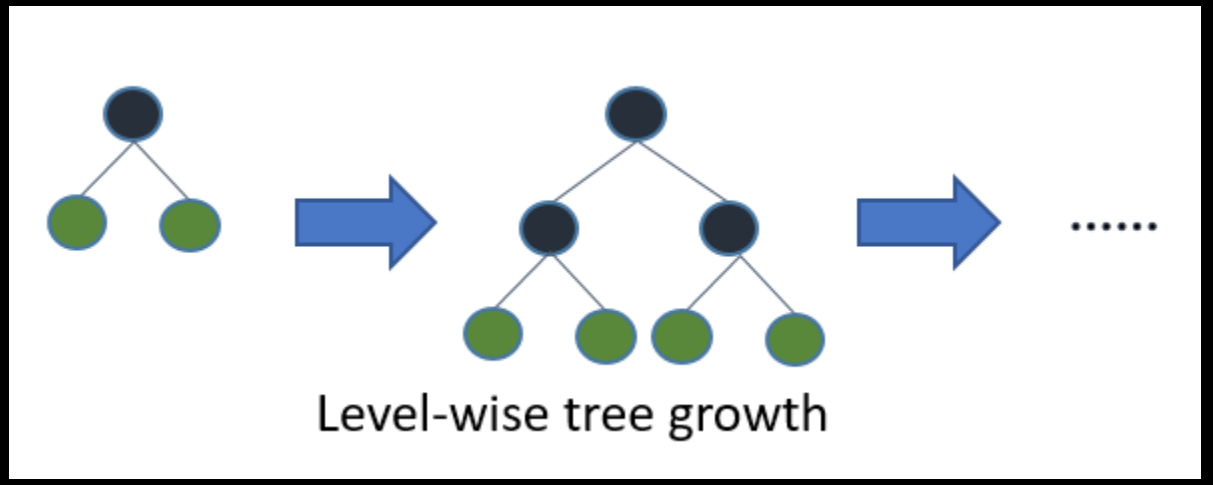
- Compressed Quantiles are efficiently transferred to GPU
- Sparsity is handled directly with highly GPU efficiency

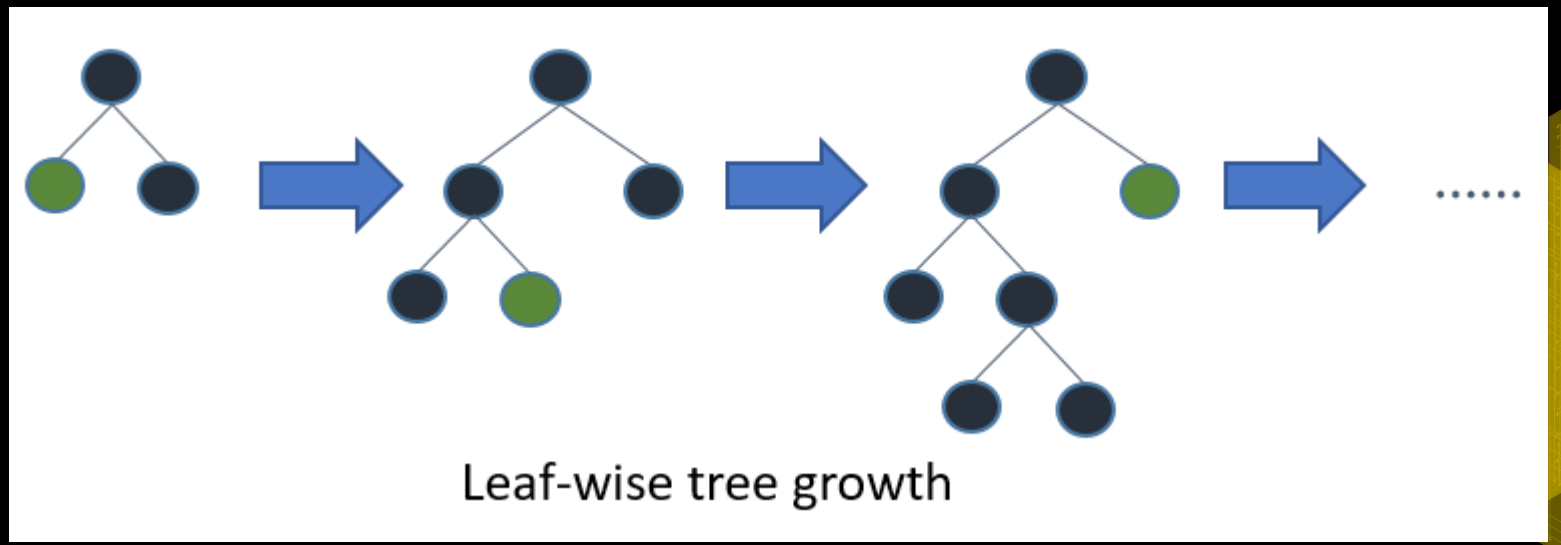


- Multi-GPU by sharding rows using NVIDIA NCCL AllReduce

https://github.com/h2oai/h2o4gpu/blob/master/examples/py/xgboost_simple_demo.ipynb

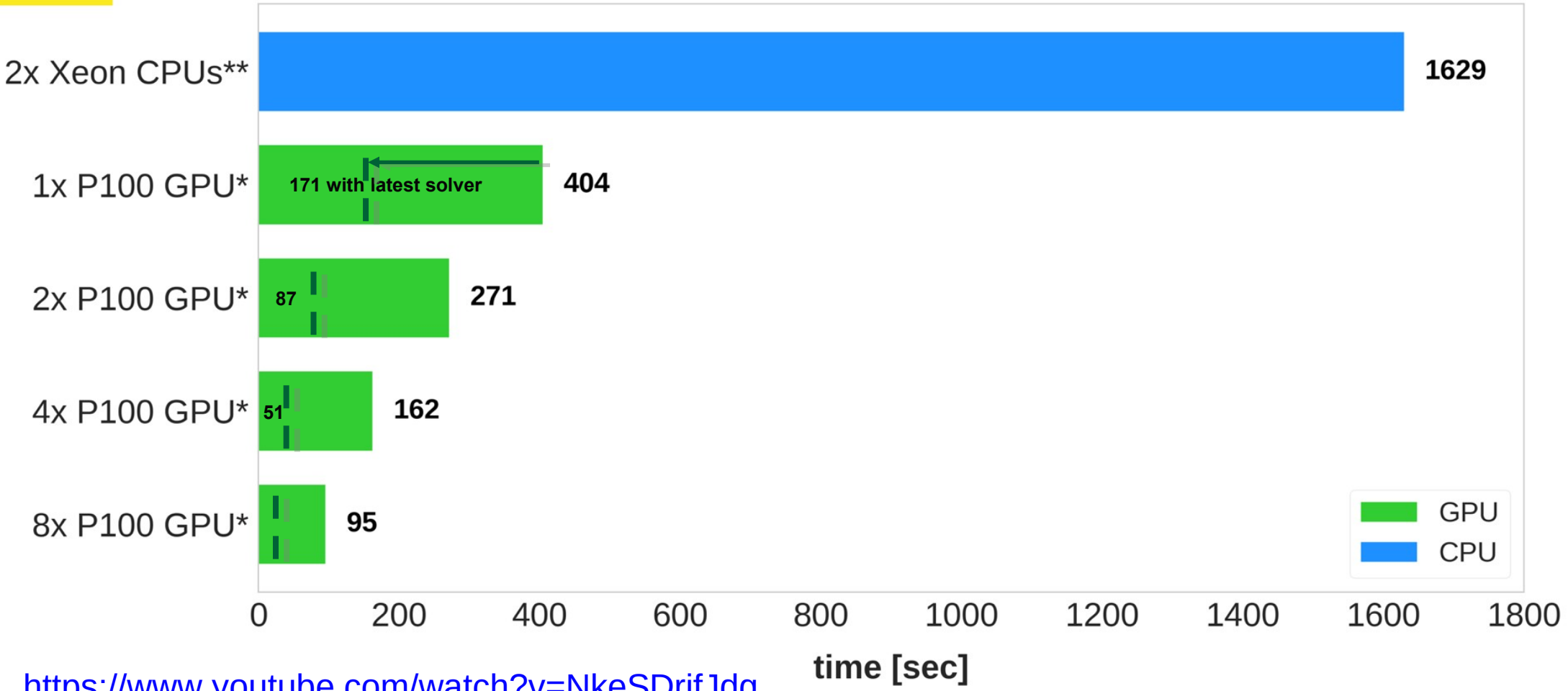
Tree Growth Algorithms





H2O.ai Machine Learning – Gradient Boosting Machine

Time to Train 16 H2O XGBoost Models

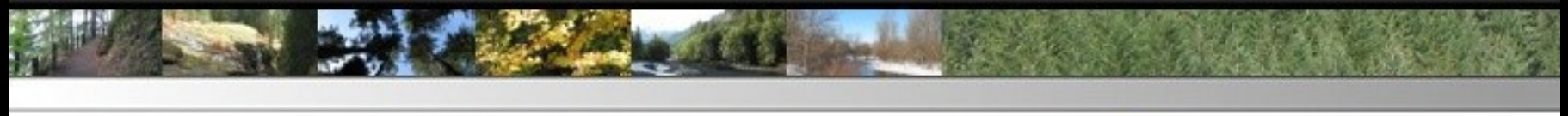


https://www.youtube.com/watch?v=NkeSDrifJdg

*NVIDIA DGX-1, **Dual Intel Xeon E5-2630 v4

http://github.com/h2oai/perf/ Higgs dataset (binary classification): 1M rows, 29 cols; max_depth: {6,8,10,12}, sample_rate: {0.7,0.8,0.9,1.0}

Example Forest - Covertype



H2O4GPU

http://github.com/h2oai/h2o4gpu
https://stackoverflow.com/questions/tagged/h2o4gpu
https://gitter.im/h2oai/h2o4gpu

Thank You!
Questions?