Scalable Machine Learning with H2O

Boston Data Education Meetup

May 2016

Erin LeDell Ph.D.
Machine Learning Scientist
H2O.ai

Introduction

- Statistician & Machine Learning Scientist at H2O.ai in Mountain View, California, USA
- Ph.D. in Biostatistics with Designated Emphasis in Computational Science and Engineering from UC Berkeley (focus on Machine Learning)
- Worked as a data scientist at several startups

Berkeley
UNIVERSITY OF CALIFORNIA

H2O.ai

H2O

Agenda

- Who/What is H2O?
- H2O Machine Learning Platform
- H2O in R & Python
- H2O Code Tutorial

H2O.ai

H2O.ai, the Company

- Team: 55; Founded in 2012
- Mountain View, CA
- Stanford & Purdue Math & Systems Engineers

H2O, the Platform

- Open Source Software (Apache 2.0 Licensed)
- R, Python, Scala, Java and Web Interfaces
- Distributed Algorithms that Scale to Big Data

H2O

Scientific Advisory Council

Dr. Trevor Hastie

- John A. Overdeck Professor of Mathematics, Stanford University
- PhD in Statistics, Stanford University
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Co-author with John Chambers, *Statistical Models in S*
- Co-author, *Generalized Additive Models*

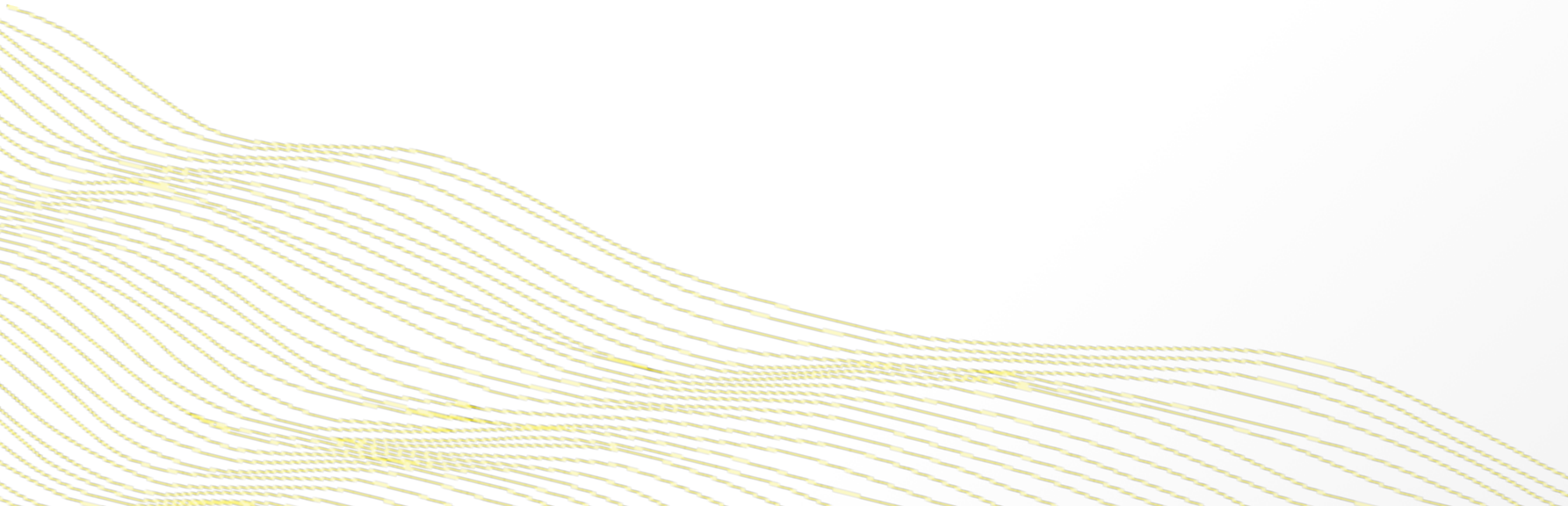
Dr. Robert Tibshirani

- Professor of Statistics and Health Research and Policy, Stanford University
- PhD in Statistics, Stanford University
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Author, *Regression Shrinkage and Selection via the Lasso*
- Co-author, *An Introduction to the Bootstrap*

Dr. Steven Boyd

- Professor of Electrical Engineering and Computer Science, Stanford University
- PhD in Electrical Engineering and Computer Science, UC Berkeley
- Co-author, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*
- Co-author, *Linear Matrix Inequalities in System and Control Theory*
- Co-author, *Convex Optimization*

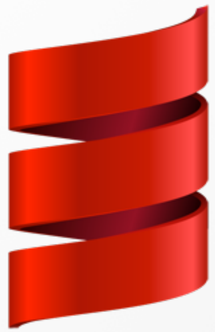
H2O Platform



H2O Platform Overview

- Distributed implementations of cutting edge ML algorithms.
- Core algorithms written in high performance Java.
- APIs available in R, Python, Scala, REST/JSON.
- Interactive Web GUI.




H2O Platform Overview

- Write code in high-level language like R (or use the web GUI) and output production-ready models in Java.
- To scale, just add nodes to your H2O cluster.
- Works with Hadoop, Spark and your laptop.

H2O Distributed Computing

H2O Cluster

- Multi-node cluster with shared memory model.
- All computations in memory.
- Each node sees only some rows of the data.
- No limit on cluster size.

Distributed K/V Store

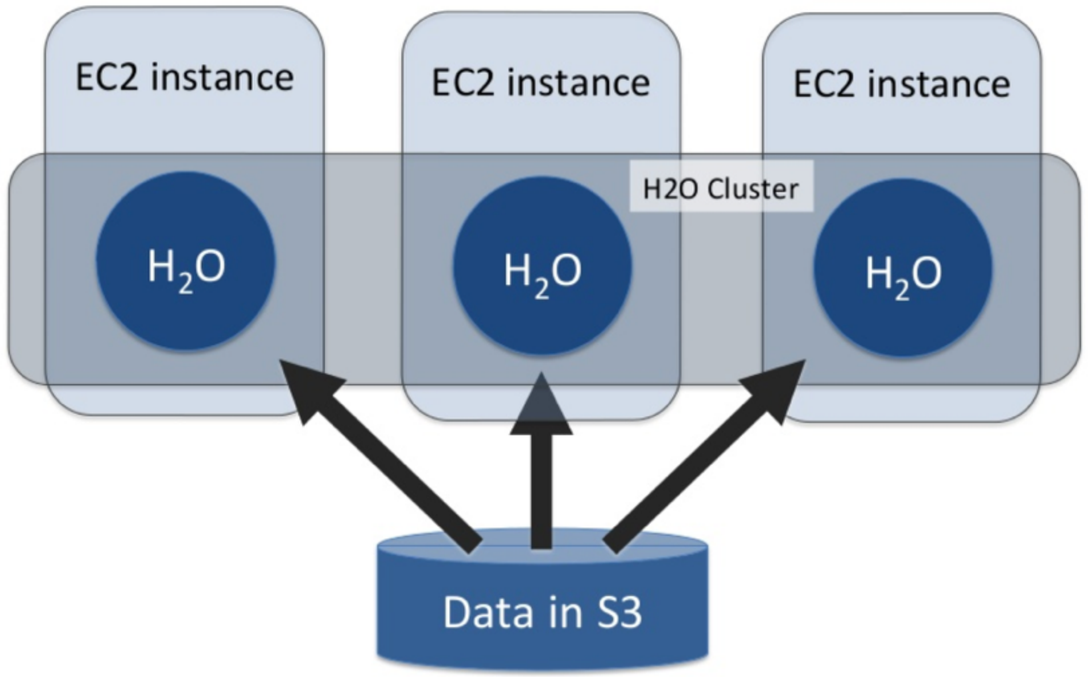
- Objects in the H2O cluster such as data frames, models and results are all referenced by key.
- Any node in the cluster can access any object in the cluster by key.

H2O Frame

- Distributed data frames (collection of vectors).
- Columns are distributed (across nodes) arrays.
- Works just like R's data.frame or Python Pandas DataFrame

H2O

H2O on Amazon EC2



H2O can easily be deployed on an Amazon EC2 cluster.

The H2O GitHub repository contains example scripts that help to automate the cluster deployment ("ec2" folder).

H2O in R and Python

h2o R Package

Installation

- Java 7 or later; R 3.1 and above; Linux, Mac, Windows
- The easiest way to install the h2o R package is CRAN.
- Latest version: http://www.h2o.ai/download/h2o/r

Design

All computations are performed in highly optimized Java code in the H2O cluster, initiated by REST calls from R.

h2o Python Module

Installation

- Java 7 or later; Python 2.7, 3.5; Linux, Mac, Windows
- The easiest way to install the h2o Python module is PyPI.
- Latest version: http://www.h2o.ai/download/h2o/py

Design

All computations are performed in highly optimized Java code in the H2O cluster, initiated by REST calls from R.

H2O R & Python Tutorial

http://tinyurl.com/h2o-boston-data

Tutorial: Intro to H2O Algorithms

The "Intro to H2O" tutorial introduces five popular supervised machine learning algorithms in the context of a binary classification problem.

The training module demonstrates how to train models and evaluating model performance on a test set.

- Generalized Linear Model (GLM)
- Random Forest (RF)
- Gradient Boosting Machine (GBM)
- Deep Learning (DL)
- Naive Bayes (NB)

Tutorial: Grid Search for Model Selection

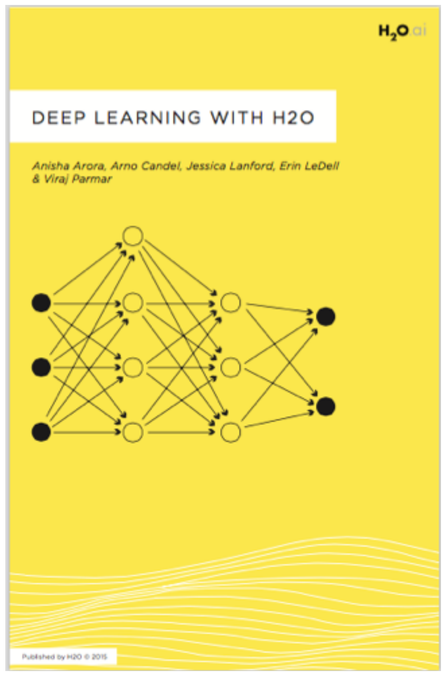
```
> print(gbm_gridperf)
H2O Grid Details
================

Grid ID: gbm_grid2
Used hyper parameters:
  -  sample_rate
  -  max_depth
  -  learn_rate
  -  col_sample_rate
Number of models: 72
Number of failed models: 0

Hyper-Parameter Search Summary: ordered by decreasing auc
  sample_rate max_depth learn_rate col_sample_rate          model_ids                auc
1           1         3       0.19                1 gbm_grid2_model_38 0.685166598389755
2         0.9         3       0.15                1 gbm_grid2_model_53 0.684956999713052
3         0.8         5       0.06                1 gbm_grid2_model_22 0.684843506375254
4         0.6         4       0.07                1  gbm_grid2_model_4 0.684327718715252
5        0.95         4       0.13                1 gbm_grid2_model_48 0.684042497773235
```

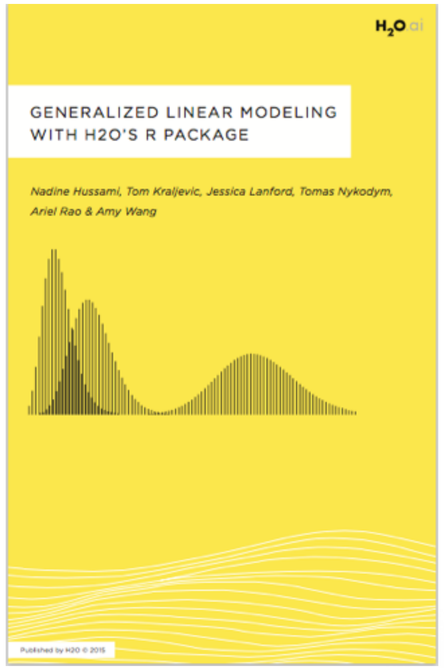
The second training module demonstrates how to find the best set of model parameters for each model using Grid Search.

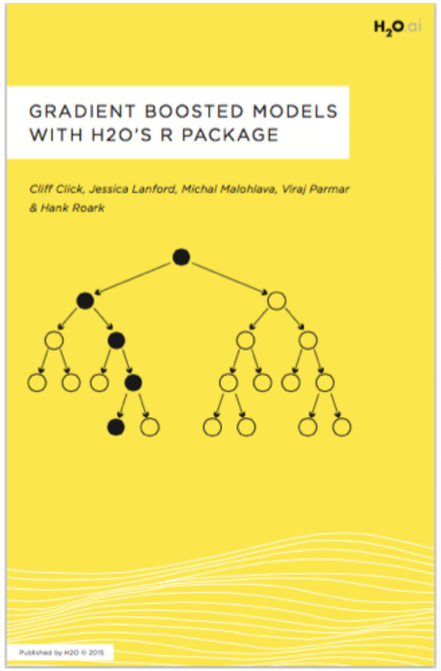
Where to learn more?

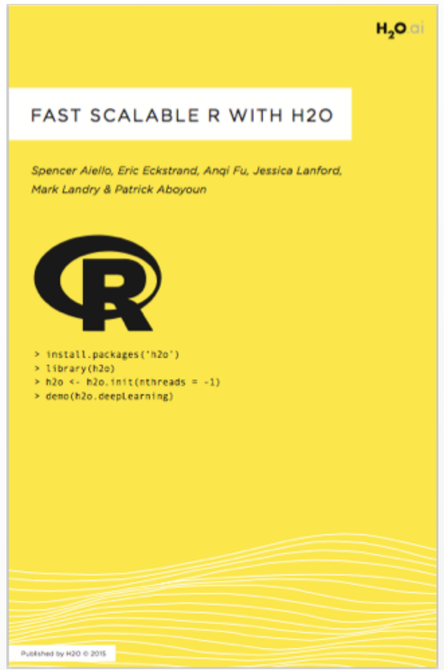
- H2O Online Training (free): http://learn.h2o.ai
- H2O Slidedecks: http://www.slideshare.net/0xdata
- H2O Video Presentations: https://www.youtube.com/user/0xdata
- H2O Community Events & Meetups: http://h2o.ai/events
- Machine Learning & Data Science courses: http://coursebuffet.com

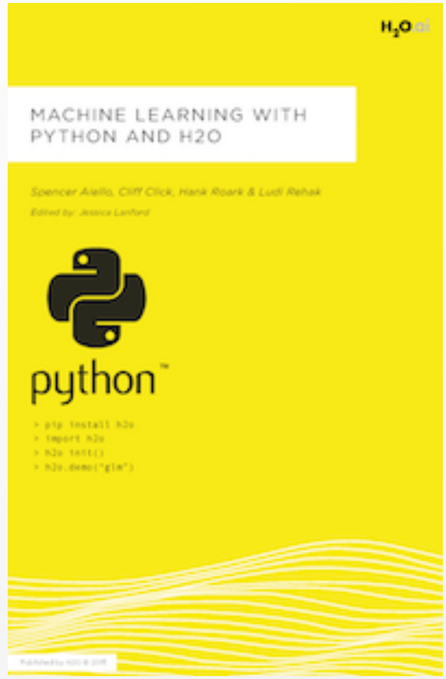
H2O Booklets











http://tinyurl.com/h2o-github-booklets

Thank you!

@ledell on Github, Twitter

erin@h2o.ai

http://www.stat.berkeley.edu/~ledell