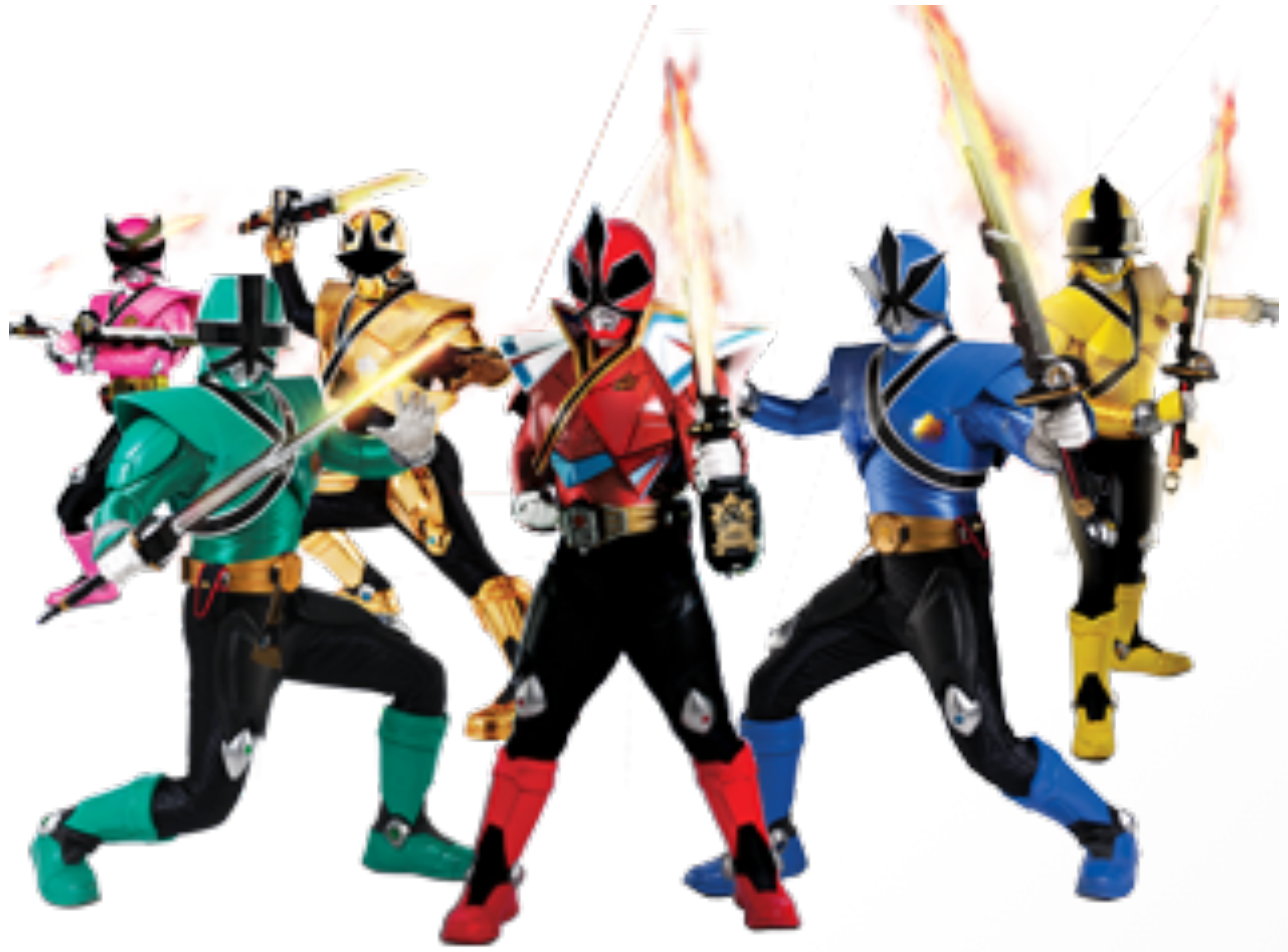
H2O Ensemble: New Developments



January 2017

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

Agenda

- Intro to Stacked Ensembles
- New/active developments in H2O Ensemble:
  - Stacking now in Java, R, Python, Flow
  - Stacking with Random Grids
  - AutoML
  - Third-party Integrations

Introduction to Stacking

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

- 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

Stacking (aka Super Learner Algorithm)

$$n\left\{\left[\overbrace{\qquad X \qquad}^{m}\right]\right. \quad \left[\,y\,\right]$$

"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

Stacking (aka 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

Stacking 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 Stacking, your computation does not go to waste!

h2oEnsemble R package
& Stacked Ensemble in h2o

Evolution of H2O Ensemble

- h2oEnsemble R package
  - PhD research project
  - first release in Nov 2014
  - major rewrite in 2015 & expansion in 2016
  - save/load binary models, but no POJO

Evolution of H2O Ensemble

- Stacked Ensembles in H2O
  - Currently in testing: “ensembles” branch
  - First release: end of Jan 2017
  - Ensemble logic ported to Java
  - R & Python APIs
  - Train ensembles from Flow GUI
  - POJO/MOJO for production use

Stacking with Random Grids

H2O Cartesian Grid Search

Example

```
hidden_opt <- list(c(200,200), c(100,300,100), c(500,500))
l1_opt <- c(1e-5,1e-7)
hyper_params <- list(hidden = hidden_opt, l1 = l1_opt)

grid <- h2o.grid(algorithm = "deeplearning",
                 hyper_params = hyper_params,
                 x = x, y = y,
                 training_frame = train,
                 validation_frame = valid)
```

H2O Random Grid Search

Example

```
search_criteria <- list(strategy = "RandomDiscrete",
                        max_runtime_secs = 600)

grid <- h2o.grid(algorithm = "deeplearning",
                 hyper_params = hyper_params,
                 search_criteria = search_criteria,
                 x = x, y = y,
                 training_frame = train,
                 validation_frame = valid)
```

Stacking with Random Grids (h2o R)

Example

```
# Create a list of all the base models
models <- c(gbm_models, rf_models, dl_models, glm_models)

# Let's stack!
fit <- h2o.stackedEnsemble(x = x, y = y,
                           selection_strategy="choose_all",
                           training_frame = train,
                           base_models = models)
```

Stacking with Random Grids (h2o Python)

Example

```
from h2o.estimators.stackedensemble \
 import H2OStackedEnsembleEstimator

# Let's stack!
stack = H2OStackedEnsembleEstimator( \
              selection_strategy="choose_all", \
              base_models=models)

stack.train(y=y, training_frame=train)
```

h2oEnsemble R package Resources

Training guide:

http://tinyurl.com/learn-h2o-ensemble

h2oEnsemble homepage on Github:

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

h2oEnsemble R Demos:

http://tinyurl.com/h2o-ensemble-demos

AutoML

H2O AutoML

- AutoML stands for "Automatic Machine Learning"
- The idea here is to remove most (or all) of the parameters from the algorithm, as well as automatically generate derived features that will aid in learning.
- Single algorithms are tuned automatically using a combination of grid search and Bayesian Optimization algorithms.
- If ensembles are permitted, then a Super Learner will be constructed.

Public code coming soon!

Third-Party Integrations

Ensemble H2O with Anything

A powerful combo: H2O + XGBoost

- SuperLearner & subsemble packages
- caret R package
- mlr R package
- Future work: h2o + sklearn?

Thank you!

@ledell on Github, Twitter

erin@h2o.ai

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