

Is the Statistical Advantage Worth the Cost? An Empirical Comparison of KANs and MLPs for Structured Data Classification

Matthew Steven P. Toledo^{1,2}, Justine Raphael H. Jacinto^{1,2}, Vivekjeet Singh Chambal^{1,2}, Rodolfo C. Camaclang III^{1,2}, Jamlech Iram N. Gojo Cruz^{1,2}, and Reginald Neil C. Recario^{1,2}

¹ Institute of Computer Science, University of the Philippines Los Baños, Philippines

² Machine Learning and Artificial Intelligence Applications Lab, University of the Philippines Los Baños, Philippines
 {mptoledo2, jhjacio2, vcchambal, rccamaclang1, jngojoecruz, rcrecario}@up.edu.ph

Abstract. This study presents an empirical benchmarking comparison between Kolmogorov-Arnold Networks (KANs) and Multi-Layer Perceptrons (MLPs) on structured tabular classification tasks. Motivated by the growing interest in KANs as an alternative function-approximating architecture, we evaluate their out-of-the-box performance on twelve publicly available datasets spanning binary, multiclass, multilabel, and ordinal problems. Both models were trained under standardized preprocessing, architecture, and fixed hyperparameter settings, with performance assessed using test accuracy and F1-Score, paired hypothesis testing, and effect size analysis. Results show that KANs statistically outperform MLPs in binary and multiclass domains and achieve a significant aggregate advantage across all datasets. However, the observed medium effect size ($d = -0.46$) raises an important cost-benefit consideration: while KANs offer superior generalization through adaptive spline-based mappings, this advantage comes with substantially higher parameter and computational complexity relative to the MLP baseline. These findings suggest KANs are the preferred choice for high-precision applications, while MLPs remain a robust and efficient option for resource-constrained environments. Future work should extend this analysis to additional data modalities to further refine these architectural selection criteria.

Keywords: KAN · MLP · Data Classification · Tabular Data · Benchmarking

1 Introduction

Structured tabular data remains one of the most widely used input formats in production machine learning systems, spanning fields from healthcare to finance. In these settings, accuracy, reliability, and training efficiency are critical, and the choice of model architecture often has a direct influence on practical decisions

[15]. Despite continued advances in deep learning, no single architecture has established itself as clear and consistent best choice for tabular classification [6, 14], and the trade-off between model expressivity and computational cost remains an ongoing concern for practitioners [30].

Recent advances have introduced Kolmogorov-Arnold Networks (KAN) as a theoretically motivated alternative to the multilayer perceptron (MLP), offering a fundamentally different approach to function approximation [22]. While early results have been promising, whether this architectural difference translates to a meaningful and computationally justified performance advantage on real-world tabular tasks remains an open empirical question. Most existing comparisons are limited in scope and vary widely in experimental conditions, making general conclusions difficult [22, 26].

This study addresses that gap by empirically comparing KANs and MLPs across a diverse set of structured tabular classification tasks under standardized experimental settings. In doing so, it not only evaluates raw performance differences but also examines their statistical significance and practical effect size, and weighs these gains against the additional computational cost of KANs. Specifically, the study aims to:

1. Implement and train KAN and MLP models with comparable architectures and fixed, untuned hyperparameters across different classification task types (binary, multiclass, multilabel, and ordinal);
2. Evaluate and compare their performance using test accuracy and F1-Score as the primary metrics;
3. Determine the statistical significance of the observed performance differences between KAN and MLP using paired hypothesis tests; and
4. Analyze the aggregate and per-task trends to weigh the predictive gains against architectural complexity.

2 Background and related works

Structured tabular data remain a dominant input format in machine learning and are widely used in domains such as healthcare, education, finance and remote sensing [7]. These datasets are challenging to model because they often contain heterogeneous feature types, missing values, nonlinear relationships, and limited sample sizes. As a result, traditional decision tree ensemble methods such as Random Forests and Gradient Boosted Trees continue to be the preferred choice for many tabular learning problems since they handle feature heterogeneity and implicit feature interactions effectively with minimal preprocessing requirements [20]. However, while these methods remain competitive, the growing interest in neural architectures that offer expressive modeling, scalability, and integration into deep learning pipelines has renewed attention on feed-forward networks as a practical alternative for tabular tasks.

Among feed-forward architectures, the multilayer perceptron remains the most common and classical architecture [27]. An MLP learns by applying successive linear transformations and fixed nonlinear activation functions across hidden

layers and optimizing parameters through gradient based methods [29, 12]. Its theoretical universality and strong empirical performance have established it as a standard baseline in many studies and benchmarks [19, 13, 2, 32]. However, the use of fixed activation functions and scalar weights can limit the ability of MLPs to capture localized nonlinear effects without increasing network depth or width, which can lead to higher computational cost and tuning complexity [18]. These limitations have driven research toward adaptive activation mechanisms and function-based representations that learn nonlinear transformations directly from data to improve flexibility and interpretability [5, 11].

One such direction is Kolmogorov Arnold Networks, which are inspired by the Kolmogorov-Arnold theorem on multivariate function representation [21, 3]. The theorem demonstrates that any continuous multivariate function can be written as the finite sum of simpler univariate continuous functions, implying that complex high-dimensional relationships can theoretically be constructed from simpler one-dimensional transformations. Each connection in the network corresponds to a small parametric function that often uses spline interpolation to learn how inputs should transform along individual dimensions. KANs replace fixed activations with learnable univariate functions on network connections, which shifts expressivity from node activations to edge-based mappings and enables localized nonlinear modeling. This fundamental architectural departure from the MLP makes a direct empirical comparison between the two a natural and important research question.

Empirical studies have shown that KANs can match or exceed MLP performance on tabular and scientific datasets while offering improved interpretability and faster convergence in some cases [22, 26]. Existing benchmarks vary widely in datasets, hyperparameter choices, and task coverage, which makes general conclusions difficult [22, 26, 9]. Most prior work focuses on binary and multiclass classification, while multilabel and ordinal settings remain largely unexplored [9]. This gap limits understanding of how KANs generalize across classification paradigms and motivates systematic evaluation under standardized conditions to determine whether KANs offer a consistent and cost-justified advantage over MLPs across a broad range of tabular classification tasks, particularly in domains such as healthcare decision support where interpretability and reliability are critical [15, 1].

3 Methods

3.1 Datasets and task categories

This study benchmarks the performance of KANs against conventional MLPs across four classification paradigms: binary, multiclass, multilabel, and ordinal. To ensure a rigorous and reproducible evaluation, twelve datasets were selected from publicly available repositories, including the University of California, Irvine (UCI) Machine Learning Repository, Kaggle, and MULAN library for multilabel learning. These datasets were chosen to represent a diverse cross-section of structured tabular data, encompassing varying sample sizes, feature modalities,

and problem domains. These domains range from healthcare and education to biology and physical sciences. Accordingly, the selected datasets are categorized into four primary task types, as shown in Table 1.

Table 1. Summary of Datasets Used for Evaluation

Task	Dataset	N	Description	Ref.
Binary	Employability (Kaggle)	2,982	Predict student interview outcomes	[17]
	AIDS Clinical (UCI)	2,139	Patient survival prediction	[16]
	Sec. Mushroom (UCI)	61,069	Edible vs. poisonous classification	[35]
Multiclass	Student Dropout (UCI)	4,424	Dropout, enrolled, or graduate	[28, 23]
	Yeast (UCI)	1,484	Protein localization sites	[24]
	Statlog Satellite (UCI)	6,435	Land cover classification	[33]
Multilabel	Emotions (MULAN)	593	Music mood classification	[34]
	Birds (MULAN)	645	Audio-based species detection	[8]
	Enron (MULAN)	1,702	Email text categorization	[10]
Ordinal	Balance Scale (UCI)	625	Psychological balance modeling	[31]
	Car Evaluation (UCI)	1,728	Car acceptability rating	[4]
	Abalone (UCI)	4,177	Age prediction via physical metrics	[25]

3.2 Data preprocessing

A uniform preprocessing pipeline was applied across all datasets. Missing values were handled through listwise deletion or imputation using the mean for numerical and mode for categorical features. Nominal attributes underwent one-hot encoding, ordinal features retained their inherent order, and all numerical features were standardized to zero mean and unit variance. Target variables were processed according to task type, with multilabel targets retained as binary indicator vectors. Datasets were partitioned using stratified sampling into 70% training, 15% validation, and 15% testing sets for standard tasks, and an 80-20 train-test split for multilabel tasks due to the computational complexity of stratifying multi-label data.

3.3 Model configuration and evaluation

To ensure comparability and fairness, each dataset undergoes identical preprocessing, model configuration, and training conditions, with experiments conducted in uniform hardware environments except where computational constraints required deviation. This standardized pipeline was intentionally kept fixed to evaluate each architecture as-is, without tuning, ensuring that any observed differences reflect architectural properties rather than optimization effort. To ensure a meaningful comparison without the confounding variable of tuning, both models utilized "vanilla" configurations established in foundational literature, summarized in Table 2. These configurations isolate the difference between

fixed node-based activations and adaptive spline mappings while maintaining equivalent parameter complexity.

Table 2. Summary of model architecture and training configuration.

Configuration	MLP	KAN
Reference	Gorishniy et al. [13]	Liu et al. [22]
Hidden layers (standard)*	[16, 16]	[16, 16]
Hidden layers (multilabel)*	[128, 64]	[64, 32]
Activation	ReLU	Spline (learnable)
Dropout	$p = 0.2$	—
Grid size / order (standard)	—	Grid 5, $k = 3$
Grid size / order (multilabel)	—	Grid 3, $k = 2$
Optimizer	Adam	Adam
Learning rate	1×10^{-3}	1×10^{-3}
Loss (binary / multilabel)	BCE with Logits	BCE with Logits
Loss (multiclass)	Cross-Entropy	Cross-Entropy
Loss (ordinal)	Ordinal Cross-Entropy	Ordinal Cross-Entropy
Epochs (standard)	100	100
Epochs (multilabel)	20	20
Batch size (large datasets)	256	256

*Input and output dimensions are prepended and appended respectively.

Performance was evaluated using Accuracy, to measure overall predictive correctness, and F1-Score, to provide a balanced assessment of precision and recall, with calculation strategies tailored to each classification paradigm. For binary, multiclass, and ordinal tasks, weighted F1-Score was used to account for class imbalance, while multilabel tasks used subset accuracy and sample-averaged F1-Score. All models underwent five independent execution runs, and average metrics are reported. To confirm that performance differences represent a systematic architectural advantage rather than stochastic noise, paired hypothesis testing was conducted at $\alpha = 0.05$, with test selection governed by the Anderson-Darling normality test. Effect sizes were reported using Cohen’s d for t-tests and rank-biserial correlation (r_{rb}) for the Wilcoxon signed-rank test.

4 Results and discussion

4.1 Computational Efficiency and Resource Requirements

Shown in Table 3 are the summary for the parameter count, training time, and inference time of both KAN and MLP for each dataset. The data consistently put KAN as more computationally "expensive" than MLP.

Table 3. Summary of parameter count and average computational time for MLP and KAN models across 12 benchmark datasets grouped by classification type.

Category Dataset		Parameter Count		Train Time (s)	Inference (ms)	
		MLP	KAN	MLP	KAN	MLP
Binary	Employability	433	7012	0.93	48.28	0.98
	AIDS Clinical Trials	673	11032	0.58	61.41	0.78
	Mushrooms	2209	36760	1.96	628.57	2.77
Multiclass	Dropout	915	15036	0.64	83.35	0.60
	Yeast	586	9352	0.59	55.50	1.19
	Statlog (Satellite)	966	15816	1.41	124.17	1.60
Multilabel	Emotions	17990	75328	1.33	1128.14	0.01
	Birds	42899	212256	1.00	1587.21	0.67
	Enron	139957	745888	3.09	7101.09	0.15
Ordinal	Balance Scale	403	6460	0.55	30.35	0.75
	Car Evaluation	452	7256	0.44	42.25	1.42
	Abalone	501	8052	1.04	57.16	1.73

In terms of architectural complexity, MLP requires only a single scalar weight per connection, resulting in a parameter complexity of $O(N^2L)$, where N is the network width and L is the depth. KAN replaces each weight with a learnable spline function, resulting in a parameter complexity of $O(N^2GL)$ and a computational complexity of $O(2kBN^2GL)$ for a training batch of size B , where G is the grid size and k is the spline order [22]. The additional factor of GL over the MLP in parameter complexity directly explains the substantially higher parameter counts observed in Table 3, where KAN consistently requires roughly 16 times more parameters than MLP across standard tasks.

Furthermore, the computational complexity factor of $2k$, arising from the recursive evaluation of order- k splines, accounts for the disproportionately higher training and inference times: KAN training times exceed 30 seconds across all datasets while MLP training rarely exceeds five seconds, and KAN inference latency frequently exceeds 100 ms while MLP remains 3 ms. Together, these theoretical complexities provide a principled explanation for the computational overhead observed empirically, and directly motivate the cost-benefit analysis presented in Section 4.4.

4.2 Performance Evaluation: Accuracy and F1-Score

Shown in Figure 1 are the training and validation accuracy of both KAN and MLP per dataset. For the majority of the datasets, both have a steady improvement for both training and validation accuracy and reach for similar upper limits, except for *Emotions*, *Birds*, and *Enron* datasets. However, KAN reaches the limits faster and stabilizes earlier than MLP for most of the dataset, except for *Emotions*, *Birds*, and *Enron* datasets.

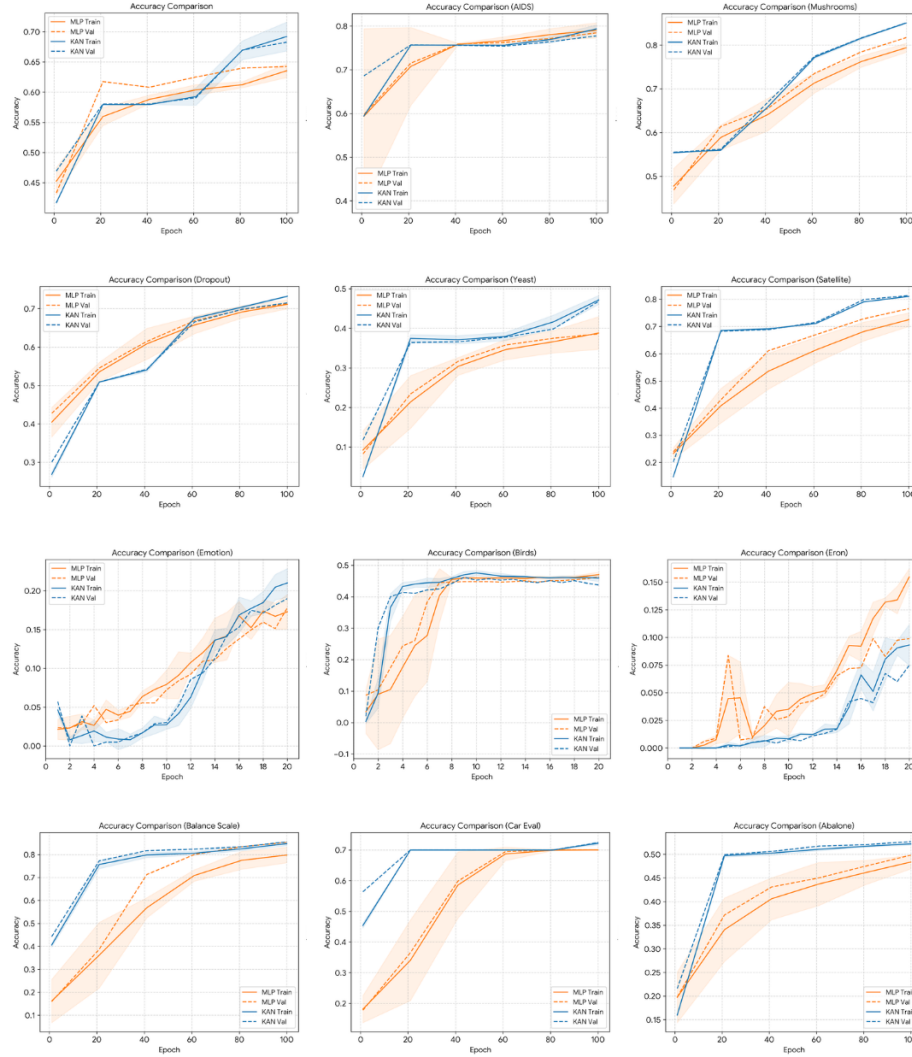


Fig. 1. From Top to Bottom: Training and validation accuracy of KAN and MLP across dataset categories from Binary (top row), Multiclass (middle row), Multilabel (third row), and Ordinal (bottom row) class tasks.

Table 4 presents the evaluation results of the KAN and MLP models across all datasets. In terms of test accuracy, among the three datasets in both binary and multiclass classification, KAN has a higher test accuracy than MLP. For the multilabel classification, KAN has a higher test accuracy than MLP for one dataset only, while MLP is better on the birds and enron datasets. Meanwhile, KAN has a higher test accuracy for two out of three datasets in the ordinal

classification. In terms of F1-Score, KAN is consistently higher than MLP on all datasets under binary and multiclass classification. For multilabel classification, KAN has a higher F1-Score than MLP only on two datasets, except for *Enron* dataset. Lastly, for ordinal classification, KAN has a higher F1-Score on two out of three datasets.

Table 4. Summary of test accuracy and F1-score for MLP and KAN models across 12 benchmark datasets grouped by classification type.

Category	Dataset	Accuracy		F1-Score	
		MLP	KAN	MLP	KAN
Binary	Employability	0.6214	0.6694	0.3482	0.5647
	AIDS Clinical Trials	0.7888	0.7933	0.2887	0.3310
	Mushrooms	0.8177	0.8513	0.8338	0.8653
Multiclass	Dropout	0.7130	0.7154	0.6410	0.6496
	Yeast	0.3946	0.4760	0.3151	0.4210
	Statlog (Satellite)	0.7700	0.8162	0.7140	0.7743
Multilabel	Emotions	0.1782	0.1899	0.3778	0.4373
	Birds	0.4636	0.4372	0.0267	0.0280
	Enron	0.0909	0.0639	0.4750	0.4309
Ordinal	Balance Scale	0.8586	0.8484	0.8246	0.8148
	Car Evaluation	0.7014	0.7185	0.5794	0.6208
	Abalone	0.4917	0.5217	0.4116	0.4700

Overall, in terms of test accuracy, KAN performed better than MLP for 9 out of 12 datasets while MLP performed better than KAN in 3 datasets. In terms of the F1-Score, KAN performed better than MLP for 10 out of 12 datasets, while MLP performed better in the remaining 2 datasets. This means that across the range of the datasets tested, KAN performed better than MLP in the majority of datasets.

4.3 Statistical Significance and Effect Size

While comparing average metrics reveals performance trends, it does not account for the variance inherent in model training. To confirm that the observed differences represent a systematic architectural advantage rather than stochastic noise, we subjected the results to formal hypothesis testing on test accuracy. Table 5 shows the summary of the statistical comparison.

For the analysis per classification type, the statistical tests revealed task-dependent outcomes. Specifically, for Binary and Multiclass datasets, the tests showed p -values less than $\alpha = 0.05$, indicating a significant difference where KAN outperformed MLP. However, for Multilabel and Ordinal tasks, there is no statistically significant difference between the performance of KAN and MLP for

Table 5. Summary of Statistical Comparison of Average Test Accuracy between MLP and KAN Architectures

Category	Stat. Test	Mean Diff. (MLP - KAN)	p -value	Effect Size	Result
Binary	Paired t -Test	-0.029	0.001	$d = -1.09$	KAN Superior
Multiclass	Wilcoxon	-0.039 [†]	0.001	$r_{rb} = -1.00$	KAN Superior
Multilabel	Paired t -Test	+0.014	0.066	$d = +0.51$	No Sig. Diff.
Ordinal	Paired t -Test	-0.012	0.111	$d = -0.44$	No Sig. Diff.
Overall	Paired t-Test	-0.018	0.001	$d = -0.46$	KAN Superior

[†]For Wilcoxon test, value represents the Estimated Median Difference.

these specific metrics. While KAN demonstrates clear superiority in binary and multiclass domains, the two models perform similarly in multilabel and ordinal contexts.

The analysis of effect sizes reveals that KAN demonstrates a large, decisive performance advantage in Binary ($d = -1.09$) and Multiclass ($r_{rb} = -1.00$) tasks. In contrast, the results for Multilabel and Ordinal tasks were less conclusive, showing only medium-to-small effects ($d = 0.51$ favoring MLP and $d = -0.44$ favoring KAN, respectively) that did not reach statistical significance.

In contrast, for the aggregate analysis across all datasets, the test showed that there is a significant difference ($p = 0.001$) between the performance of KAN and MLP in favor of KAN. With a medium effect size ($d = -0.46$), this confirms that KAN generally outperforms MLP across the broad spectrum of classification tasks tested.

4.4 Cost-Benefit Analysis: Accuracy vs. Efficiency

Despite the statistical significance of KAN’s performance advantage, the practical implication of a "medium" effect size ($d = -0.46$) warrants a critical cost-benefit analysis. KAN may have demonstrated a generalized superiority, but the margin of improvement is not transformative enough to think MLPs obsolete, particularly when considering computational constraints. The B -spline computations inherent to KANs typically incur a higher computational overhead compared to the highly optimized matrix multiplications of standard MLPs.

Consequently, the choice between these architectures should be dependent on context. For high-stakes applications where predictive precision is extremely crucial, such as in medical diagnosis or financial forecasting, the statistical advantage of KAN justifies the additional computational cost. However, for environments with resource constraints, such as real-time edge computing or low-latency systems, the "medium" performance gain may not outweigh the efficiency losses. In such scenarios, the MLP remains a robust and computationally economical baseline. Thus, while KAN represents a theoretically superior architecture for

these datasets, its adoption should be driven by the specific tolerance for computational overhead versus the need for marginal accuracy gains.

5 Conclusion

This study presented an empirical comparison between Kolmogorov-Arnold Networks (KANs) and Multi-Layer Perceptrons (MLPs) on structured data classification tasks. Using twelve datasets spanning binary, multiclass, multilabel, and ordinal problems, both models were evaluated in terms of test accuracy and F1-Score under standardized training conditions.

Aggregate results confirm that KAN statistically outperforms MLP, though the performance gap varies by classification type. The magnitude of this improvement, quantified as a medium effect size, poses a pivotal cost-benefit question. The decision to adopt KAN should therefore be driven by the specific operational requirement: prioritize KAN for maximum accuracy, but retain MLP when computational efficiency is the primary constraint.

Beyond empirical results, the findings highlight KAN’s flexibility in adapting to complex tabular patterns through its spline-based functional mappings, which allow it to approximate nonlinear relationships more efficiently. These characteristics point to KAN as a promising direction for structured data modeling, offering a balance between performance and computational efficiency.

Several limitations of this study should be acknowledged. The experiments were conducted under fixed hyperparameters and architectural configurations to fairly compare both models in their default state, which improves comparability but may not fully capture each model’s optimal performance under extensive tuning. Additionally, multilabel datasets were trained for fewer epochs ($i = 20$) than other task types ($i = 100$) due to computational constraints, which may have limited convergence for more complex models such as KAN. Future work should extend this analysis to additional data modalities including text, images, and time series, and may include deeper theoretical analyses and hyperparameter optimization to improve the robustness and interpretability of the results.

References

1. Amann, J., Blasimme, A., Vayena, E., Frey, D., Madai, V.I.: Explainability for artificial intelligence in healthcare: A multidisciplinary perspective. *BMC Medical Informatics and Decision Making* **20**(1), 310 (2020). <https://doi.org/10.1186/s12911-020-01332-6>
2. Arik, S.Ö., Pfister, T.: Tabnet: Attentive interpretable tabular learning. In: *AAAI Conference on Artificial Intelligence*. vol. 35, pp. 6679–6687 (2021). <https://doi.org/10.1609/aaai.v35i8.16826>
3. Arnold, V.I.: On functions of three variables. *Doklady Akademii Nauk SSSR* **114**, 679–681 (1957)
4. Bohanec, M.: Car Evaluation. UCI Machine Learning Repository (1988), DOI: <https://doi.org/10.24432/C5JP48>

5. Bohra, P., Campos, J., Gupta, H., Aziznejad, S., Unser, M.: Learning activation functions in deep (spline) neural networks. *IEEE Open Journal of Signal Processing* **1**, 295–309 (2020). <https://doi.org/10.1109/OJSP.2020.3039379>
6. Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., Kasneci, G.: Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems* **35**(6), 7499–7519 (2022). <https://doi.org/10.1109/TNNLS.2022.3229161>
7. Borisov, V.A., Haug, J., Kasneci, G.: Cancelout: A layer for feature selection in deep neural networks. In: *International Conference on Artificial Neural Networks*. pp. 72–83. Springer (2022). https://doi.org/10.1007/978-3-031-15919-0_7
8. Briggs, F., Huang, Y., Raich, R., Tsoumakas, G., et al.: Birds Dataset. MULAN Multi-Label Dataset Repository (2008), <http://mulan.sourceforge.net/datasets-mlc.html>
9. Cherednichenko, O., Poptsova, M.: Kolmogorov–arnold networks for genomic tasks. *Briefings in Bioinformatics* **26**(2), bbaf129 (03 2025). <https://doi.org/10.1093/bib/bbaf129>, <https://doi.org/10.1093/bib/bbaf129>
10. Cohen, W.W.: Enron Dataset. MULAN Multi-Label Dataset Repository (2015), <http://mulan.sourceforge.net/datasets-mlc.html>
11. Ducotterd, S., Neumayer, S., Goujon, A., Unser, M.: Improving lipschitz-constrained neural networks by learnable linear spline activation functions. *Journal of Machine Learning Research* **25**(291), 1–30 (2024), <https://jmlr.org/papers/volume25/22-1347/22-1347.pdf>
12. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press (2016), <http://www.deeplearningbook.org>
13. Gorishniy, Y., Rubachev, I., Babenko, A.: On embeddings for numerical features in tabular deep learning. In: *Advances in Neural Information Processing Systems*. vol. 35, pp. 24991–25004 (2022), <https://arxiv.org/abs/2203.05556>
14. Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A.: Revisiting deep learning models for tabular data. In: *Advances in Neural Information Processing Systems*. vol. 34, pp. 18932–18943 (2021), <https://arxiv.org/abs/2106.11959>
15. Grinsztajn, L., Oyallon, E., Varoquaux, G.: Why do tree-based models still outperform deep learning on typical tabular data? In: *Advances in Neural Information Processing Systems*. vol. 35, pp. 507–520 (2022), https://proceedings.neurips.cc/paper_files/paper/2022/hash/0378c7692da36807bdec87ab043cdadc-Abstract-Datasets_and_Benchmarks.html
16. Hammer, S.M., Katzenstein, D.A., Hughes, M.D., Gundacker, H., Schooley, R.T., Haubrich, R.H., Henry, W.K., Lederman, M.M., Phair, J.P., Niu, M., Hirsch, M.S., Merigan, T.C.: AIDS Clinical Trials Group Study 175. UCI Machine Learning Repository (2023), <https://archive.ics.uci.edu/dataset/890/aids+clinical+trials+group+study+175>
17. Hamoutni, A.: Students’ employability dataset - philippines. Kaggle (2024), <https://www.kaggle.com/datasets/anashamoutni/students-employability-dataset/data>
18. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 2nd edn. (1998)
19. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. *Neural Networks* **2**(5), 359–366 (1989). [https://doi.org/10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8)

20. Kadra, A., Lindauer, M., Hutter, F., Grabocka, J.: Well-tuned simple nets excel on tabular datasets. In: *Advances in Neural Information Processing Systems*. vol. 34, pp. 23928–23941 (2021), <https://arxiv.org/abs/2106.11189>
21. Kolmogorov, A.N.: On the representation of continuous functions of several variables by superposition of continuous functions of one variable and addition. *Doklady Akademii Nauk SSSR* **114**, 953–956 (1956)
22. Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljacic, M., Hou, T., Tegmark, M.: Kan: Kolmogorov-arnold networks (2025), https://proceedings.iclr.cc/paper_files/paper/2025/file/afaed89642ea100935e39d39a4da602c-Paper-Conference.pdf
23. Martins, M.V., Tolledo, D., Machado, J., Baptista, L.M.T., Realinho, V.: Early prediction of student’s performance in higher education: A case study. In: Rocha, Á., Adeli, H., Dzemyda, G., Moreira, F., Ramalho Correia, A.M. (eds.) *Trends and Applications in Information Systems and Technologies*. pp. 166–175. Springer International Publishing, Cham (2021)
24. Nakai, K.: Yeast. UCI Machine Learning Repository (1991), DOI: <https://doi.org/10.24432/C5KG68>
25. Nash, W., Sellers, T., Talbot, S., Cawthorn, A., Ford, W.: Abalone. UCI Machine Learning Repository (1994), DOI: <https://doi.org/10.24432/C55C7W>
26. Poeta, E., Giobergia, F., Pastor, E., Cerquitelli, T., Baralis, E.: A benchmarking study of kolmogorov-arnold networks on tabular data (2024)
27. Przybyła-Kasperek, M., Marfo, K.F.: A multi-layer perceptron neural network for varied conditional attributes in tabular dispersed data. *PLOS One* **19**(12), e0311041 (2024). <https://doi.org/10.1371/journal.pone.0311041>
28. Realinho, V., Vieira Martins, M., Machado, J., Baptista, L.: Predict Students’ Dropout and Academic Success. UCI Machine Learning Repository (2021), DOI: <https://doi.org/10.24432/C5MC89>
29. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. *Nature* **323**(6088), 533–536 (1986). <https://doi.org/10.1038/323533a0>
30. Shwartz-Ziv, R., Armon, A.: Tabular data: Deep learning is not all you need. *Information Fusion* **81**, 84–90 (2022). <https://doi.org/10.1016/j.inffus.2022.01.006>
31. Siegler, R.: Balance Scale. UCI Machine Learning Repository (1976), DOI: <https://doi.org/10.24432/C5488X>
32. Somepalli, G., Goldblum, M., Schwarzschild, A., Bruss, C.B., Goldstein, T.: Saint: Improved neural networks for tabular data via row attention and contrastive pre-training. *arXiv preprint arXiv:2106.01342* (2021), <https://arxiv.org/abs/2106.01342>
33. Srinivasan, A.: Statlog (Landsat Satellite). UCI Machine Learning Repository (1993), DOI: <https://doi.org/10.24432/C55887>
34. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.: Emotions Dataset. MULAN Multi-Label Dataset Repository (2013), <http://mulan.sourceforge.net/datasets-mlc.html>
35. Wagner, D., Heider, D., Hattab, G.: Secondary Mushroom. UCI Machine Learning Repository (2021), DOI: <https://doi.org/10.24432/C5FP5Q>