SQUEEZING LEMONS WITH HAMMERS: AN EVALUATION OF AUTOML AND TABULAR DEEP LEARNING FOR DATA-SCARCE CLASSIFICATION APPLICATIONS

Ricardo Knauer and Erik Rodner
KI-Werkstatt, University of Applied Sciences Berlin
https://kiwerkstatt.htw-berlin.de
ricardo.knauer@htw-berlin.de

ABSTRACT

Many industry verticals are confronted with small-sized tabular data. In this low-data regime, it is currently unclear whether the best performance can be expected from simple baselines, or more complex machine learning approaches that leverage meta-learning and ensembling. On 44 tabular classification datasets with sample sizes ≤ 500, we find that L2-regularized logistic regression performs similar to state-of-the-art automated machine learning (AutoML) frameworks (AutoPrognosis, AutoGluon) and off-the-shelf deep neural networks (TabPFN, HyperFast) on the majority of the benchmark datasets. We therefore recommend to consider logistic regression as the first choice for data-scarce applications with tabular data and provide practitioners with best practices for further method selection.

1 INTRODUCTION

Machine learning algorithms thrive on data (Banko & Brill, 2001; Halevy et al., 2009). In particular, the availability of massive text corpora crawled from the web has been the foundation for large language models such as GPT-3, and has given rise to astounding emergent abilities such as in-context learning (Brown et al., 2020). In other domains, though, data are not as easy to come by. In the clinical diagnostic setting, for instance, medical conditions can be too rare to collect many instances; in the clinical prognostic setting, patients can be lost at future follow-ups. Under these circumstances, much simpler predictive models have traditionally been advocated (Moons et al., 2015; Steyerberg, 2019). This preference for simpler models is supported by a systematic review that indeed did not find a difference in discriminative performance between logistic regression and more complex machine learning models on average (Christodoulou et al., 2019). But also beyond the clinical domain, it has been recommended to try simpler baselines first before switching to more complex models (McElfresh et al., 2023). In the largest tabular data analysis to date, the overall best performance was achieved by a meta-trained transformer ensemble (Hollmann et al., 2023); perhaps more importantly, simpler models like logistic regression performed on par with or better than more complex models on 31% of the datasets (McElfresh et al., 2023). However, with median sample sizes > 1000, these large-scale analyses only provide insufficient guidance for practitioners to select the most suitable machine learning approach when data is limited. In particular, meta-learning and ensembling appear to be well-suited for the low-data regime (Hollmann et al., 2023; McElfresh et al., 2023), but it remains unclear if or when approaches that leverage these strategies, such as automated machine learning (AutoML) (Alaa & van der Schaar, 2018; Erickson et al., 2020; Imrie et al., 2023; Salinas & Erickson, 2023) or pretrained deep neural networks (Bonet et al., 2024; Hollmann et al., 2023; Müller et al., 2023), indeed offer an appreciable performance benefit for data-scarce applications (e.g., with sample sizes ≤ 500).

Our contributions are as follows:

1. We perform the **first extensive comparison of state-of-the-art machine learning methods, i.e., AutoML frameworks and off-the-shelf deep learning models, against a simple L2-regularized logistic regression baseline on 44 (very-)small-sized tabular datasets**.

In summary, we find that L2-regularized logistic regression shows a similar discriminative performance to more complex models on 55% of the datasets (Sect. 3.2), and that correlations of simple dataset meta-features with performance differences are only small (Sect. 3.3). We therefore recommend to use more complex models only if the performance with logistic regression is considered insufficient for the application at hand, in line with McElfresh et al. (2023).

2. We provide practitioners with the **best L2-regularization hyperparameter obtained on each dataset that can be used for meta-learning in data-scarce applications** (Sect. 3.2).

2 Related work

Challenges in the low-data regime Data-scarce applications pose unique challenges for machine learning, especially when combined with a relatively low number of labels in the smaller class per feature, or outcome events per variable (EPV). Clearly, overfitting is a concern for more complex models. Perhaps less obviously, finding good hyperparameter settings also becomes more difficult when data is limited. Low sample sizes produce cross-validation folds that are even smaller than the original sample, which may in turn be too small to adequately represent both the original sample and the population of interest, causing suboptimal hyperparameters. In case of regularized logistic regression, for instance, it has been shown that penalty strengths are generally estimated with increasing uncertainty as the sample size or EPV decreases, which can increase the between-sample variability of the predictive performance (Riley et al., 2021; Šinkovec et al., 2021; Van Calster et al., 2020). This implies that data-driven hyperparameter optimization methods may fail to increase, or may even decrease, the predictive performance for an individual dataset.

Machine learning solutions The traditional approach to attenuate overfitting and facilitate the hyperparameter search process in the low-data regime is to use regularized logistic regression with a constrained search space (Moons et al., 2015; Steyerberg, 2019); recent solutions like AutoPrognosis, AutoGluon, TabPFN, and HyperFast leverage more complex models in combination with meta-learning and ensembling. AutoPrognosis (Alaa & van der Schaar, 2018; Imrie et al., 2023) is an AutoML framework that automates building end-to-end machine learning pipelines, including preprocessing, model, and hyperparameter selection. Both logistic regression and tree ensembles are considered for algorithm search. Its default pipeline optimization procedure is initialized using meta-learned hyperparameters from external datasets, and a pipeline ensemble is constructed following the pipeline search process. In contrast to AutoPrognosis, AutoGluon (Erickson et al., 2020; Salinas & Erickson, 2023) is an AutoML framework that meta-learns a model hyperparameter portfolio from external datasets and different random seeds (zero-short hyperparameter optimization), and considers k-nearest neighbors, tree ensembles, and neural networks for selection. It focuses the training time budget on ensembling instead of further hyperparameter optimization. Recent off-the-shelf deep learning models, sometimes referred to as tabular foundation models (Müller et al., 2023), also prioritize ensembling over hyperparameter tuning. TabPFN is a meta-trained transformer ensemble that performs in-context learning on tabular data (Hollmann et al., 2023). HyperFast meta-trains a hypernetwork with external datasets, uses the pretrained hypernetwork to generate smaller, task-specific main networks with the actual training dataset, and optionally fine-tunes and ensembles the main networks (Bonet et al., 2024; Müller et al., 2023). Overall, AutoGluon, TabPFN, and HyperFast have generally outperformed (L1-, L2- or unregularized) logistic regression on 40 small- to medium-sized tabular datasets (median sample size 1241, range [500, 45,211]) (Bonet et al., 2024; Hollmann et al., 2023; Müller et al., 2023; Puri et al., 2023). In the next section, we extend the prior results to the data-scarce setting, by evaluating AutoPrognosis, AutoGluon, TabPFN, as well as HyperFast against a simple L2-regularized logistic regression baseline on 44 smaller-sized tabular datasets (median sample size 204, range [32, 500]).

3 Experiments

In the following, we describe our experimental setup to assess two AutoML frameworks and two pretrained deep neural networks against L2-regularized logistic regression (Sect. 2) on 44 (very-) small-sized tabular datasets with a 1h runtime limit on 8 vCPU cores with 32GiB memory. We then

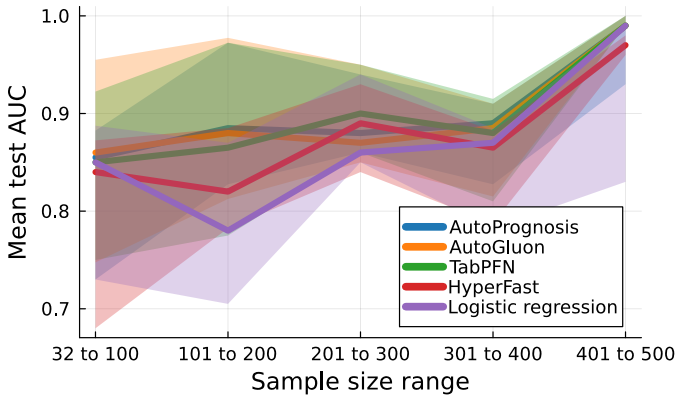

Figure 1: Mean test AUCs with their medians and interquartile ranges shown across sample size ranges in steps of 100 for AutoML, deep learning, and logistic regression.

present our experimental results together with the best L2-regularization hyperparameter setting for each dataset. Additionally, we conduct a meta-feature analysis to evaluate under which conditions, i.e., dataset properties, AutoML and deep learning methods perform better than logistic regression. Overall, we observe a similar discriminative performance for L2-regularized logistic regression to more complex models on 55% of the datasets. Furthermore, correlations of simple dataset meta-features with performance differences are only small. We therefore recommend to generally start predictive modeling in the low-data regime with a logistic regression baseline first, and to use more complex approaches only if the obtained performance is insufficient, echoing the conclusions of McElfresh et al. (2023).

3.1 Experimental setup

Datasets and methods We used all binary classification datasets with sample sizes ≤ 500 from Penn Machine Learning Benchmarks (PMLB) (Olson et al., 2017; Romano et al., 2022), yielding 44 curated tabular datasets in total (median sample size 204, range [32, 500]). We shuffled all datasets, encoded all labels to $\{0, 1\}$ for AutoML and deep learning methods and to $\{-1, 1\}$ for L2-regularized logistic regression, subsampled the clean1 dataset to 100 features for TabPFN (Feuer et al., 2023), and "min-max" scaled all features for logistic regression. We used AutoPrognosis 0.1.21 with the default settings, AutoGluon 1.0.0 with the "best quality" preset, and TabPFN 0.1.9 with 32 ensemble members (Hollmann et al., 2023). For HyperFast 0.1.3, we observed runtimes exceeding our 1h budget using the default settings, so we decided not to fine-tune the main networks, but instead to increase the number of ensemble members to 32 (Bonet et al., 2024; Hollmann et al., 2023). For L2-regularized logistic regression, we used a continuous conic formulation (Knauer & Rodner, 2023) with the default optimality gaps via MOSEK 10 (MOSEK ApS, 2023) and JuMP 1.4.0 (Dunning et al., 2017) from Julia 1.8.3.

Evaluation metrics For each dataset and method, we measured the training area under the receiver operating characteristic curve (AUC) on the whole data with a 1h runtime limit, as well as the mean test AUC via a stratified, 3-fold cross-validation procedure with a 1h runtime limit per fold (i.e., 4h per dataset in total). For L2-regularized logistic regression, we instead used cross-validation to tune the L2-regularization hyperparameter λ. We set λ to one value of the geometric sequence [0.5, 0.1, 0.02, 0.004], used the deviance to find the best hyperparameter value λ^*, and tracked the training AUC with λ^*. To compute the mean test AUC, we used a nested cross-validation procedure (Knauer & Rodner, 2023).

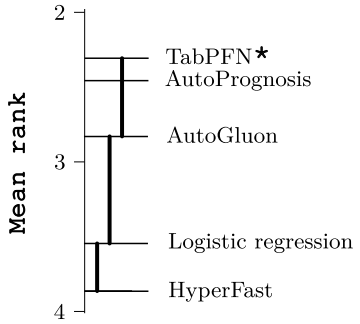

Figure 2: Critical difference diagram to detect pairwise mean test AUC differences, based on the Holm-adjusted Wilcoxon signed-rank test (Demšar, 2006; Benavoli et al., 2016), for AutoML, deep learning, and logistic regression. Approaches that are not statistically different at the 0.05 significance level are connected by a bold vertical bar. *Note that TabPFN results are biased (Sect. 3.2).

3.2 Experimental results

AutoPrognosis, AutoGluon, TabPFN, HyperFast, and logistic regression perform relatively similar at different sample sizes, with a trend towards better performance when the sample size increases for all methods (Fig. 1). The largest difference occurs at sample sizes from 101 to 200 where HyperFast and logistic regression only achieve a median discriminative performance of 0.82 and 0.78, respectively. Similar to McElfresh et al. (2023), though, we find that each approach wins on at least one dataset and looses on at least one other dataset (Table 1 in Appendix A). Logistic regression performs on par with or better than the best AutoML or deep learning method on 16% of the datasets, possibly because the latter regularly overfit, especially with smaller sample sizes. On 34% of the datasets, logistic regression performs within 1% of the best AutoML or deep learning approach; on 48% of the datasets within 2% of the best approach; and on 55% of the datasets within 3%. Interestingly, even AutoML methods that consider logistic regression for algorithm selection (i.e., AutoPrognosis) may be outperformed by logistic regression alone (e.g., on the backache dataset), again possibly due to overfitting. From a statistical perspective, Fig. 2 shows that AutoPrognosis and TabPFN achieve a better rank than logistic regression ($p < 0.05$), whereas AutoGluon and HyperFast are not different from a simple logistic regression baseline ($p > 0.05$). It must be noted, though, that TabPFN was meta-trained on 45% of the evaluated datasets (Hollmann et al., 2023). The performance estimates for TabPFN are therefore likely to be overoptimistic. Finally, the best L2-regularization hyperparameters are shown in Table 1 in Appendix A, and they can be readily used for meta-learning, e.g., by leveraging the most similar PMLB dataset(s) for zero- or few-shot hyperparameter optimization (Feurer et al., 2015; Reif et al., 2012).

3.3 Meta-feature analysis

In this section, we extract meta-features from our datasets and compute associations of these meta-features with performance differences between each AutoML / deep learning method and logistic regression. This allows us to find out when more complex machine learning models succeed and derive best practices for method selection.

Meta-feature extraction We extracted meta-features for each dataset using PyMFE 0.4.3 for all available meta-feature groups (Alcobaça et al., 2020), including but not limited to simple meta-features (such as the sample size and EPV (Sect. 2)), statistical meta-features (such as the feature means and standard deviations), clustering meta-features (such as the mean silhouette value and Dunn index), information-theoretic meta-features (such as the feature and class entropies), model-based meta-features (such as the feature importances and number of leaf nodes in a decision tree), as well as complexity and landmarking meta-features (such as the predictive performance of a linear support vector machine and k-nearest neighbors classifier). Some meta-features, such as the feature importance measure, contained multiple values. Therefore, we aggregated these values with all available summary functions in PyMFE, including but not limited to the mean, standard deviation, and frequency in a particular histogram bin, yielding 3932 meta-features in total.

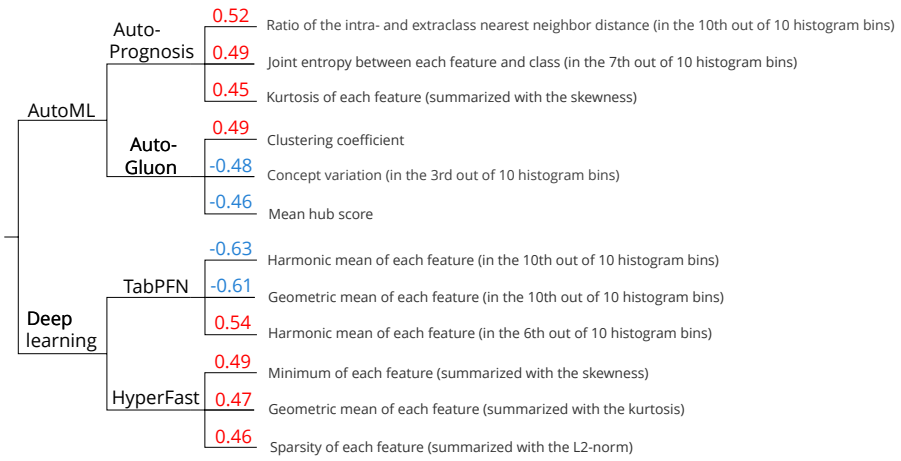

Figure 3: Top-3 meta-features according to their relationship with the mean test AUC differences between each AutoML / deep learning method and logistic regression, based on absolute Spearman rank correlations. Positive correlation coefficients are shown in red, negative coefficients in blue.

Correlation analysis Fig. 3 shows the top-3 correlations of our meta-features with the performance differences between each AutoML / deep learning approach and a logistic regression baseline. The most discriminating meta-feature for AutoPrognosis is the ratio of the intra- and extraclass nearest neighbor distance, for AutoGluon the clustering coefficient. Both complexity measures are larger for harder classification problems (Lorena et al., 2019), and show a positive relationship with the performance differences. AutoML methods therefore appear to be better suited for more complex datasets that need more complex classifiers. For TabPFN and HyperFast, the most discriminating meta-features are summary scores from the feature distribution, i.e., the harmonic mean and minimum of each feature. Pretrained deep neural networks may therefore be better suited for datasets with feature distributions similar to their meta-training datasets. Of note, simple meta-features show much smaller correlations coefficients than those depicted in Fig. 3. The correlation coefficients of the sample size with the performance differences are only -0.09, -0.09, -0.02, and 0.08 for AutoPrognosis, AutoGluon, TabPFN, and HyperFast, respectively; the correlation coefficients of the EPV are only -0.30, -0.36, -0.33, and -0.20. Simple meta-features therefore appear not to be a good guide when to choose more complex machine learning approaches over a simple logistic regression baseline for data-scarce applications.

4 Conclusion

The low-data regime presents challenges for machine learning. In particular, overfitting and suboptimal hyperparameter settings can make it more difficult to achieve a good performance when data is limited. In this work, we evaluated a simple L2-regularized logistic regression baseline against state-of-the-art AutoML and deep learning approaches on 44 (very-)small-sized tabular datasets, and found that the discriminative performance is in fact similar 55% of the time. What is more, simple dataset meta-features like the sample size or EPV are not sufficiently correlated with the performance differences to select the most suitable machine learning approach before actually trying it out (at least when considering a bivariate monotonic correlation measure like we have). For that reason, we recommend not to "squeeze lemons with hammers", but to use logistic regression for data-scarce applications first and only switch to more complex methods if the performance is considered insufficient - giving preference to more transparent, interpretable algorithms if possible. Finally, we present the best L2-regularization hyperparamters that we obtained for each of the 44 datasets, and encourage practitioners to use them for meta-learning in the low-data regime.

Acknowledgments

This research was funded by the Bundesministerium für Bildung und Forschung (16DHBKI071, 01IS23041C).

REFERENCES

Ahmed Alaa and Mihaela van der Schaar. Autoprognosis: Automated clinical prognostic modeling via bayesian optimization with structured kernel learning. In *International conference on machine learning*, pp. 139–148. PMLR, 2018.

Edesio Alcobaça, Felipe Siqueira, Adriano Rivolli, Luís PF Garcia, Jefferson T Oliva, and André CPLF De Carvalho. Mfe: Towards reproducible meta-feature extraction. *The Journal of Machine Learning Research*, 21(1):4503–4507, 2020.

Michele Banko and Eric Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pp. 26–33, 2001.

Alessio Benavoli, Giorgio Corani, and Francesca Mangili. Should we really use post-hoc tests based on mean-ranks? *The Journal of Machine Learning Research*, 17(1):152–161, 2016.

David Bonet, Daniel Mas Montserrat, Xavier Giró-i Nieto, and Alexander G Ioannidis. Hyperfast: Instant classification for tabular data. *arXiv preprint arXiv:2402.14335*, 2024.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Evangelia Christodoulou, Jie Ma, Gary S Collins, Ewout W Steyerberg, Jan Y Verbakel, and Ben Van Calster. A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. *Journal of clinical epidemiology*, 110:12–22, 2019.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, 7(1):1–30, 2006.

Iain Dunning, Joey Huchette, and Miles Lubin. Jump: A modeling language for mathematical optimization. *SIAM review*, 59(2):295–320, 2017.

Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*, 2020.

Benjamin Feuer, Chinmay Hegde, and Niv Cohen. Scaling tabpfn: Sketching and feature selection for tabular prior-data fitted networks. *arXiv preprint arXiv:2311.10609*, 2023.

Matthias Feurer, Jost Springenberg, and Frank Hutter. Initializing bayesian hyperparameter optimization via meta-learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE intelligent systems*, 24(2):8–12, 2009.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

Fergus Imrie, Bogdan Cebere, Eoin F McKinney, and Mihaela van der Schaar. Autoprognosis 2.0: Democratizing diagnostic and prognostic modeling in healthcare with automated machine learning. *PLOS Digital Health*, 2(6):e0000276, 2023.

Ricardo Knauer and Erik Rodner. Cost-sensitive best subset selection for logistic regression: A mixed-integer conic optimization perspective. In *German Conference on Artificial Intelligence (Künstliche Intelligenz)*, pp. 114–129. Springer, 2023.

Ana C Lorena, Luís PF Garcia, Jens Lehmann, Marcilio CP Souto, and Tin Kam Ho. How complex is your classification problem? a survey on measuring classification complexity. *ACM Computing Surveys (CSUR)*, 52(5):1–34, 2019.

Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Ganesh Ramakrishnan, Micah Goldblum, Colin White, et al. When do neural nets outperform boosted trees on tabular data? *arXiv preprint arXiv:2305.02997*, 2023.

Karel GM Moons, Douglas G Altman, Johannes B Reitsma, John PA Ioannidis, Petra Macaskill, Ewout W Steyerberg, Andrew J Vickers, David F Ransohoff, and Gary S Collins. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (tripod): explanation and elaboration. *Annals of internal medicine*, 162(1):W1–W73, 2015.

MOSEK ApS. MOSEK optimizer API for Python. Manual, 2023.

Andreas Müller, Carlo Curino, and Raghu Ramakrishnan. Mothernet: A foundational hypernetwork for tabular classification. *arXiv preprint arXiv:2312.08598*, 2023.

Randal S Olson, William La Cava, Patryk Orzechowski, Ryan J Urbanowicz, and Jason H Moore. Pmlb: a large benchmark suite for machine learning evaluation and comparison. *BioData mining*, 10:1–13, 2017.

Vikram Puri, Aman Kataria, and Bhuvan Puri. Semi-automated diabetes prediction using autogluon and tabpfn models. In *International Conference on Artificial Intelligence of Things*, pp. 289–295. Springer, 2023.

Matthias Reif, Faisal Shafait, and Andreas Dengel. Meta-learning for evolutionary parameter optimization of classifiers. *Machine learning*, 87:357–380, 2012.

Richard D Riley, Kym IE Snell, Glen P Martin, Rebecca Whittle, Lucinda Archer, Matthew Sperrin, and Gary S Collins. Penalization and shrinkage methods produced unreliable clinical prediction models especially when sample size was small. *Journal of Clinical Epidemiology*, 132:88–96, 2021.

Joseph D Romano, Trang T Le, William La Cava, John T Gregg, Daniel J Goldberg, Praneel Chakraborty, Natasha L Ray, Daniel Himmelstein, Weixuan Fu, and Jason H Moore. Pmlb v1. 0: an open-source dataset collection for benchmarking machine learning methods. *Bioinformatics*, 38(3):878–880, 2022.

David Salinas and Nick Erickson. Tabrepo: A large scale repository of tabular model evaluations and its automl applications. *arXiv preprint arXiv:2311.02971*, 2023.

Hana Šinkovec, Georg Heinze, Rok Blagus, and Angelika Geroldinger. To tune or not to tune, a case study of ridge logistic regression in small or sparse datasets. *BMC Medical Research Methodology*, 21:1–15, 2021.

Ewout W Steyerberg. *Clinical prediction models: a practical approach to development, validation, and updating*. Springer, 2019.

Ben Van Calster, Maarten van Smeden, Bavo De Cock, and Ewout W Steyerberg. Regression shrinkage methods for clinical prediction models do not guarantee improved performance: simulation study. *Statistical methods in medical research*, 29(11):3166–3178, 2020.

A APPENDIX

Table 1: Training AUC (mean test AUC) and λ^* across 44 PMLB datasets with sample size M and feature set size N, ordered for sample size ranges in steps of 100.

PMLB dataset	**M**	**N**	**AutoML**		**Deep learning**		**Logistic regression**	λ^*
			Auto-Prognosis	**Auto-Gluon**	**TabPFN**	**HyperFast**		
parity5	32	5	0.50 (0.27)	0.04 (**0.98**)	1.00 (0.02)	1.00 (0.02)	0.50 (0.17)	0.5
analcatdata_fraud	42	11	0.93 (**0.86**)	0.99 (0.68)	1.00 (0.79)	0.99 (0.73)	0.89 (0.77)	0.5
analcatdata_aids	50	4	1.00 (**0.73**)	0.94 (0.67)	1.00 (0.63)	0.80 (0.53)	0.78 (0.61)	0.004
analcatdata_bankruptcy	50	6	1.00 (**0.98**)	1.00 (0.97)	1.00 (0.96)	0.99 (0.88)	0.99 (0.97)	0.004
analcatdata_japansolvent	52	9	1.00 (0.85)	0.99 (0.88)	1.00 (**0.91**)	0.97 (**0.91**)	0.94 (0.85)	0.1
labor	57	16	1.00 (0.88)	1.00 (0.95)	1.00 (**0.99**)	1.00 (0.98)	1.00 (0.97)	0.02
analcatdata_asbestos	83	3	0.87 (**0.87**)	0.89 (0.85)	0.93 (0.85)	0.87 (**0.87**)	0.87 (0.86)	0.5
lupus	87	3	0.92 (0.84)	0.86 (0.77)	0.86 (0.82)	0.83 (0.79)	0.85 (**0.85**)	0.1
postoperative_patient_data	88	8	0.59 (**0.49**)	0.12 (0.46)	0.99 (0.44)	0.87 (0.34)	0.65 (0.38)	0.5
analcatdata_cyyoung9302	92	10	1.00 (**0.89**)	0.99 (0.85)	0.99 (0.87)	0.96 (0.84)	0.94 (0.87)	0.1
analcatdata_cyyoung8092	97	10	0.91 (0.73)	0.99 (**0.87**)	0.98 (0.85)	0.91 (0.84)	0.93 (0.79)	0.1
analcatdata_creditscore	100	6	1.00 (**1.00**)	1.00 (0.99)	1.00 (**1.00**)	0.94 (0.87)	0.97 (0.94)	0.02
median M = 32, ..., 100			0.97 (**0.86**)	0.99 (**0.86**)	1.00 (0.85)	0.95 (0.84)	0.91 (0.85)	
appendicitis	106	7	0.88 (0.78)	0.91 (0.85)	0.97 (0.82)	0.86 (**0.87**)	0.86 (0.84)	0.5
molecular_biology_promoters	106	57	1.00 (0.88)	1.00 (**0.91**)	1.00 (0.88)	1.00 (0.89)	1.00 (0.88)	0.5
analcatdata_boxing1	120	3	0.97 (**0.89**)	0.96 (0.85)	0.99 (0.76)	0.72 (0.67)	0.68 (0.67)	0.5
mux6	128	6	0.50 (**1.00**)	1.00 (**1.00**)	1.00 (**1.00**)	1.00 (0.95)	0.78 (0.70)	0.5
analcatdata_boxing2	132	3	0.92 (**0.82**)	0.91 (0.75)	0.85 (0.71)	0.75 (0.70)	0.70 (0.68)	0.5
hepatitis	155	19	0.87 (**0.85**)	0.99 (0.80)	0.99 (**0.85**)	0.92 (0.83)	0.93 (0.84)	0.5
corral	160	6	1.00 (**1.00**)	1.00 (**1.00**)	1.00 (**1.00**)	1.00 (**1.00**)	0.97 (0.96)	0.1
glass2	163	9	1.00 (0.89)	1.00 (**0.91**)	1.00 (0.89)	0.89 (0.79)	0.81 (0.72)	0.004
backache	180	32	0.90 (0.60)	1.00 (0.71)	1.00 (0.75)	0.93 (**0.78**)	0.90 (0.72)	0.5
prnn_crabs	200	7	1.00 (**1.00**)	1.00 (**1.00**)	1.00 (**1.00**)	0.83 (0.81)	1.00 (**1.00**)	0.004
median M = 101, ..., 200			0.95 (**0.89**)	1.00 (0.88)	1.00 (0.87)	0.91 (0.82)	0.88 (0.78)	
sonar	208	60	1.00 (0.88)	1.00 (**0.92**)	1.00 (**0.92**)	0.95 (0.89)	0.95 (0.85)	0.5
biomed	209	8	1.00 (**1.00**)	1.00 (0.96)	1.00 (0.95)	0.96 (0.93)	0.96 (0.94)	0.004
prnn_synth	250	2	0.98 (0.94)	0.96 (**0.95**)	0.97 (**0.95**)	0.93 (0.94)	0.94 (0.94)	0.02

analcatdata_lawsuit	264	4	1.00 (0.99)	1.00 (0.99)	1.00 (**1.00**)	0.99 (0.98)	1.00 (**1.00**)	0.004
spect	267	22	0.87 (**0.84**)	0.94 (0.81)	0.95 (0.83)	0.90 (0.83)	0.90 (0.82)	0.5
heart_statlog	270	13	0.94 (**0.91**)	0.97 (0.87)	0.98 (0.90)	0.93 (0.89)	0.93 (0.89)	0.5
breast_cancer	286	9	0.76 (0.69)	0.96 (0.67)	0.90 (**0.73**)	0.80 (0.69)	0.73 (0.70)	0.5
heart_h	294	13	0.92 (0.87)	0.96 (0.87)	0.97 (**0.88**)	0.89 (0.85)	0.88 (0.86)	0.5
hungarian	294	13	0.99 (**0.86**)	1.00 (0.85)	0.96 (**0.86**)	0.92 (0.84)	0.89 (0.85)	0.5
median M = 201, ..., 300			0.98 (0.88)	0.97 (0.87)	0.97 (**0.90**)	0.93 (0.89)	0.93 (0.86)	
cleve	303	13	0.96 (**0.90**)	0.98 (**0.90**)	0.99 (0.89)	0.93 (0.88)	0.90 (0.88)	0.5
heart_c	303	13	0.94 (**0.91**)	0.95 (0.90)	0.98 (**0.91**)	0.94 (0.89)	0.92 (**0.91**)	0.5
haberman	306	3	0.86 (0.70)	0.76 (0.71)	0.82 (**0.72**)	0.65 (0.58)	0.70 (0.66)	0.5
bupa	345	5	0.70 (0.66)	0.77 (0.65)	0.72 (**0.68**)	0.72 (0.66)	0.68 (0.67)	0.1
spectf	349	44	1.00 (0.91)	1.00 (**0.94**)	1.00 (0.93)	0.91 (0.87)	0.94 (0.88)	0.1
ionosphere	351	34	1.00 (0.97)	1.00 (**0.98**)	1.00 (**0.98**)	0.96 (0.97)	0.97 (0.90)	0.5
colic	368	22	0.99 (**0.87**)	0.98 (**0.87**)	1.00 (**0.87**)	0.90 (0.86)	0.89 (0.86)	0.5
horse_colic	368	22	0.99 (**0.88**)	0.98 (0.85)	1.00 (0.84)	0.89 (0.83)	0.87 (0.82)	0.5
median M = 301, ..., 400			0.98 (**0.89**)	0.98 (**0.89**)	1.00 (0.88)	0.91 (0.87)	0.90 (0.87)	
house_votes_84	435	16	1.00 (**0.99**)	1.00 (**0.99**)	1.00 (**0.99**)	0.99 (0.98)	0.99 (**0.99**)	0.1
vote	435	16	1.00 (**1.00**)	1.00 (0.99)	1.00 (**1.00**)	0.99 (0.99)	1.00 (0.99)	0.5
saheart	462	9	0.82 (**0.77**)	0.69 (0.75)	0.83 (**0.77**)	0.81 (0.76)	0.79 (**0.77**)	0.5
clean1	476	168	1.00 (0.93)	1.00 (**1.00**)	1.00 (0.99)	0.98 (0.96)	1.00 (**1.00**)	0.004
irish	500	5	1.00 (**1.00**)	1.00 (**1.00**)	1.00 (**1.00**)	0.98 (0.97)	0.85 (0.83)	0.1
median M = 401, ..., 500			1.00 (**0.99**)	1.00 (**0.99**)	1.00 (**0.99**)	0.98 (0.97)	0.99 (**0.99**)	