Scalable predictive models and hierarchical ensembles

Tomáš Frýda

Motivation

- Evaluate anytime learning capabilities of different models

- Find a better way to build ensembles

Plan

- Done
  - Implement a benchmarking environment
  - Integrate FAKEGAME into H2O
- Work In Progress
  - Benchmark H2O, H2O Ensemble and FAKEGAME
- To Do
  - Implement better optimization of hyperparameters

H2O

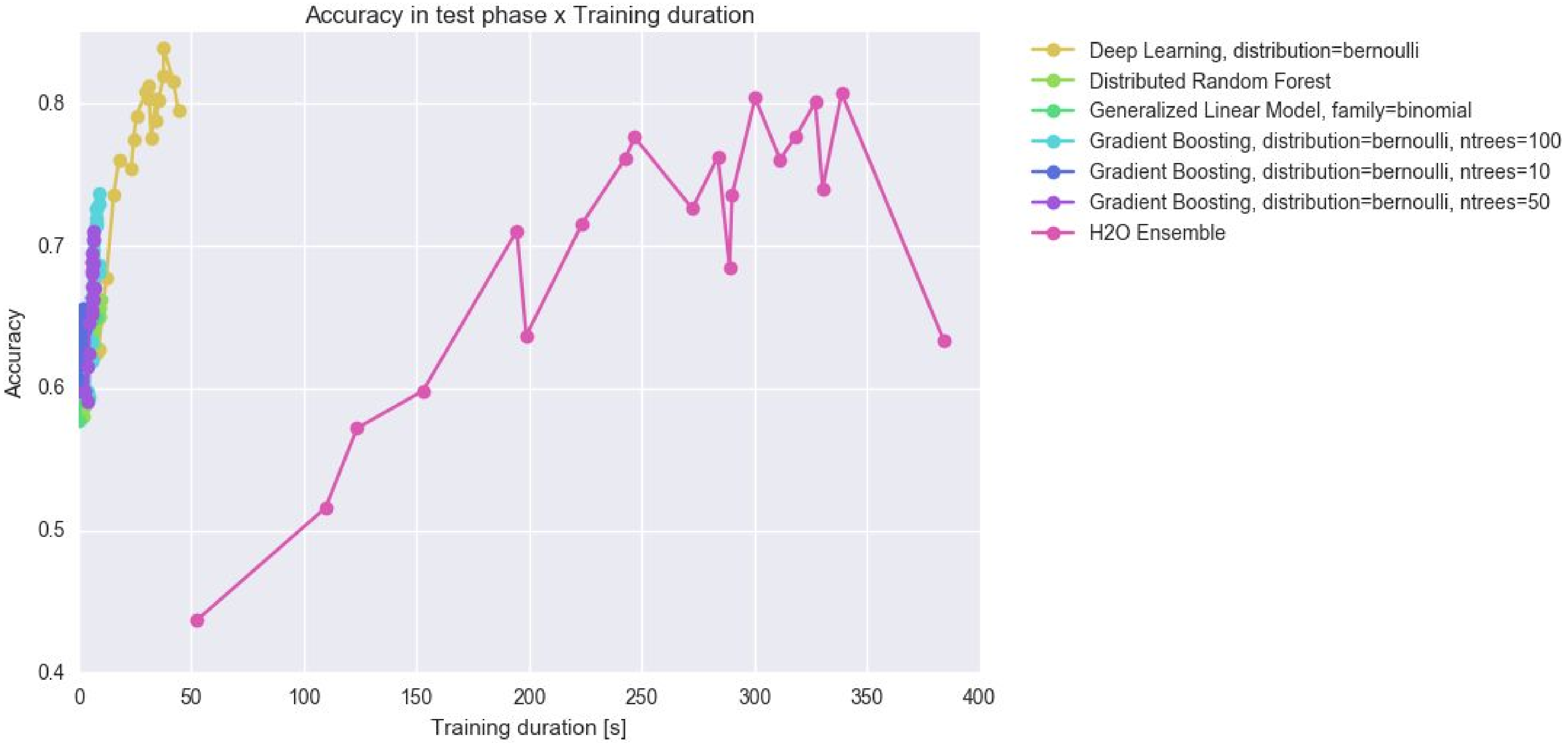
- Scalable machine learning framework
- Models written in a form of Map-Reduce jobs

- Consists of Java core and various bindings
    - All bindings use REST api to talk to H2O

Benchmarking Environment

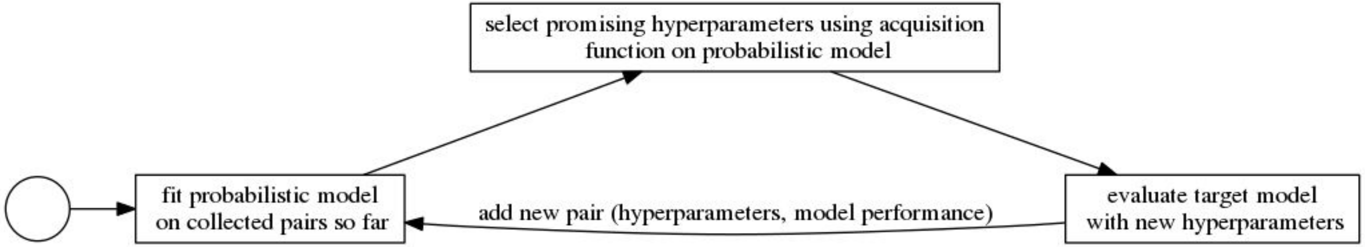
- Python module
- Anytime benchmarking realized using subsamples of original data
- Uses H2O Python bindings and rpy2 with H2O R bindings
  - To keep the same subsamples for all models (even for h2o-ensemble which is available only as R package)
- Experiments are simple YAML files
- Output is written into sqlite db as well as CSV

Results So Far



Future work

- Sequential Model-Based Bayesian Optimization



select promising hyperparameters using acquisition function on probabilistic model
fit probabilistic model on collected pairs so far
evaluate target model with new hyperparameters
add new pair (hyperparameters, model performance)

- Tree-structured Parzen Estimator
- Sequential Model-Based Algorithm Configuration
  - extended and improved SMBO

References

http://www.h2o.ai/

http://cig.felk.cvut.cz/projects/game/

http://fakegame.sourceforge.net/doku.php

https://www.cs.ubc.ca/~hutter/papers/10-TR-SMAC.pdf

http://hyperopt.github.io/hyperopt-sklearn/

https://papers.nips.cc/paper/4443-algorithms-for-hyper-parameter-optimization.pdf