AN EFFICIENT OPTIMIZATION ALGORITHM FOR AUTOMATIC TUNING OF MACHINE-LEARNING MODELS

Allocation: Exploratory/48 Knh
PI: Matthew West[1]
Co-PI: Chenchao Shou[1]

[1]University of Illinois at Urbana-Champaign

EXECUTIVE SUMMARY

Hyperparameters are crucial to the performance of a machine-learning algorithm. The difference between poor and good hyperparameters can mean the difference between a useless model and state-of-the-art performance. In this project, we developed an efficient optimization algorithm (Progressive Stochastic Response Surface, or ProSRS) for automatically tuning of hyperparameters. ProSRS exploits multiple cores of a machine by performing the tuning in parallel. We compared ProSRS to popular Bayesian optimization algorithms on a suite of standard benchmark functions and two real hyperparameter-tuning problems. ProSRS not only achieves significantly faster optimization convergence but is also one to three orders of magnitude cheaper in computational cost.

RESEARCH CHALLENGE

Machine learning has emerged in the past decade as one of the most exciting technologies, with widespread applications including object recognition, speech recognition, fraud detection, spam filtering, and recommender systems [1–5]. Many machine-learning algorithms have a set of tunable configuration parameters, known as hyperparameters. These hyperparameters (e.g., regularization constants, learning rates, etc.) generally have a huge impact on the performance of a machine-learning algorithm. Indeed, the difference between poor and good hyperparameter settings can mean the difference between a useless model and state-of-the-art performance [6].

Recently, interest has grown in developing automatic procedures for tuning hyperparameters of machine-learning algorithms. Among these procedures, Bayesian optimization is a popular method. However, one issue with this method is the high computational cost, which limits the range of the hyperparameter-tuning problems that it can be applied to. Therefore, there is a need to develop a more efficient optimization algorithm for hyperparameter-tuning applications.

METHODS & CODES

We developed an efficient optimization algorithm (ProSRS) for hyperparameter tuning. Unlike Bayesian optimization that uses Gaussian processes [7], our algorithm uses radial basis functions, which are much more efficient computationally. Moreover, we developed a novel "zoom strategy" to further improve the efficiency of the algorithm.

Our codes are structured in a master-worker configuration. For each iteration, the optimization algorithm (master) proposes multiple sets of hyperparameters to a batch of workers, with each worker being assigned to exactly one set of hyperparameters.
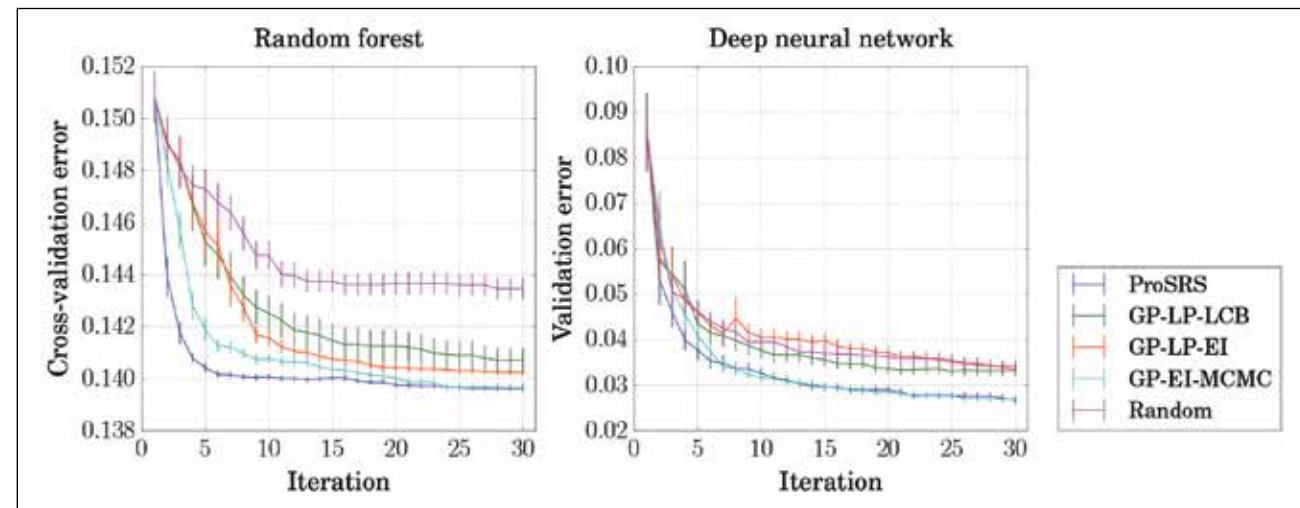

Figure 1: Performance of optimization algorithms for hyperparameter-tuning problems. The error bar shows the standard deviation of 20 independent runs. The ProSRS algorithm is best among all the algorithms.
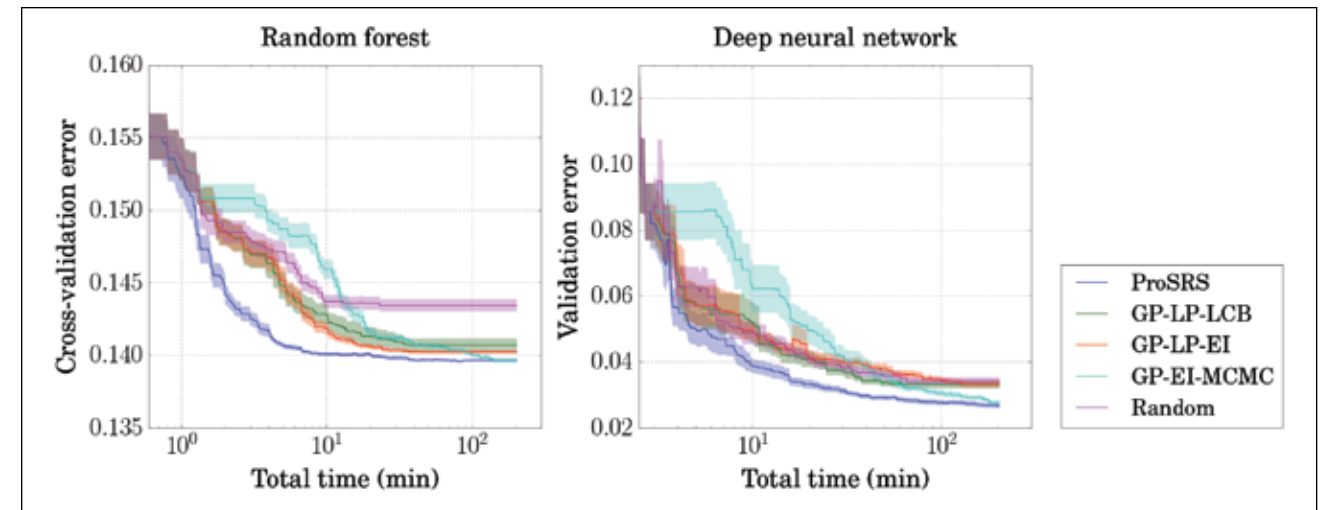

Figure 2: Optimization efficiency of different algorithms for hyperparameter-tuning problems. The total time on the horizontal axis is the actual elapsed time. Our ProSRS algorithm is significantly faster (more efficient) than the other algorithms.

A worker trains a machine-learning model with the assigned hyperparameters and returns the validation error of the model back to the master. The tasks of the workers are performed in parallel using multiple cores of a machine. Our algorithm is implemented in Python with common Python libraries (Numpy, Scipy, Scikit-learn, and Mpi4py).

RESULTS & IMPACT

We compared our algorithm to three state-of-the-art parallel Bayesian optimization algorithms: GP-EI-MCMC [8] and GP-LP [9] with LCB and EI acquisition functions. We tuned five hyperparameters of a random forest and seven hyperparameters of a deep neural network.

Fig. 1 shows the optimization performance versus iteration for different algorithms. We see that our ProSRS algorithm is better than the GP-EI-MCMC algorithm (particularly on the random forest tuning problem), and is much better than the two GP-LP algorithms and the random search algorithm. Fig. 2 shows the overall optimization efficiency that takes into account not only the optimization performance per iteration but also the cost of the algorithm and the cost of training machine-learning models. As we can see, our ProSRS algorithm is best among all the algorithms. Because of the high cost of the GP-EI-MCMC algorithm, the advantage of our algorithm over GP-EI-MCMC becomes even more pronounced compared to that of the iteration-based performance measurement (Fig. 1).

The fact that our algorithm shows superior optimization performance with significantly lower cost means that our algorithm is suitable for a wider range of hyperparameter-tuning problems, not just very expensive tuning problems.

WHY BLUE WATERS

Hyperparameter tuning requires training and evaluating machine-learning models for many iterations. To perform a full numerical experiment, we need to compound this base unit of computation with the number of tested algorithms, the number of hyperparameter-tuning problems, and the number of repeated runs for each algorithm and each problem. Therefore, this project demands large-scale computation that only Blue Waters can provide. The high-quality computing service offered by Blue Waters and the professionalism of the NCSA staff have been key to the success of the project.