The Effect of Hyper-Parameter Tuning on Machine Learning performance (106)

Patrick Ward
S.C.S.S.
Trinity College Dublin
Dublin
paward@tcd.ie

Eoin Roche
S.C.S.S.
Trinity College Dublin
Dublin
conevin@tcd.ie

Conor Nevin
S.C.S.S.
Trinity College Dublin
Dublin
rochee6@tcd.ie

1 INTRODUCTION

How researchers select hyperparameters can affect an algorithms ability to make predictions from data. This project evaluated the effectiveness of hyperparameters selection algorithms using the skLean implementations of Random Search and Grid search and the GPyOpt implementation of Bayesian learning. The regressor was implemented using the xgboost library. The present research suggests that without hyperparameter tuning algorithms, tuning can become somewhat of a black box which requires expert experience [2,3,4,5]. There are time, complexity and accuracy trade-offs to be considered when choosing a hyperparameters selection methodology. This project evaluated the effect of random search, grid search and Bayesian search hyperparameter selection algorithms using k-fold cross-validation on two linear regression machine learning problems. This research project was able to demonstrate significant improvements to overall accuracy via Bayesian learning. Although Bayesian Learning tended to produce the highest accuracy it took the most time to produce a result, random search tended to converge on a solution in the least time and tended to produce less accurate results than Bayesian learning. Grid was the least accurate and took the longest time to converge.

2 RELATED WORK

In recent years hyperparameter optimisation has begun to move from an art to a science [2]. Many of the papers written before 2011 seem to favour human judgment as for the optimal optimisation strategy [2, 6, 7]. The literature following this period tends to focus on the need for algorithms to create a more reliable process [1-9]. Research carried out since, focuses on grid search, random search and Bayesian learning [2-9]. Grid search will generally find the best solution but is not scalable for high numbers of parameters [10, 12], random search does not always converge on the best solution [4, 13], and Bayesian learning requires a relatively expensive calculation but tends to converge on the highest accuracy [5,6,8,9,13]. Grid search is a brute force approach, random search is based on trial and error, and Bayesian learning is a probabilistic method. There have been several papers contrasting two of these three methods [2-13] but there has been little focus from research on contrasting the three together.

3 METHODOLOGY

3.1 Evaluation Metrics

This project evaluated hyperparameters on how accurately they performed using k-fold cross-validation, how many iterations they took to converge on an optimal solution and how much time they took to achieve an optimal solution. K-fold cross-validation is a metric for the predictive value of a machine learning algorithm. It avoids biasing a solution towards the validation set by changing the validation set on each iteration. 10 was selected as a value for k due to the relatively small data sets used for our analysis. Larger values of K would have reduced the amount of data available for testing and would have affected our systems ability to get consistent results from the data. The algorithms accuracy was evaluated using mean squared error.

The number of iterations required to converge on an optimal solution was considered to measure the effectiveness of the algorithm on large data sets. When dealing with large data sets each iteration can take a significant amount of time, this metric was considered alongside the time taken for each iteration to give an measure for the scalability of the algorithms.

All comparisons between algorithms were made using the average values from running 30 iterations of each algorithm and constructing confidence intervals from the results.

3.2 Data Selection

The analysis was carried out using several datasets provided by the Scikit-learn python machine learning library. These data sets are provided for testing with the library because they are particularly well suited to machine learning. Since these data sets are used regularly alongside the Scikit-learn library they worked well with our hyperparameter selection algorithms and helped to reduce the complexity of our research problem. The datasets also have homogeneous pre-processing which reduced variability our testing. A drawback of these datasets is that they contain a relatively low number of data points (see table 1).

Table 1: Data used in evaluation

NAME	NUMBER OF INSTANCES	NUMBER OF PREDICTIVE ATTRIBUTES	PROLEM TYPE
BOSTON HOUSING PRICES	**506**	**13**	**Regression**
DIABETES PROGRESSION	**442**	**10**	**Regression**

3.3 Hyperparameter Selection Algorithms

This research project evaluated grid search, random search and Bayesian learning algorithms. We selected these algorithms to assess and compare the effectiveness of the three algorithmic styles widely discussed in the literature. Grid search is a brute force approach, random search is based on trial and error, and Bayesian learning is a probabilistic method.

The hyperparameters selected for were lambda, alpha, the regularisation penalty, the type of descent algorithm used and the type of feature selection algorithm used. Alpha and lambda are continuous variables, for the purposes of random search and Bayesian learning these variables were given a range intended to incorporate all reasonable values of the parameters. For the grid search, values at .1 intervals for both of these ranges were used. Grid search has a trade-off between the number of combinations required to satisfy the entire search and the granularity with which it can search.

It is worth noting that grid search might gain a higher overall accuracy if smaller intervals were used but for each interval, it requires checking that value against all other potential values for hyperparameters in the grid. .1 was selected as a reasonable middle ground between time and granularity of the search.

The descent algorithm, regularisation penalty and feature selection algorithm have discrete values which were given as parameters to each of the Hyperparameter Selection Algorithms, see table 1 for the ranges and values of each hyperparameter.

4 RESULTS AND DISCUSSION

4.1 Results

The evaluation of these three hyperparameter selection methodologies showed Bayesian learning to achieve the highest accuracy with an average mean squared error 7% lower than random search. Grid Search performed 28 percent worse than Bayesian on the Diabetes dataset and achieved a mean squared accuracy 12,000% higher than Bayesian on the Boston housing dataset (see table 2).

Table 2: Mean Squared Error by method

	BOSTON HOUSING	**DIABETES**
BAYESIAN LEARNING	23.15	3211
RANDOM SEARCH	25.00	3623
GRID SEARCH	3660	4551

Grid search was the fastest algorithm per iteration. This was followed by random search and then grid search (see table 3).

Table 3: Time in seconds taken per 20 iterations by method

	BOSTON HOUSING	**DIABETES**
BAYESIAN LEARNING	9s	10.3s
RANDOM SEARCH	6s	6.5s
GRID SEARCH	1.5s	1.1s

On average grid search took the least amount of iterations to converge on a solution. Grid search outperformed random search by 480% per 20 iteration. This was followed by Bayesian Learning which it performed 740% quicker (see table 4).

Table 4: Iterations to taken to converge

	BOSTON HOUSING	**DIABETES**
BAYESIAN LEARNING	8	3
RANDOM SEARCH	2	8
GRID SEARCH	45	36

Random search was on the fastest to converge. It converged in 10 percent less iterations than Bayesian learning. Grid search took the longest to converge. It converged in 700 percent less iterations than random search.

4.2 Discussion

Bayesian learning produces the most accurate results, producing a mean squared error 6% lower than the next best algorithm, which was random search (see table 2). Grid search converged on its solution in the most amount of time and was liable to miss the ideal solution as was demonstrated by the Boston housing dataset (see table 2) which caused it to perform approximately 12,000% worse than the next best algorithm. There is a tradeoff between random search and Bayesian Learning for time to converge and accuracy. Random search is on average 55% faster to converge than Bayesian but converges at an average of 6% lower accuracy (see tables 2 and 5).

Table 5: Mean time taken per 20 iterations multiplied by mean time to converge

	(MEAN TIME TO CONVERGE) X (MEAN TIME PER 20 ITERATIONS)
BAYESIAN LEARNING	53.1
RANDOM SEARCH	31.25
GRID SEARCH	52.65

5 LIMITATIONS AND OUTLOOK

The scope of this research could be extended by using a more varied pool of larger data sets, including more hyperparameter tuning methodologies and evaluating more machine learning methodologies. For example, this project did not review the effectiveness of optimisation methods such as gradient-based optimisation or evolutionary optimisation. It evaluated the machine learning algorithms for two datasets, the better sample size would allow for more conclusive results. Finally, the hyperparameters algorithms were evaluated on just linear regression problems. The scope of the project could be increased to consider other common machine learning methodologies such as logistic regression, support vector machines and clustering methods to allow for more generalised results about the effectiveness of the algorithms.

REFERENCES

[1] Koehrsen W. A Conceptual Explanation of Bayesian Hyperparameter Optimization for Machine Learning [Internet]. Towards Data Science. 2018 [29 October 2018]. Available from: https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f

[2] Snoek J, Larochelle H, Adams RP. Practical bayesian optimization of machine learning algorithms. InAdvances in neural information processing systems 2012 (pp. 2951-2959).

[3] Thornton, C., Hutter, F., Hoos, H. H., & Leyton-Brown, K. (2013, August). Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 847-855). ACM.

[4] Lins, I. D., Moura, M. C., Droguett, E. L., Zio, E., & Jacinto, C. M. (2011). Reliability prediction of oil wells by support vector machine with particle swarm optimization for variable selection and hyperparameter tuning. In Advances in Safety, Reliability and Risk Management (pp. 1499-1507). CRC Press.

[5] Claesen, M., & De Moor, B. (2015). Hyperparameter search in machine learning. arXiv preprint arXiv:1502.02127.

[6] Bergstra J, Yamins D, Cox DD. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures.

[7] Bergstra JS, Bardenet R, Bengio Y, Kégl B. Algorithms for hyper-parameter optimization. InAdvances in neural information processing systems 2011 (pp. 2546-2554).

[8] Lins, I. D., Moura, M. C., Droguett, E. L., Zio, E., & Jacinto, C. M. (2011). Reliability prediction of oil wells by support vector machine with particle swarm optimization for variable selection and hyperparameter tuning. In Advances in Safety, Reliability and Risk Management (pp. 1499-1507). CRC Press.

[9] Claesen, M., & De Moor, B. (2015). Hyperparameter search in machine learning. arXiv preprint arXiv:1502.02127.

[10] Claesen, M., Simm, J., Popovic, D., Moreau, Y., & De Moor, B. (2014). Easy hyperparameter search using Optunity. arXiv preprint arXiv:1412.1114.

[11] WANDEKOKEN, E. (2011). Support Vector Machine Ensemble Based on Feature and Hyperparameter Variation (Master's thesis, Universidade Federal do Espírito Santo).

[12] McGibbon, R. T., Hernández, C. X., Harrigan, M. P., Kearnes, S., Sultan, M. M., Jastrzebski, S., ... & Pande, V. S. (2016). Osprey: Hyperparameter

[13] Bergstra J, Yamins D, Cox DD. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures.