Improving Model Performance with Ensemble Machine Learning

D-Lab @ Berkeley

April 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

Ensemble Learning

In statistics and machine learning, ensemble methods use multiple learning algorithms to obtain better predictive performance than could be obtained by any of the constituent algorithms.

– Wikipedia (2016)

Common Types of Ensemble Methods

Bagging	• Reduces variance and increases accuracy • Robust against outliers or noisy data • Often used with Decision Trees (i.e. Random Forest)
Boosting	• Also reduces variance and increases accuracy • Not robust against outliers or noisy data • Flexible – can be used with any loss function
Stacking / Super Learning	• Used to ensemble a diverse group of strong learners • Involves training a second-level machine learning algorithm called a "metalearner" to learn the optimal combination of the base learners

History of Stacking

Stacked Generalization

- David H. Wolpert, "Stacked Generalization" (1992)
- First formulation of stacking via a metalearner
- Blended Neural Networks

Stacked Regressions

- Leo Breiman, "Stacked Regressions" (1996)
- Modified algorithm to use CV to generate level-one data
- Blended Neural Networks and GLMs (separately)

Super Learning

- Mark van der Laan et al., "Super Learner" (2007)
- Provided the theory to prove that the Super Learner is the asymptotically optimal combination
- First R implementation in 2010

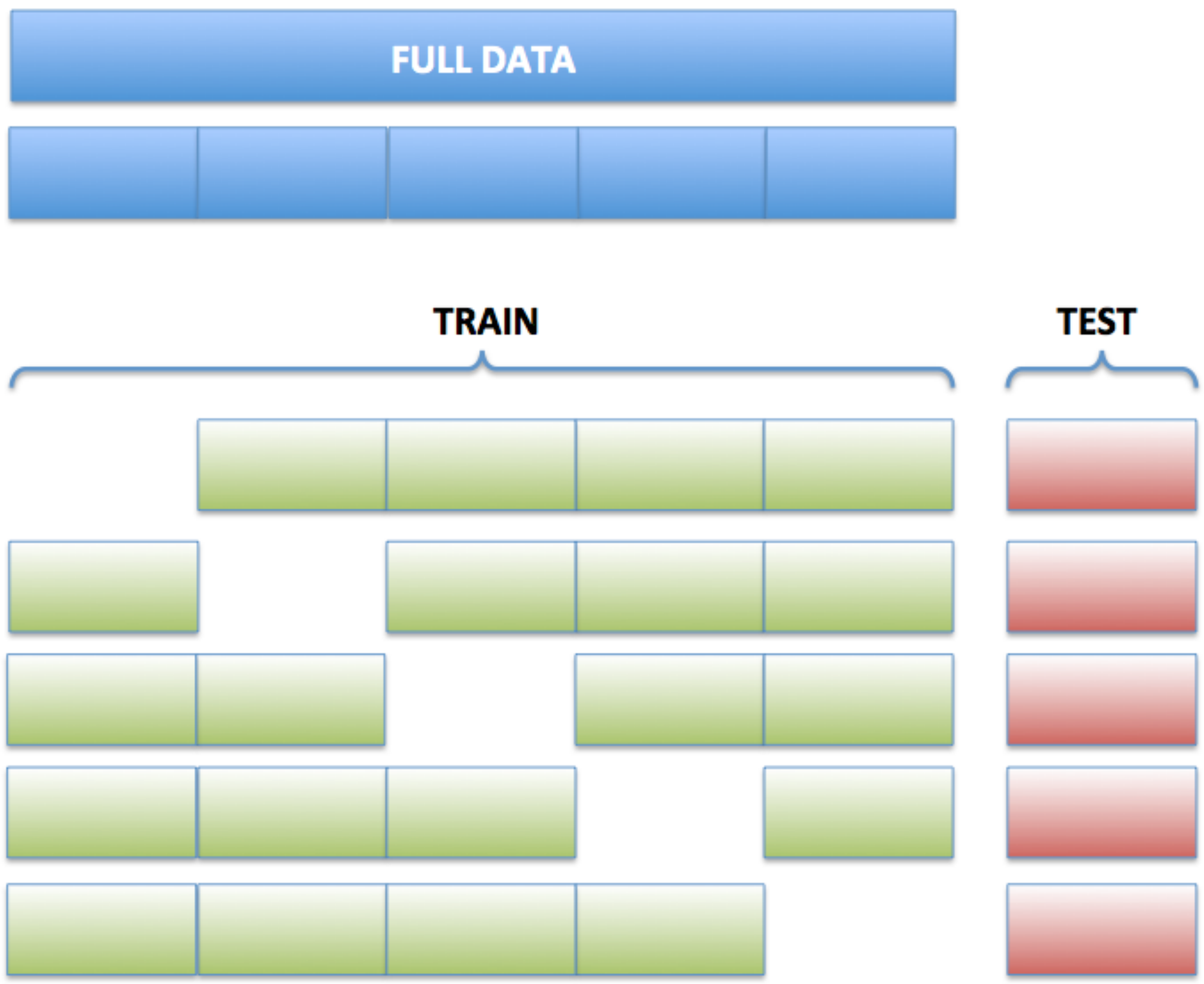
The Super Learner Algorithm

$$n\left\{\overbrace{\begin{bmatrix} & & \\ & X & \\ & & \end{bmatrix}}^{m}\right. \quad \begin{bmatrix} \\ y \\ \\ \end{bmatrix}$$

"Level-zero" data

- Start with design matrix, X, and response, y
- Specify L base learners (with model params)
- Specify a metalearner (just another algorithm)
- Perform k-fold CV on each of the L learners

k-fold Cross-validation



The Super Learner Algorithm

$$n\left\{\begin{bmatrix} \\ p_1 \\ \\ \end{bmatrix} \cdots \begin{bmatrix} \\ p_L \\ \\ \end{bmatrix} \begin{bmatrix} \\ y \\ \\ \end{bmatrix} \right. \rightarrow n\left\{\overbrace{\begin{bmatrix} & & \\ & Z & \\ & & \end{bmatrix}}^{L} \begin{bmatrix} \\ y \\ \\ \end{bmatrix}\right.$$

"Level-one" data

- Collect the predicted values from k-fold CV that was performed on each of the L base learners
- Column-bind these prediction vectors together to form a new design matrix, Z
- Train the metalearner using Z, y

Super Learning vs. Parameter Tuning/Search

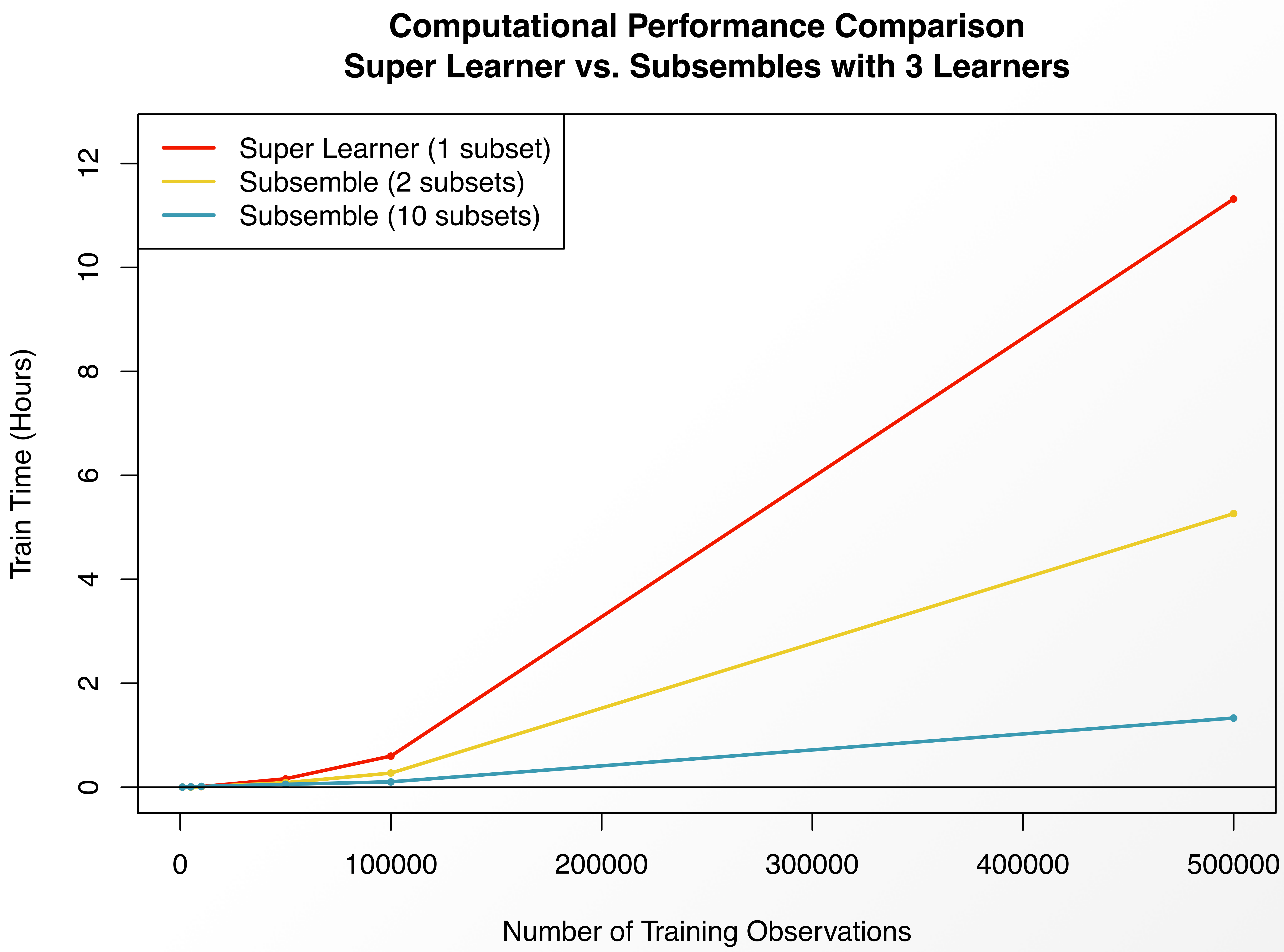
- A common task in machine learning is to perform model selection by specifying a number of models with different parameters.
- An example of this is Grid Search or Random Search.
- The first phase of the Super Learner algorithm is computationally equivalent to performing model selection via cross-validation.
- The latter phase of the Super Learner algorithm (the metalearning step) is just training another single model (no CV).
- With Super Learner, your computation does not go to waste!

The Subsemble Algorithm

"Subsemble: An ensemble method for combining subset-specific algorithm fits."
Stephanie Sapp, Mark J. van der Laan, John Canny. *Journal of Applied Statistics* (2013)

- Start with design matrix, X, and response, y
- Specify L base learners (with model params)
- Specify a metalearner (just another algorithm)
- Partition the training set into J subsets
- Perform a modified version of k-fold CV to get level-one data
- Resulting ensemble contains J x L models

The Subsemble Algorithm



Super Learner R Software Overview

SuperLearner

- Original Super Learner R implementation (2010).
- Comes with support for many existing machine learning R packages and can be customized to wrap any other.

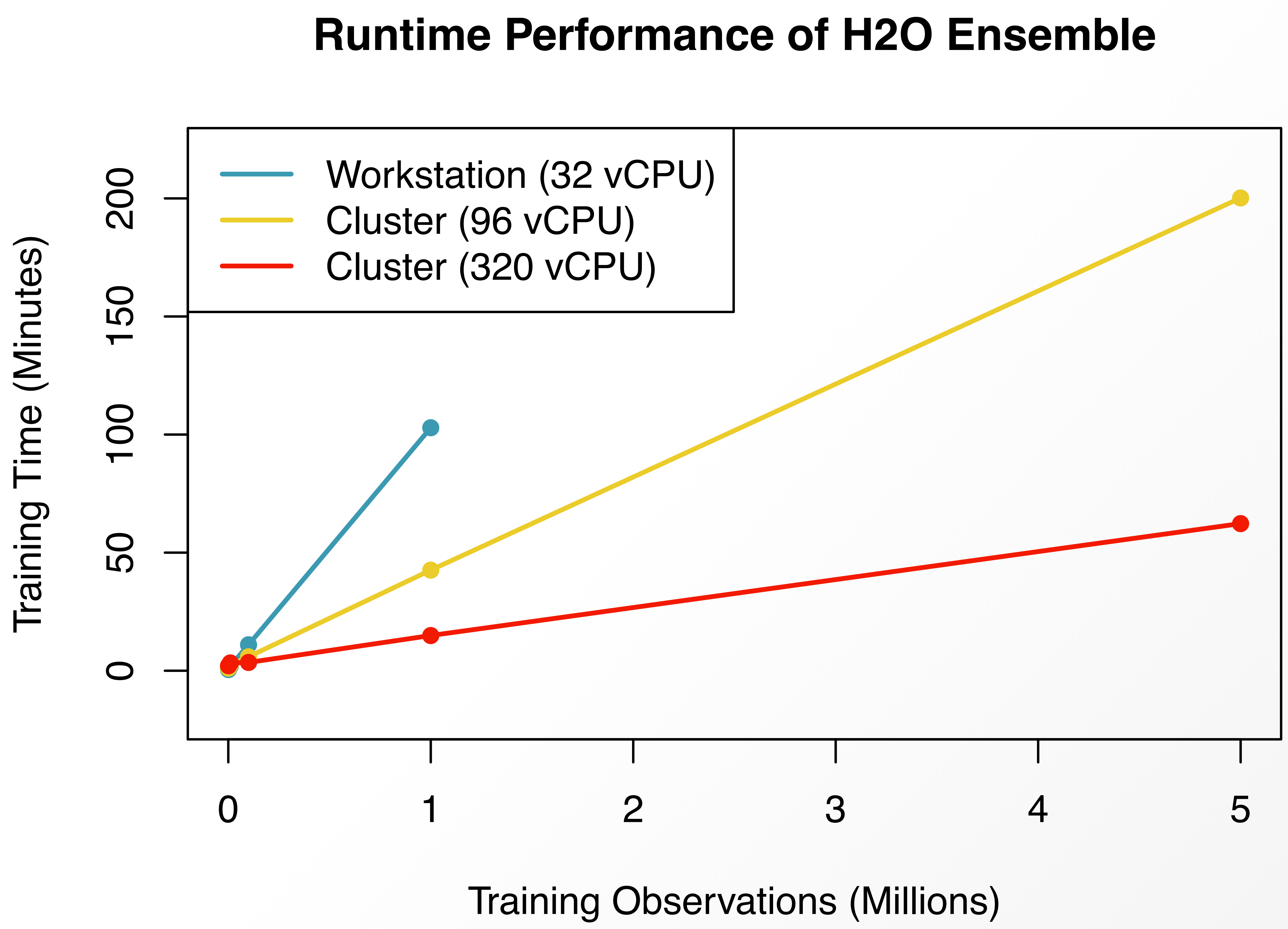
subsemble

- Implements the Subsemble algorithm for combining models trained on partitions of the data, a variant of Super Learning.
- Like SuperLearner, can be used with any R algorithm.

h2oEnsemble

- H2O Ensemble implements the standard Super Learner algorithm using H2O distributed algorithms.
- Includes functions for automatically creating diverse ensembles.

h2oEnsemble Scalability



Install R Packages

Example

```
install.packages("devtools")
library("devtools")

install_github("ecpolley/SuperLearner")
install_github("ledell/subsemble")

install.packages("h2o")
install_github("h2oai/h2o/R/ensemble/h2oEnsemble-package")
```

SuperLearner R Package

Example

```
SL.library <- c("SL.knn",
                "SL.glm",
                "SL.randomForest")

method <- "method.NNLS"

family <- "binomial"
```

SuperLearner R Package

Example

```
fit <- SuperLearner(Y = Y, X = X,
                    family = family,
                    SL.library = SL.library,
                    method = method)

pred <- predict(fit, newdata = newX)
```

subsemble R Package

Example

```
learner <- c("SL.knn",
             "SL.glm",
             "SL.randomForest")

metalearner <- "SL.nnls"

family <- "binomial"
subsets <- 4
```

subsemble R Package

Example

```
fit <- subsemble(x = x, y = y,
                 family = family,
                 learner = learner,
                 metalearner = metalearner,
                 subsets = subsets)

pred <- predict(fit, newx)
```

H2O Ensemble R Interface

Example

```
library(h2oEnsemble)  #Install from GitHub

learner <- c("h2o.randomForest.1",
             "h2o.deeplearning.1",
             "h2o.deeplearning.2")

metalearner <- "h2o.glm.wrapper"

family <- "binomial"
```

H2O Ensemble R Interface

Example

```
fit <- h2o.ensemble(x = x, y = y, training_frame = train,
                    family = family,
                    learner = learner,
                    metalearner = metalearner)

pred <- predict(fit, test)
```

h2o R Package

Installation

- The easiest way to install the h2o R package is to install directly from 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 Ensemble R Package

Branch: **master** ▾ h2o-3 / h2o-r / **ensemble** / +

ledell Update h2oEnsemble README — Latest commit **4824ede** a minute ago

..		
demos	Added save/load functions to h2oEnsemble	8 days ago
h2oEnsemble-package	Optimized predict.h2o.ensemble function	2 days ago
README.md	Update h2oEnsemble README	a minute ago
SuperLearner_wrappers.R	Added h2o-3 version of h2oEnsemble package	4 months ago
create_h2o_wrappers.R	Added example to h2o-r/ensemble/create_h2o_wrappers.R	4 months ago
example_twoClass_higgs.R	Updated higgs example in h2oEnsemble	5 days ago

README.md

H2O Ensemble

The `h2oEnsemble` R package provides functionality to create ensembles from the base learning algorithms that are accessible via the `h2o` R package (H2O version 3.0 and above). This type of ensemble learning is called "super learning", "stacked regression" or "stacking." The Super Learner algorithm learns the optimal combination of the base learner fits. In a 2007 article titled, "Super Learner," it was shown that the super learner ensemble represents an asymptotically optimal system for learning.

How to Win Kaggle

#	Δrank	Team Name * in the money	Score	Entries	Last Submission UTC (Best – Last Submission)
1	↑1	Perfect Storm *	0.869558	128	Thu, 15 Dec 2011 05:35:00 (-3.2d)
2	↑4	Gxav *	0.869295	54	Thu, 15 Dec 2011 09:41:23 (-26.9h)
3	↑14	occupy *	0.869288	9	Thu, 20 Oct 2011 00:40:05
4	↑16	D'yakonov Alexander (MSU, Moscow, Russia)	0.869197	64	Thu, 15 Dec 2011 22:08:19 (-5.1d)
5	↓1	Indy Actuaries	0.869135	23	Thu, 15 Dec 2011 18:35:33 (-2.9d)
6	↑20	UCI_Combination	0.869097	19	Tue, 06 Dec 2011 06:41:59 (-3.5d)
7	↑42	vsh	0.869034	26	Thu, 15 Dec 2011 19:16:44
8	↓1	Xooma	0.868984	74	Thu, 15 Dec 2011 23:25:53 (-1.8h)
9	↓8	vsu	0.868942	14	Thu, 15 Dec 2011 14:02:51 (-0h)
10	↑12	cointegral	0.868913	2	Mon, 21 Nov 2011 12:24:20

https://www.kaggle.com/c/GiveMeSomeCredit/leaderboard/private

How to Win Kaggle

The big learning experience for me is how strong a team can be if the skills of its members complement each other. Rather like an ensemble in fact. None of us would have got in the top placings as individuals.

What we basically did was extract about 25-35 features from the original dataset, and applied an ensemble of five different methods; a regression random forest, a classification random forest, a feed-forward neural network with a single hidden layer, a gradient regression tree boosting algorithm, and a gradient classification tree boosting algorithm. The neural network was a pain to implement properly but improved things by a decent amount over the bagging and boosting based elements.

#17 | Posted 3 years ago

Alec Stephenson

Competition **1st** | Overall **642nd**
Posts **82** | Votes **55**
Joined **1 Sep '10** | Email User

Permalink

https://www.kaggle.com/c/GiveMeSomeCredit/forums/t/1166/congratulations-to-the-winners/7229#post7229

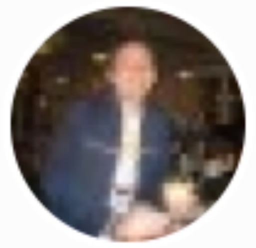
How to Win Kaggle

I used an ensemble of 15 models including GBMs, weighted GBMs, Random Forest, balanced Random Forest, GAM, weighted GAM (all with bernoulli/binomial error), SVM and bagged ensemble of SVMs.

I haven't try to fine tune each models individually but looked for diversity of fits.

My best score (0.89345, not in the private leaderboard as I haven't selected it in my final set) was an ensemble of 11 models which excluded the SVMs fits.

#18 | Posted 3 years ago



Xavier Conort

Competition **2nd** | Overall **33rd**
Posts **49** | Votes **94**
Joined **23 Sep '11** | Email User

Permalink

https://www.kaggle.com/c/GiveMeSomeCredit/forums/t/1166/congratulations-to-the-winners/7230#post7230

How to Win Kaggle

Arno Candel
@ArnoCandel

Following

Nothing like waking up to a top10 #kaggle @h2oai submission kaggle.com/c/santander-cu … #datascience #machinelearning

7 ↑25 Arno Candel H2O.ai 0.840915 9 Sat, 05 Mar 2016 15:38:05

Your Best Entry ↑

Top Ten!

You made the top ten by improving your score by 0.000483.

You just moved up 35 positions on the leaderboard. Tweet this!

RETWEETS 7 LIKES 14

7:41 AM - 5 Mar 2016

San Jose, CA

Stacking with Random Grids

New H2O Ensemble function in v0.1.8:

h2o.stack

http://tinyurl.com/h2o-randomgrid-stack-demo

H2O Ensemble Resources

H2O Ensemble training guide:
http://tinyurl.com/learn-h2o-ensemble

H2O Ensemble homepage on Github:
http://tinyurl.com/github-h2o-ensemble

H2O Ensemble R Demos:
http://tinyurl.com/h2o-ensemble-demos

Thank you!

@ledell on Github, Twitter

erin@h2o.ai

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