In Search of Best Learning Curve Model

Dean Nguyen
Supervisors: Tom Viering, Marco Loog
EEMCS, Delft University of Technology, The Netherlands

June 19, 2022

A Dissertation Submitted to EEMCS faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering

Abstract

Learning curves have been used extensively to analyse learners' behaviour and practical tasks such as model selection, speeding up training and tuning models. Nonetheless, we still have a relatively limited understanding of the behaviour of learning curves themselves, in particular, whether there exists a parametric function that can best model all learning curves. Therefore, this study aims to determine which parametric models proposed over the years provide the best fit when applied to empirical learning curves. To answer this question, the study focuses on supervised learning and is divided into two parts: classification and regression tasks, and the learning curve data for each task was fitted using the Levenberg-Marquardt algorithm. Subsequently, the fitted models were analysed using the Friedman test, the Wilcoxon signed-rank test, and other metrics. The results indicate that a power law applies in most cases. However, a universal model has not been found, as the best model differs between classification and regression tasks, even though they belong to the power law family. Moreover, there are some deviations from these aggregate results when examining the learners individually, suggesting that a more granular approach is better suited for practical applications.

1 Introduction

Background & Motivation. In machine learning, specifically supervised learning, a crucial problem is the lack of labelled data and whether a learner's performance can be improved by acquiring more training data. Learning curves provide an insight into how a learning algorithm performs by plotting the performance or generalisation error against the size of the dataset that the algorithm has been trained on. Such a learning curve calculated from real data points of varying training set size is called an *empirical* learning curve [1]. For a formal and theoretical definition of learning curves, readers can look into [2], and [3]. Learning curves apply to a plethora of practical tasks such as model selection, extrapolation of the learning curve performance to reduce data collection costs, and speeding up training and tuning of models; which hold the potential to increase the efficiency of the machine learning pipeline significantly [2], [3]. Therefore, it is essential to develop accurate insights into learning curve behaviour. Over the years, many *parametric* models of learning curves have been proposed for extrapolation of the performance, which allows us to determine when we can halt the data collection process once an adequate performance is achieved [4]. However, most studies into these parametric models were done on only a tiny subset of learners and datasets and yielded divergent results in a comparative analysis [3]. Therefore, a systematic, empirical study with a careful experimental setup may illuminate which parametric models are generalisable across datasets and learners or whether such a universal model is even possible in a field as diverse as machine learning.

Research Question. The research question can be stated as follows:

> Which parametric learning curve model provides the best fit when applied to empirical learning curves?

Recently, a new learning curve database (LCDB) was created, which readily provides 150 GB of ground truth, and prediction vectors [1]. Given the recency of the creation of this database, the resulting data and its implications have not yet been fully explored. Therefore, rich and fruitful insights may result from further investigation and analysis.

Additionally, learning curves for regression tasks have not been looked into as closely as for classification tasks. Therefore, as suggested by [1], a new, general, extensible experimental setup is proposed here to create empirical learning curves for regression tasks. This allows us to see whether there is a difference in behaviour between learning curves of these two tasks and whether a universal parametric learning curve model for both tasks is possible.

Outline. The paper is presented in the following structure. First, section 2 presents findings from the literature and their limitations. Sections 3 and 4 then explain the methodology and general experimental setup in detail, which can be used for classification and regression tasks. Once an understanding of the experimental setup is obtained, section 5 presents the results of the experiments. Subsequently, section 6 discusses the results, limitations of the current research, and possible directions for future works. Finally, section 8 presents the ethical aspects of the project in the context of responsible research and how these have been addressed.

2 Related Work

In the past, multiple parametric models have been proposed in studies to fit empirical learning curves, as reviewed in [3]. The best fit can be determined by how well a parametric model interpolates over previously seen dataset sizes and how it extrapolates beyond them. In the review [3], Viering et al. gathered from the current literature that the power law provides the best fit for most models. The first of these studies which found the power law in the shape of learning curves is [4] by Frey and Fisher, which generated learning curves from the decision tree algorithm called C4.5 and fourteen small datasets, and fitted four parametric models to them. This finding was further confirmed and extended in [5] by Gu et al. with larger datasets, six parametric models with the notable inclusion of 4-parameter models, and the logistic discrimination approach next to the decision tree approach. Similarly, convincing results for the power law were found for neural networks [6], [7], [8].

Nonetheless, there were also deviations from these findings found in other studies. For example, Singh [9] found that the logarithm model performed the best in the context of four learning algorithms and the same four parametric models studied in [4]. Moreover, Brumen et al. [10] seemed to find that the exponential model had the best fit and advanced the hypothesis that the power law applies in aggregate, while the exponential model applies better for specific learners as proposed by Heathcote et al. [11]. This suggests that the focus should be on individual cases in specific contexts instead of relying on the results of analysing a large amount of data.

Although, as we have seen, there have been many studies on this subject, the results are still rather inconclusive, and there are still many limitations with the aforementioned studies. Firstly, most studies did not provide statistical tests, which question the significance of the results. Furthermore, in [4] and [9], the authors used the coefficient of determination R^2 to evaluate the model. R^2 is, however, an invalid metric since most models are nonlinear - an underlying assumption of R^2, and was shown in practice to be problematic if used for nonlinear models [12]. Additionally, in [10], only the performance of interpolation on training data points was examined. However, most benefits, such as reduction in data collection costs, can be derived from such a parametric model only through its extrapolation capability; therefore, interpolation is not adequate to determine the goodness of fit. Finally,

a significant limitation in these studies is that not many datasets, parametric models, and learners were investigated; in particular, 4-parameter models were only studied in [5] and [13], and there are no studies for regression tasks.

Altogether, the studies which showed deviation from the power laws have significant limitations and should be viewed carefully. Additionally, although there are some convincing results concerning the power law, these studies still suffer from some limitations, and a conclusive result cannot yet be determined given the field's current state.

3 Methodology

The general experimental methods were created with Thang - a colleague from the research group. The methodology allowed classification and regression experiments to be created easily and flexibly in different settings.

3.1 Learning Curve Creation

Firstly, k-fold cross-validation was used to create the k datasets for the individual learning curves. Using k-fold cross-validation generally allows for less bias in the test sets, which in turn allows for a more accurate measurement of the learner's performance, and is standard practice for estimating learning curves [3]. For each fold, the size of the training set, called an anchor a_i, was varied over a range of values determined by a schedule. Here, a geometric schedule was used to generate the anchors: $a_i := \lceil 2^{\frac{7+i}{2}} \rceil$, as used in [1], with i being the current index, starting from 1 up to the maximum i with a_i smaller than the dataset size. For each anchor, the learning algorithm was trained on a subset of the training set of that anchor S_i, and the rest of the unused data was then moved to the test set to avoid wasting data in conventional methods of generating learning curves [3]. The training subsets were also set to be monotonically increasing, that is, when training an algorithm on the set S_1 of instances and later on the set S_2 of instances where $|S_1| < |S_2|$, then $S_1 \subset S_2$. This allowed us to simulate the data acquisition process [1], which is the context in which the use of learning curves is most prevalent.

Moreover, the data was also preprocessed in several ways. For the numerical data of each dataset, first, it was preprocessed by imputing using the median for missing values. Then the numerical data was normalised using min-max scaling to avoid having numerical attributes having different scales as most machine learning algorithms do not perform well in that setting [14]. On the other hand, categorical attributes were encoded using one-hot encoding. Finally, to further simulate the data acquisition process, this preprocessing was only done at each training set S_i to avoid biasing the results.

The process was repeated for all folds to generate k individual learning curves; the final estimated empirical learning curve was their average.

3.2 Fitting & Evaluating Parametric Models

The parametric models were fitted to the generated learning curves using the *Levenberg-Marquardt* method to solve nonlinear least-square problems [15]. Additionally, the learning curves and the predictions from the fitted parametric model were normalised to a value between 0 and 1. Normalisation facilitated the comparison between classification and regression tasks since the performance of regressors can yield ranges that different order of magnitudes from one another depending on the dataset.

Since most benefits of a parametric model are derived through its extrapolation capability, only the extrapolation performance is examined. The metric of choice to evaluate the parametric models was the Mean Absolute Error (MAE), as MAE is particularly resistant to big outliers observed in the data. Additionally, any outliers such as NaN, infinity, or a number larger than 100 were removed, and the number of fits removed is reported in section 5. For analysing the results, first, the overall average rankings were analysed using code and methods from [1], with the Friedman's test to determine whether there are significant differences between curve models [16], and pairwise Wilcoxon signed-rank tests [17] with Holm's alpha correction $\alpha = 0.05$ [18] for comparing curve model pairs. In particular, the Friedman test ranks the parametric models for each curve fitting experiment and then takes the average of the ranks for each model to calculate the test statistics. All significant results, with $p < 0.05$ based on the Friedman test, were further examined using the post-hoc Wilcoxon signed-rank tests to determine where the differences occur. We visualise these results using Critical Diagrams (CDs), which display the average ranks of the different models (the lower, the better), and statistically non-significant pairs are tied together by a red bar. Using the average rank allows us to analyse the data in a manner less susceptible to outliers than comparing average MAE directly and is the primary method of determining how well a model performs compared to other models. Subsequently, further analyses of the average MAE and individual learners with divergent best models are presented to gain a more granular view of the results.

4 Experimental Setup

Classification. For classification tasks, the learning curve data was collected from the Learning Curve Database (LCDB) [1] instead of using the experimental setup. The LCDB was used because the data is much richer than what can be realistically carried out throughout this project. A combination of 20 learners and 246 datasets was used to calculate the results. Although the LCDB provides learning curves with several metrics, the learning curves analysed here use accuracy.

Sixteen parametric models, as shown in Table 1, were fitted using the `scipy` implementation of the Levenberg-Marquardt method. In particular, `last1` is a baseline model that always predicts the training anchors' last point. Curve fitting was done for different sizes of the overall training set, following [1], there were six partitions: "all" in which all experiments were utilised for fitting, "5%", in which anchors of up to a maximum of 5% of the training set size were utilised, "10%", "20%", "40%" and "80%". However, specialised methods involving clustering and preselection of initial points were utilised to fit the curves following research and fitting data received from Donghwi et al. [19]; as after some analysis for certain curves, the previously fitted parametric models generated in the LCDB proved to be not adequate.

Model	Formula	Model	Formula
last1	a	vap3	$\exp(a + \frac{b}{x} + c\log(x))$
pow2	$-ax^{-b}$	expp3	$c - \exp((-b + x)^a)$
log2	$-a\log(x) + b$	expd3	$c - (-a + c)\exp(-bx)$
exp2	$a\exp(-bx)$	logpow3	$a/((x\exp(-b))^c + 1)$
lin2	$ax + b$	pow4	$a - b(d + x)^{-c}$
ilog2	$-a/\log(x) + b$	mmf4	$(ab + cx^d)/(b + x^d)$
pow3	$a - bx^{-c}$	wbl4	$-b\exp(-ax^d) + c$
exp3	$a\exp(-bx) + c$	exp4	$c - \exp(-ax^d + b)$

Table 1: Parametric curve models

Regression. On the other hand, learning curves for regression tasks were created using k-fold to generate the fold, with $k = 25$; and the metric used to evaluate the learner was Mean Squared Error (MSE). The five learners investigated were `LinearRegression`, `SGDRegressor` , `GradientBoostingRegressor`, `SVR`, and `DecisionTreeRegressor` from the `scikit-learn` library using the default parameters. To train these learners, ten datasets were chosen from the OpenML platform, as shown in Table 2.

OpenML ID	Name	Nominal Attributes	Numeric Attributes	Number of Instances
189	kin8nm	0	8	8192
216	elevators	0	18	16599
218	house_8L	0	8	22784
315	us_crime	1	126	1994
503	wind	0	14	6574
537	houses	0	8	20640
562	cpu_small	0	12	8192
23515	sulfur	0	6	10081
42225	diamonds	3	6	53940
42726	abalone	1	7	4177

Table 2: Regression datasets

The parametric models were also fitted using the LevenbergâMarquardt algorithm for the sixteen parametric models in Table 1. However, since the number of datasets and learners was relatively small and the Donghwi et al. method was optimised for the LCDB, each curve was instead fitted 500 times with random initial points of a uniform distribution over $[0, 1)$, with the data being refitted each time if the performance was infinity or NaN. The fit with the best performance was used, with modified code from [1].

5 Results

The results are presented separately for classification, based on further analysis of the data from the LCDB, and for regression, based on data gathered from the previously described experimental setup. For classification, 1.3 % of the fits were removed due to outliers, and 7.5% of the fits were removed for regression. The remaining fits were analysed using normalised data with the Friedman test and pairwise Wilcoxon signed-rank and visualised using

Critical Diagrams, as described in section 3.2. An example of a normalised learning curve fitted to the model `pow4` is shown in Figure 1. The parametric model was fitted using the blue training anchors and evaluated using the red test anchors.

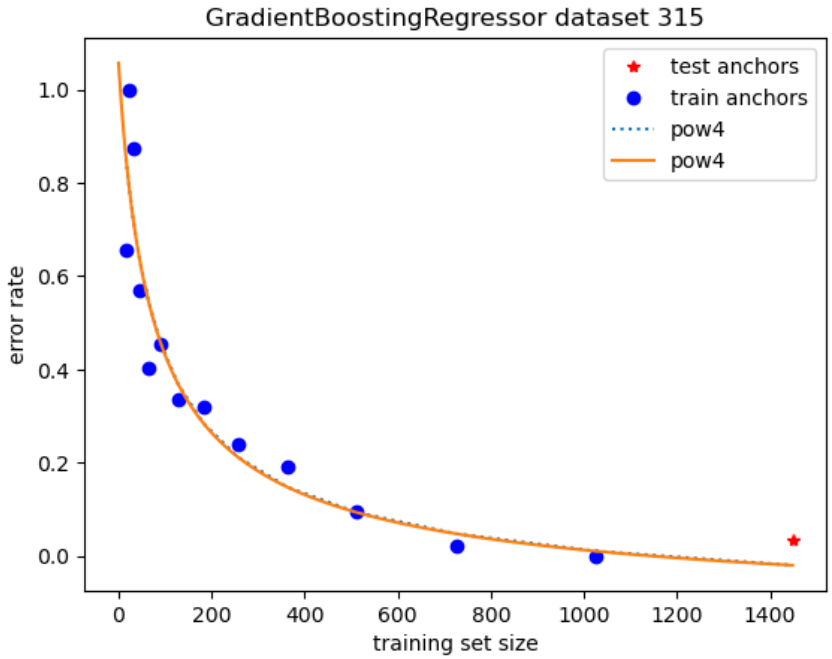
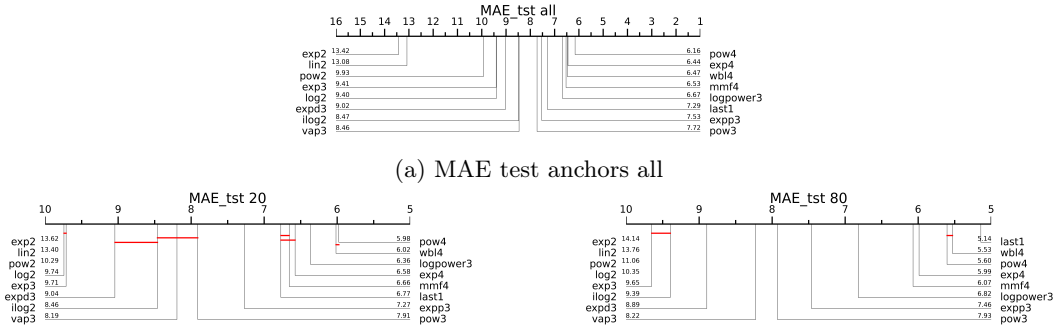


Figure 1: Example of a normalised parametric model fitted to a learning curve.

5.1 Classification

From Figure 2a, we can see that `pow4` performed significantly better than other models when considered over all experiments, as it has the lowest average rank, and a red line does not tie it to any other models. This agrees with the current literature for a power law, but for the fact that the `pow4` has four parameters. However, it was beaten by the baseline model `last1` when 80% of the training set is used (Figure 2c).



(a) MAE test anchors all

(b) MAE test anchors 20%

(c) MAE test anchors 80%

Figure 2: Critical diagrams for the ranks for the extrapolation to all test points for classification tasks. "all" considers all experiments, 20% fits learning curve to anchors up to 20% of the total dataset, likewise for 80%. If the two curve models are tied by a red line, their performances are not statistically significant. The number indicates the average rank, the lower, the better.

Although `pow4` was the curve model with the best average ranking overall, there were some divergent best curve models when looking at the average ranks of individual classifiers (over all fitting experiments of that learner), as shown in Table 3.

Learner	**Best Curve Model**
LinearDiscriminantAnalysis	last1
QuadraticDiscriminantAnalysis	last1
BernoulliNB	last1
SVC_sigmoid	last1, wbl4

Table 3: Learners with divergent best curve models in average rank (multiple best models are indicated if one is not significantly better than the others)

Additionally, looking at the average MAE in Table 4 shows that `last1` was in the lead instead:

Curve Model	Average MAE
last1	0.204 ± 0.0007
ilog2	0.244 ± 0.0008
mmf4	0.244 ± 0.0019
exp4	0.250 ± 0.0018
logpower3	0.256 ± 0.0024
wbl4	0.258 ± 0.0025
pow4	0.259 ± 0.0020
expp3	0.276 ± 0.0019
expd3	0.285 ± 0.0019
log2	0.292 ± 0.0012
pow3	0.305 ± 0.0021
pow2	0.311 ± 0.0012
exp3	0.312 ± 0.0020
vap3	0.323 ± 0.0020
lin2	0.572 ± 0.0022
exp2	0.606 ± 0.0026

Table 4: Average MAE and standard error of parametric models for classification tasks

Finally, the pie chart in Figure 3 shows the percentage of each model being in the number one position for each fitting experiment. Interestingly, `last1` also dominated and `pow4` has a much smaller slice than expected, with possible explanations discussed in section 6

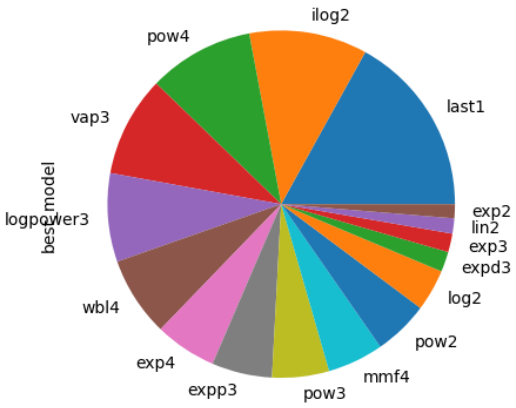
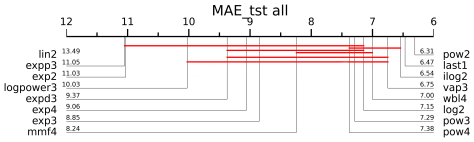


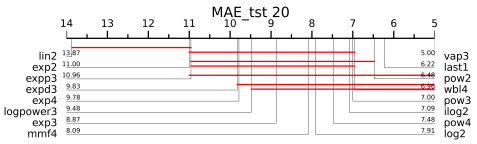
Figure 3: Pie chart of the best model for fitting experiments of classification tasks

5.2 Regression

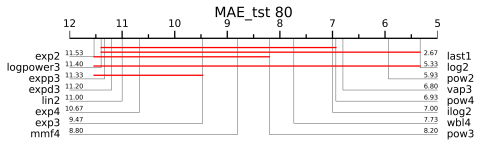
The results for regression tasks also seem to align with the current literature for a power law [3], with `pow2` having the best average rank for all experiments (Figure 4a). However, for anchors up to 20% and 80% of the dataset, `pow2` only stood in the third place.



(a) MAE test anchors all



(b) MAE test anchors 20%



(c) MAE test anchors 80%

Figure 4: Critical diagrams for the ranks for the extrapolation to all test points for regression tasks. "all" considers all experiments, 20% fits learning curve to anchors up to 20% of the total dataset, likewise for 80%. If the two curve models are tied by a red line, their performances are not statistically significant. The number indicates the average rank, the lower, the better.

If we look at the average MAE in Table 5, it shows a resemblance to that of the classification tasks in Table 4 with `last1` being on top and `pow2` in second place:

Curve Model	**Mean MAE**
last1	0.210 ± 0.0066
pow2	0.253 ± 0.0109
wbl4	0.270 ± 0.0119
ilog2	0.284 ± 0.0120
exp3	0.306 ± 0.0154
mmf4	0.313 ± 0.0148
expd3	0.315 ± 0.0144
logpower3	0.333 ± 0.0134
pow3	0.333 ± 0.0177
pow4	0.333 ± 0.0199
exp4	0.334 ± 0.0123
vap3	0.347 ± 0.0176
log2	0.375 ± 0.0134
expp3	0.474 ± 0.0220
exp2	0.484 ± 0.0200
lin2	1.062 ± 0.0376

Table 5: Average MAE and standard error of parametric models for regression tasks.

Similar to Figure 3, `last1` dominates the pie chart of Figure 5 with `pow2` also having a much smaller slice when it comes to the model that occupies the number one position for each fit.

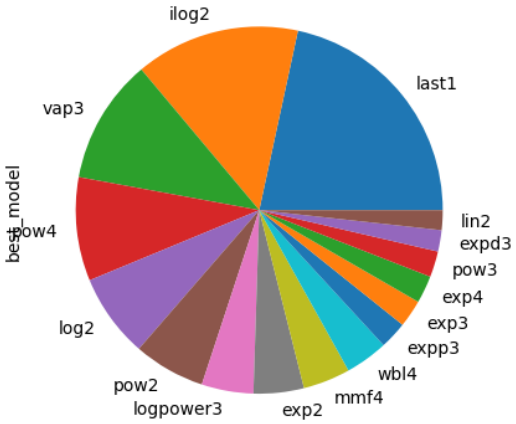


Figure 5: Pie chart of the best model for fitting experiments of regression tasks.

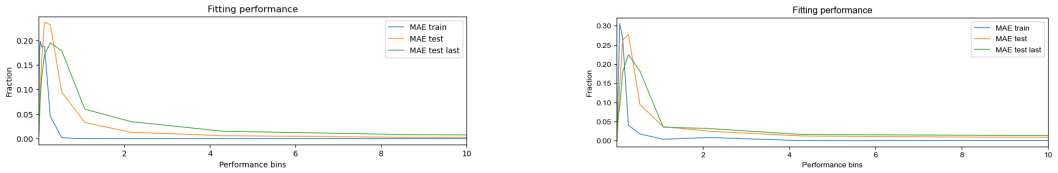
6 Discussion

The discussion proceeds by discussing the general findings of the experiments and theories as to what gives rise to the results. Next, the limitations of the research are reflected upon, and recommendations for future studies are given to guide future research on this topic.

6.1 General Findings

An important finding from the results is that although a power law applies in most cases, which corroborates the views of the current literature [3], a universal parametric model for learning curves is not yet realised. This is because different models are the best in different contexts. While a power law was discovered for both classification and regression, they were found with different numbers of parameters. For classification, a power law with four parameters was found; meanwhile, a power law with two parameters was found for regression. Moreover, looking at specific learners for classification tasks (Table 3), there were deviations from the power law, where `pow4` is overtaken by either `last1` or `wbl4`.

Another observation of the results is that different analyses show different curve models to be the best. For example, the best-found curves with average ranks differ from those found by looking at the average MAE, with `last1` being the best for both regression and classification tasks. This fact can be mainly explained by the existence of outliers, which can skew the result if a bad fit occurs, as seen in Figure 6. Meanwhile, the baseline model `last1` can never have significant outliers since it always predicts a point in the curve itself, always confining it to a range between 0 and 1 as all the curves are scaled. Therefore, an analysis based on average rank (i.e. the Friedman test) yields more robust results and allows us to determine the best model better.



(a) Classification fit performance

(b) Regression fit performance

Figure 6: Histogram summary of the fit performances for all fitting experiments

Similarly, if we look at the average ranks for test anchors up to 80% for both classification (Figure 2c) and regression (Figure 4c), we can see that `last1` provides the best extrapolation. This is due to how learning curves can often be expected to plateau for large sample sizes [1].

Finally, examining the pie charts of Figure 3 and Figure 5, we see that `last1` was the number one parametric model for the majority of fitting experiments for both classification and regression tasks instead of the ones found using the Friedman test. This further indicates that different models work better in different circumstances, and the best model is highly dependent on the analysis. Although the Friedman test gives the model which performs better on average and is more robust than other methods of analysis, it may not always give the best model in a specific context. Hence, perhaps a more individualised approach (for learners and tasks) may yield more fruitful results for practical applications.

6.2 Limitations & Future Recommendations

Although the research shows some promising results on how learning curves for supervised learning behave, there still exist many opportunities and mysteries to be uncovered in future studies.

Firstly, many factors have not been accounted for in the experimental setup, which may significantly affect the shape of the learning curves. These led to some learning curves with little performance improvements even with a large amount of data, as discussed in the previous subsection. For example, preprocessing has not been investigated extensively. Most of the preprocessing done for the experiments was admittedly somewhat arbitrary and was done mainly to ensure that the learners still achieved acceptable results. However, exploring the data and preprocessing it plays a significant role in the machine learning pipeline, and different learners and datasets may require different preprocessing techniques. Hence, a promising avenue would be to look extensively into how preprocessing should be done and tailored to each learner and whether we can uncover different laws once preprocessing has been optimised, as seen for hyperparameter tuning for deep neural networks [6]. Similarly, individual studies for hyperparameters can yield fruitful results. It would be interesting to see the effects of combining optimised preprocessing and hyperparameter tuning on different learners and how it affects the rankings of the different parametric models.

For regression tasks, as seen in Figure 4a, although we found that `pow2` is significantly better than other models, there are many more insignificant pairs compared to classification tasks, evident from the multitude of red lines. The number of insignificant pairs may be due to the relatively small number of learners and datasets investigated. Therefore, a more extensive experiment with more datasets and learners may yield more convincing results and more definitive rankings of all the different learners.

Moreover, the task of fitting the learning curves turns out to be much more complicated than previously expected, and the fitting of learning curves primarily relies on the previous work done by Mohr et al. [1] and the new research done by Donghwi et al. [19]. These works resolved issues for several problematic models; however, significant research still needs to be done into the fitting of these parametric curve models in the future before definitive conclusions can be reached.

Finally, all these recommendations can be combined. Once studies on preprocessing and hyperparameter tuning reach a sufficient level of understanding, they can be carried out with more data, learners, and tasks, which may yield more insights about the nature of learning curves and which parametric model can fit them best.

7 Conclusions

In summary, this research project aims to investigate which parametric learning curve model provides the best fit for empirical learning curves and whether there indeed exists a universal model encompassing all kinds of machine learning algorithms, tasks, and datasets. First, the literature was surveyed, indicating that a power law applies in most cases, although some studies find deviations under certain faulty conditions. To further verify and increase the robustness of these findings, additional analyses were done on learning curves in the LCDB for classification tasks, and new experiments were carried out for regression tasks - a context not yet studied. An important finding is that although a power law applies in most cases, no universal model is confirmed for all tasks, learners, and datasets. For classification tasks, the robust average rank analysis based on the Friedman test indicates that `pow4` is significantly better than all other curve models when considering all fits. For regression tasks, the average rank analysis indicates that `pow2` is significantly the best model, which is also a power law but with only two parameters. Moreover, if we take a closer look at the individual learners, there are already deviations from these aggregated results (Table 3). Therefore, a more individualised approach may yield fruitful results for practical applications. Finally, the project still has certain limitations, such as the limited number of datasets, learners investigated for regression tasks, and difficulties in fitting the learning curves. Addressing these issues may uncover more intriguing patterns and results in future studies and lead to significant benefits in the ubiquitous field of machine learning.

8 Responsible Research

A crucial aspect of responsible research is reproducibility and addressing the reproducibility crisis [20]. We reflect on this using the Yale Law School Roundtable's six recommendations for reproducible research [21]. Firstly, it is recommended that the source code and data be made available publicly. The source code is hosted on GitHub and can be accessed via this link, and the datasets used are described in section 4. Secondly, the released code should be versioned using a unique ID; however, this is not followed since the source code will not be updated for the foreseeable future. Thirdly, it is recommended that the computing environment and software versions are described. The experiment and fitting scripts were run on a DefltBlue supercomputer compute node using one Intel Xeon 2648R ('Cascade Lake') processor using the Linux operating system [22]. Meanwhile, the analysis scripts were run locally on an Asus TUF Dash F15 laptop with an Intel i7-11370H processor. The software environment is described in the GitHub repository in a `yaml` file. Per the fourth

recommendation, the source code is published with an MIT licence to facilitate reuse. The paper will also be available in the TU Delft education repository, which satisfies the fifth recommendation. Finally, the code and data generated all use non-proprietary, open-source software, which can be expected to be readable well into the future as per recommendation six.

Additionally, scientific integrity is another critical aspect that must be considered. Precautions have been taken to avoid pitfalls and inadvertent misconduct. All data were used in the analysis and transparently provided above, which can also be regenerated using the code provided. Furthermore, the cherry-picking of data was avoided, and where data was discarded has been communicated, for example, in 3.2. Therefore, most kinds of possible fraud can be ruled out. Care is also taken to avoid plagiarism, and any original ideas not the author's own are attributed and cited in the text. Lastly, limitations of the research are reflected in and addressed in section 6.

Finally, in general, there are relatively few ethical aspects to consider in the project apart from very general concerns. The learning curve is a very general method of analysis and optimisation that can be used in various contexts. Although the work presented here can be applied to increase the efficiency of the machine learning pipeline, which will be of great use in all kinds of applications, some of these applications may also be nefarious, such as the improvement of facial recognition technology to control a population [23]. Nonetheless, any improvements in technology can lead to inadvertent consequences due to the actions of bad-faith actors; therefore, the ethical education and preparation of engineers are of primary importance, which, however, cannot be addressed directly in this paper.

References

[1] F. Mohr, T. J. Viering, M. Loog, and J. N. van Rijn, "Lcdb 1.0: An extensive learning curves database for classification tasks," unpublished.

[2] F. Mohr and J. N. van Rijn, "Learning curves for decision making in supervised machine learning - A survey," *CoRR*, vol. abs/2201.12150, 2022.

[3] T. J. Viering and M. Loog, "The shape of learning curves: a review," *CoRR*, vol. abs/2103.10948, 2021.

[4] L. J. Frey and D. H. Fisher, "Modeling decision tree performance with the power law," in *Seventh International Workshop on Artificial Intelligence and Statistics*. PMLR, 1999.

[5] B. Gu, F. Hu, and H. Liu, "Modelling classification performance for large data sets," in *International Conference on Web-Age Information Management*. Springer, 2001, pp. 317–328.

[6] J. Hestness, S. Narang, N. Ardalani, G. Diamos, H. Jun, H. Kianinejad, M. Patwary, M. Ali, Y. Yang, and Y. Zhou, "Deep learning scaling is predictable, empirically," *arXiv preprint arXiv:1712.00409*, 2017.

[7] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[8] J. S. Rosenfeld, A. Rosenfeld, Y. Belinkov, and N. Shavit, "A constructive prediction of the generalization error across scales," *arXiv preprint arXiv:1909.12673*, 2019.

[9] S. Singh, "Modeling performance of different classification methods: deviation from the power law," *Project Report, Department of Computer Science, Vanderbilt University, USA*, 2005.

[10] B. Brumen, I. Rozman, M. Heričko, A. Černezel, and M. Hölbl, "Best-fit learning curve model for the c4. 5 algorithm," *Informatica*, vol. 25, no. 3, pp. 385–399, 2014.

[11] A. Heathcote, S. Brown, and D. J. Mewhort, "The power law repealed: The case for an exponential law of practice," *Psychonomic bulletin & review*, vol. 7, no. 2, pp. 185–207, 2000.

[12] A.-N. Spiess and N. Neumeyer, "An evaluation of r2 as an inadequate measure for nonlinear models in pharmacological and biochemical research: a monte carlo approach," *BMC pharmacology*, vol. 10, no. 1, pp. 1–11, 2010.

[13] P. Kolachina, N. Cancedda, M. Dymetman, and S. Venkatapathy, "Prediction of learning curves in machine translation," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2012, pp. 22–30.

[14] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly, 2019.

[15] H. P. Gavin, "The levenberg-marquardt algorithm for nonlinear least squares curve-fitting problems," *Department of Civil and Environmental Engineering, Duke University*, vol. 19, 2019.

[16] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940.

[17] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in statistics*. Springer, 1992, pp. 196–202.

[18] S. Holm, "A simple sequentially rejective multiple test procedure," *Scandinavian journal of statistics*, pp. 65–70, 1979.

[19] D. Kim, T. J. Viering, and M. Loog, "Different approaches to fitting and extrapolating the learning curve," unpublished.

[20] M. Baker, "Reproducibility crisis," *Nature*, vol. 533, no. 26, pp. 353–66, 2016.

[21] "Reproducible research," *Computing in Science Engineering*, vol. 12, no. 5, pp. 8–13, 2010.

[22] Delft High Performance Computing Centre (DHPC), "DelftBlue Supercomputer (Phase 1)," https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1, 2022.

[23] J. Wakefield, "Ai emotion-detection software tested on uyghurs," *BBC News*, May 2021.