To Tune or not to Tune: Hyperparameter Influence on the Learning Curve

Prajit Bhaskaran[1] and Tom Viering[1]

Delft University of Technology, Mekelweg 5, 2628 CD Delft, Netherlands
P.Bhaskaran@student.tudelft.nl , T.J.Viering@tudelft.nl

Abstract. A learning curve displays the measure of error on test data of an ML algorithm trained on different amounts of training data. They can be modeled by parametric curve models that help predict accuracy improvement through extrapolation methods. However, most learning curve studies have only investigated learners with default hyperparameter settings. Research into tuning the learners and its effect on the learning curve has not been adequately researched. This research looks at the influence of hyperparameter tuning on the learning curve. We investigate how the learning curve shape changes and how different parametric models are affected when a learner undergoes tuning. We summarise the main findings of [1] in this abstract. Our work illustrates that hyperparameter tuning can remove unwanted learning curve behaviours, and that tuning may help improve learning curve extrapolation.

Keywords: learning curves · hyperparameter tuning · curve fitting

1 Experimental Setup

This abstract answers how the learning curve of a tuned learner differs from the default curve and whether a tuned learner may offer better curve fits. The datasets used are 6,31,37,42,50, 61,299,333,334, 715,737,823,923,1116,1120, 1462,1464, 1466,1504, 40536 from OpenML [2]. The datasets are preprocessed using a one hot encoding of categorical features and feature scaling. Training, validation and test sets are generated using stratified cross validation. Eight stratified folds are created generating eight curves which average to one curve.Figure 1 and 2 display the hyperparameters used for KNeighbors and Decision tree.

H-parameters	KN default	KN tuned
neighbors	5	[1,20]
weights	uniform	uniform, dist
p	2	2
metric	Minkowski	Minkowski

Table 1: Hyperparameters used for KNeighbors classifier

H-parameters	DT default	DT tuned
criterion	gini	gini,entropy,logloss
splitter	best	best, random
max depth	None	None, [1,10]
min sample split	2	2
min sample leaf	1	[1,12]
random state	42	42

Table 2: Hyperparameters used for decision tree classifier

For the parametric fitting of learning curves, we use models POW2, POW3, EXPP2, EXPP3, log2, logpower3 [3]. The nature of these models display a steep slope at early training sizes and then a gradual plateau making them strong candidates for curve fitting. The curve fitting procedure is outlined in [4]. Initial parameters were randomly generated, and the optimisation (using Levenberg-Marquadt algorithm) was repeated 25 times with random initial parameters.

2 Results

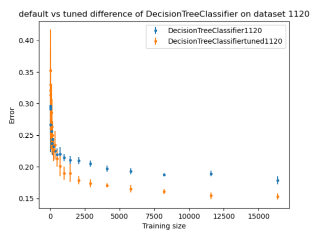

Fig. 1: Learning curves of tuned and default decision tree classifier

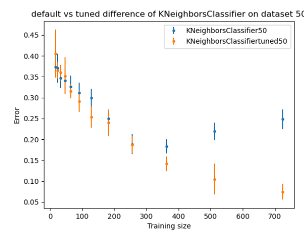

Fig. 2: Learning curves of tuned and default KNeighbors classifier

Figure 1 and 2 displays the learning curve for the decision tree and Kneighbors classifier on different datasets with vertical lines displaying the standard error. The first figure shows that as the training sizes increase, the tuned learner starts performing better than the default learner. This is expected as more training data allows for better model tuning. Occasionally, learning curves of default learners display non-monotone behaviours (error does not always decrease with more data), and in some cases the error significantly worsens, see Figure 2. Tuning can avoid such unexpected behaviours which can also be seen in the figure. 4/20 datasets displayed this phenomenon. Therefore, we suspect that curve fitting for these learning curves will also be easier.

One question that arises is whether learning curves of tuned models follow a certain family of curve fitting functions that may be different to the default learners, for example a power shape for a tuned learner and a logarithmic shape for a default learner. However, most curve fitting results before and after tuning are similar and do not seem to differ significantly. However, working with the four datasets that displayed non-monotne behaviour, the MSE for the tuned decision tree classifier is lower for 5/6 of the curve models and the tuned Kneighbors classifier has MSE values lower for all of the curve models. The standard deviation for both tuned learners is also considerably lower for each parametric curve fit. This suggests that when dealing with ill-behaved learning curves, tuning may offer much better curve fits for the different parametric models. This means that extrapolating learning curves for tuned learners may be significantly easier.

References

1. Bhaskaran, Prajit, and Viering, Tom. "To Tune or not to Tune: hyperparameter Influence on the Learning Curve". Online available at: https://repository.tudelft.nl/islandora/object/uuid5fe7cb314b22
2. Vanschoren, Joaquin, and van Rijn, Jan N., and Bischl, Bernd, and torgo, Luis. "OpenML: networked science in machine learning". SIGKDD Explorations 15 (2), 49-60. doi:10.1145/2641190.2641198
3. Viering, Tom, and Marco Loog. "The shape of learning curves: a review." arXiv preprint arXiv:2103.10948 (2021)
4. Mohr, Felix and Viering, Tom and Loog, Marco and van Rijn, Jan N. In Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECMLPKKD), Grenoble, France (2022).