Tabular Foundational Models: An Overview

David Salinas. TU Berlin. December 2025.

Why Tabular Data Matters

Why Tabular Data Matters

- Tabular data is the most prevalent format in real-world ML applications

Why Tabular Data Matters

- Tabular data is the most prevalent format in real-world ML applications
  - Healthcare records, financial transactions, customer databases, …

Why Tabular Data Matters

- Tabular data is the most prevalent format in real-world ML applications
  - Healthcare records, financial transactions, customer databases, …
  - Many applications: fraud detection, predicting demand, credit scoring, …

Why Tabular Data Matters

- Tabular data is the most prevalent format in real-world ML applications
  - Healthcare records, financial transactions, customer databases, …
  - Many applications: fraud detection, predicting demand, credit scoring, …
  - Large portion of ML model deployed in industry

Why Tabular Data Matters

- Tabular data is the most prevalent format in real-world ML applications
  - Healthcare records, financial transactions, customer databases, …
  - Many applications: fraud detection, predicting demand, credit scoring, …
  - Large portion of ML model deployed in industry
- State of the art dominated by gradient boosted decision trees for many years

Why Tabular Data Matters

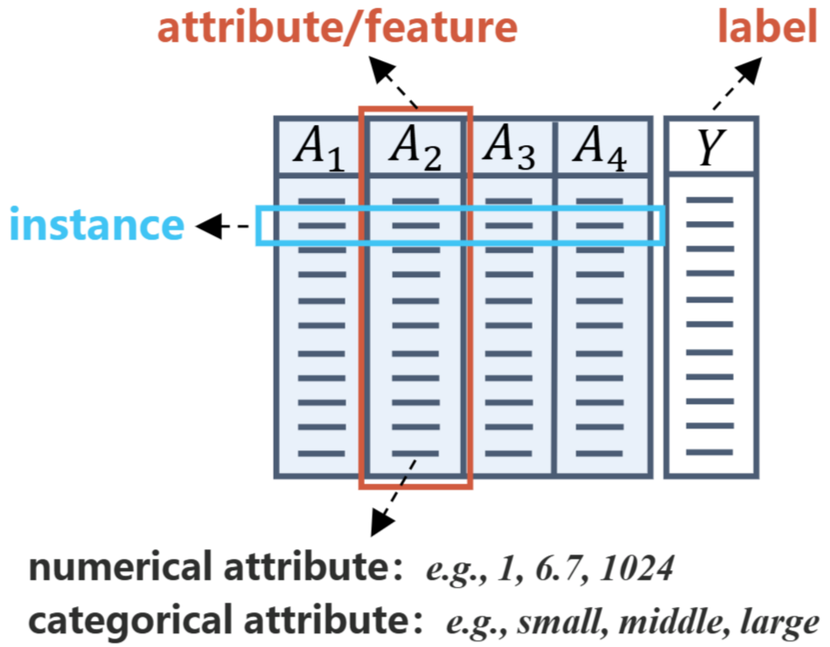
- Tabular data is the most prevalent format in real-world ML applications
  - Healthcare records, financial transactions, customer databases, …
  - Many applications: fraud detection, predicting demand, credit scoring, …
  - Large portion of ML model deployed in industry
- State of the art dominated by gradient boosted decision trees for many years
  - XGBoost, LightGBM, CatBoost became the default choice

Why Tabular Data Matters

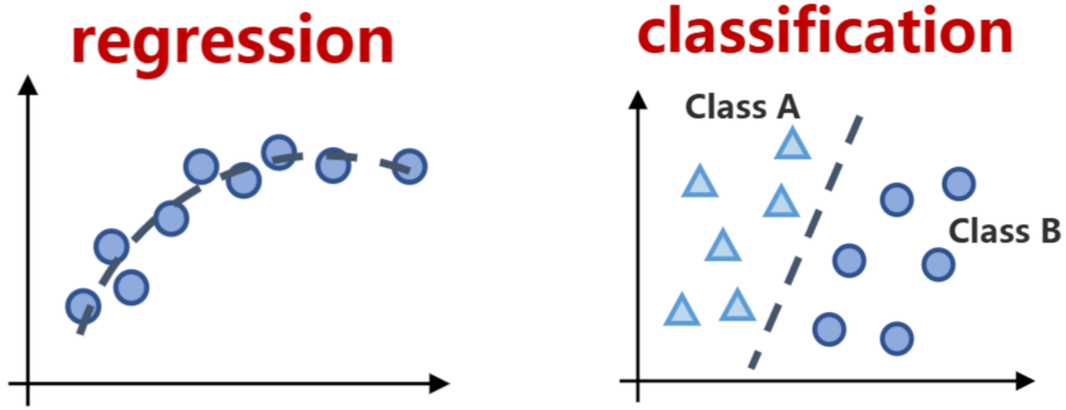
- Tabular data is the most prevalent format in real-world ML applications
  - Healthcare records, financial transactions, customer databases, …
  - Many applications: fraud detection, predicting demand, credit scoring, …
  - Large portion of ML model deployed in industry
- State of the art dominated by gradient boosted decision trees for many years
  - XGBoost, LightGBM, CatBoost became the default choice
  - Consistently outperformed neural approaches across benchmarks

Tabular prediction

- Input: a training data frame, a target column and a training time budget
- Output: a predictor able to give predictions given a test dataframe
- Metrics:
  - RMSE (regression), log-prob (classification)
  - Prediction latency, memory, …




```python
import pandas as pd
from autogluon.tabular import TabularPredictor

df_train = pd.read_csv('train.csv')
df_test = pd.read_csv('train.csv')

predictor = TabularPredictor(label='class').fit(df_train)
predictions = predictor.predict(df_test)
```



Figures from https://arxiv.org/pdf/2504.16109

This talk

Menu du jour

- Entrée: introduction to Tabular PFNs
- Menu: EquiTabPFN, dealing with the lack of equivariance of PFNs
- Dessert: benchmarking tabular models

Tabular PFNs

Deep-Learning + Tabular = ❤️?

- Many attempts to bring deep learning to tabular domains
  - Typically transformer-based methods trained on single datasets
  - TabTransformer, FT-Transformer, SAINT and others
- These approaches generally failed to outperform boosted trees
  - At best, *matched* performance of tree-based methods
  - Required more computation and careful tuning
- No clear advantage to justify the added complexity
  - Tabular Data: Deep Learning is Not All You Need [Shwartz-Ziv 2021]
  - Why do tree-based models still outperform deep learning on tabular data? [Grinsztajn 2022]

TabPFN – A Paradigm Shift

- TabPFN marked a significant departure from previous approaches
  - First foundational model for tabular data that works
- Key innovation:
  - 1. Train on **synthetic data**
    - Solve data scarcity => can fit model on 100s millions of synthetic datasets
  - 2. Fit and predict in a **single forward pass** with In Context Learning (ICL)
    - No iterative training loop at inference time
    - Provide training data and test points as input → model outputs predictions directly
- Substantially outperforms boosted trees on small/medium datasets, even full blown **AutoML systems**
- Challenge becomes designing the *prior*, not the *algorithm* (the name Priorlabs indicates this)

PFN – How Does It Work?

- Training happens on synthetic datasets, not real data:
  - Sample tabular datasets $(X_{train}, y_{train}, X_{test}, y_{test}) \sim p(\mathscr{D})$
- A transformer trained to predict the posterior predictive distribution directly:
  - Estimate $p(y_{test} \mid X_{test}, X_{train}, y_{train})$ with encoder-decoder architecture
- The distribution $p(\mathscr{D})$ is carefully engineered to resemble real-world tabular data
- Model learns Bayesian inference by observing millions of synthetic learning problems

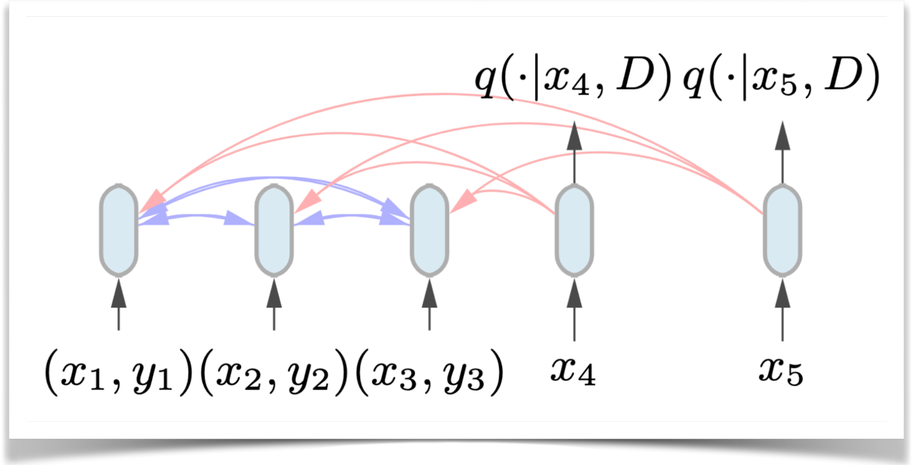
Architecture

Architecture

- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}

Architecture

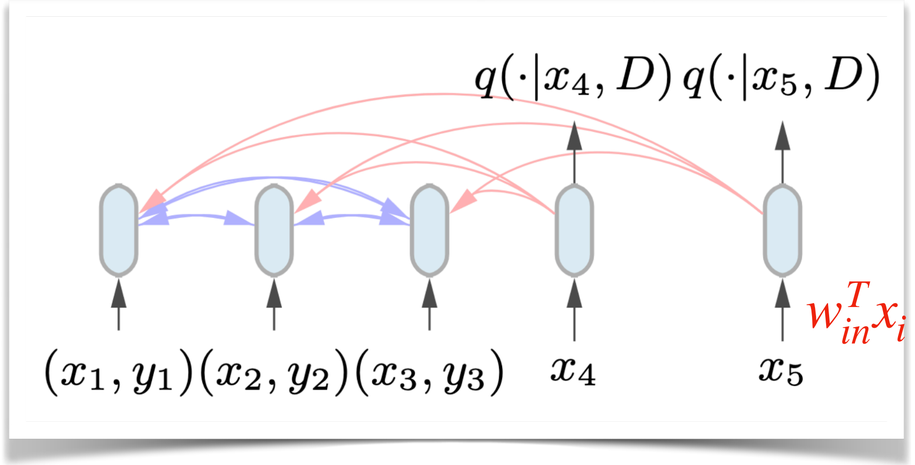
- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}



TabPFN-v1 architecture

Architecture

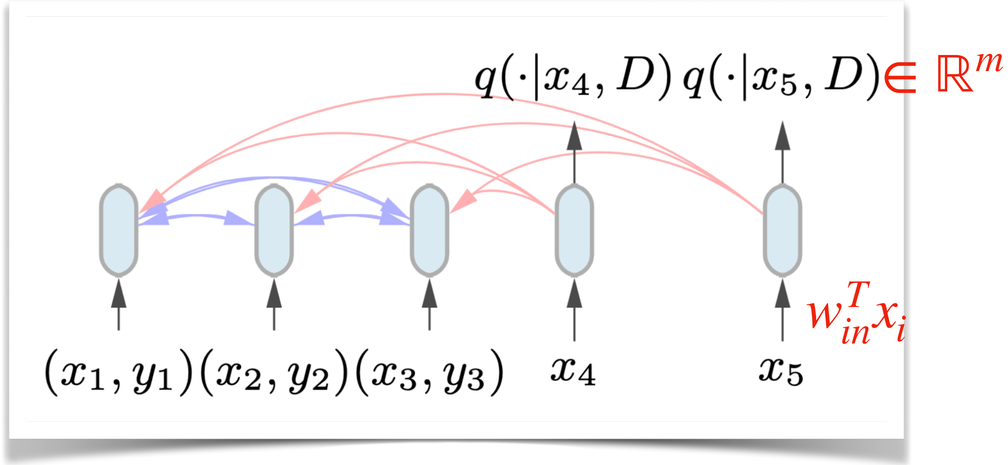
- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}



TabPFN-v1 architecture

Architecture

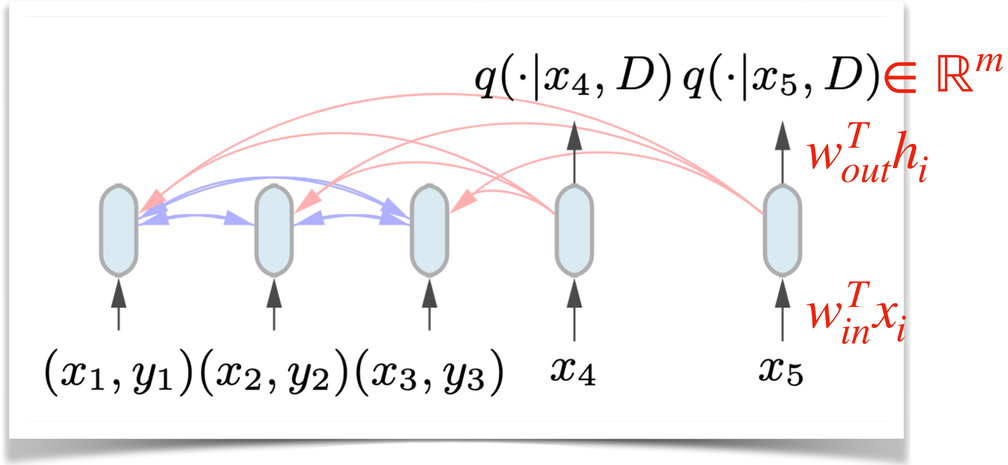
- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}



TabPFN-v1 architecture

Architecture

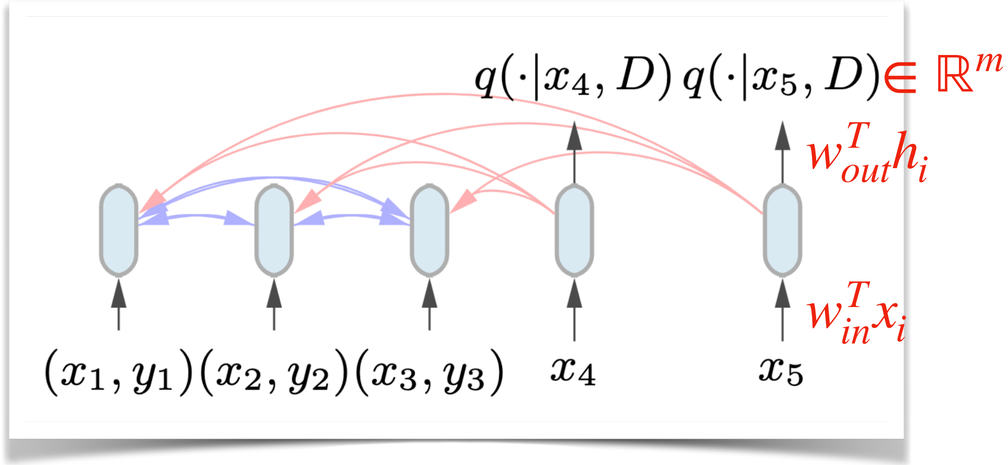
- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}



TabPFN-v1 architecture

Architecture

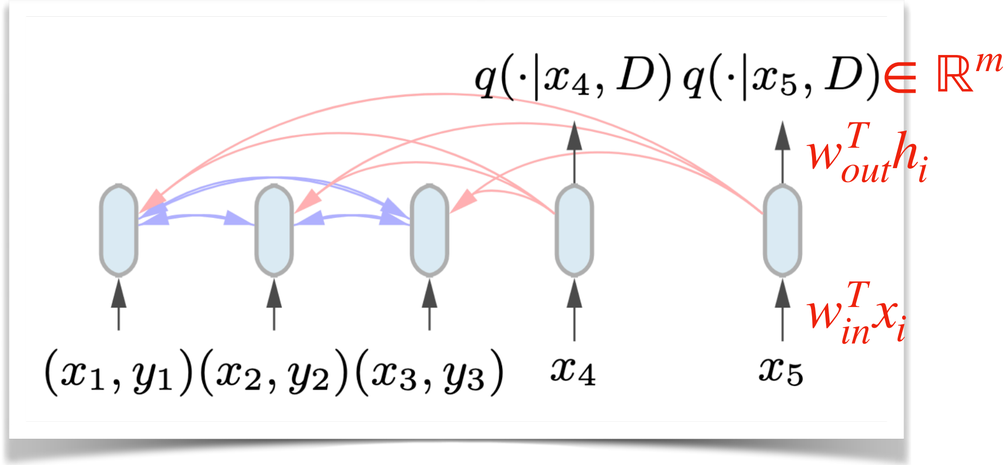
- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}
- Model is trained only up to a number of features and target dimension (with zero-padding)



TabPFN-v1 architecture

Architecture

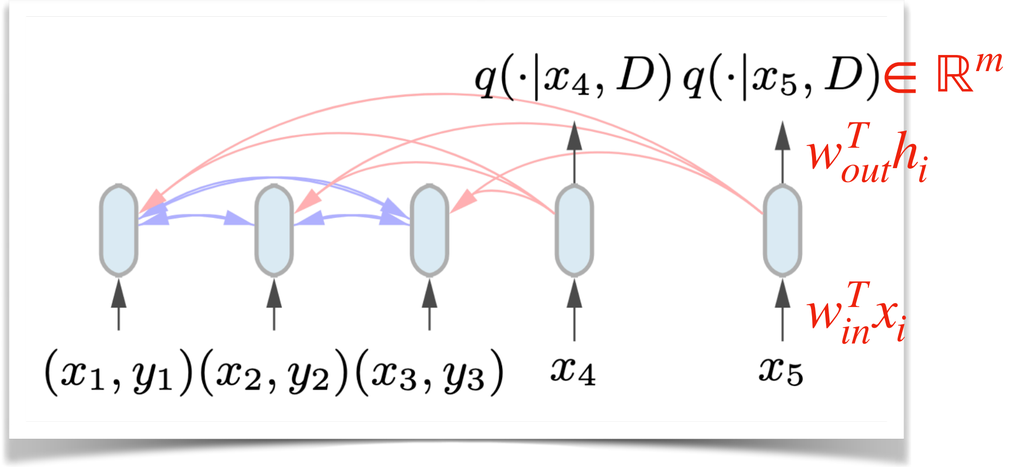
- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}
- Model is trained only up to a number of features and target dimension (with zero-padding)
- Cannot perform inference on number of features/classes not seen!



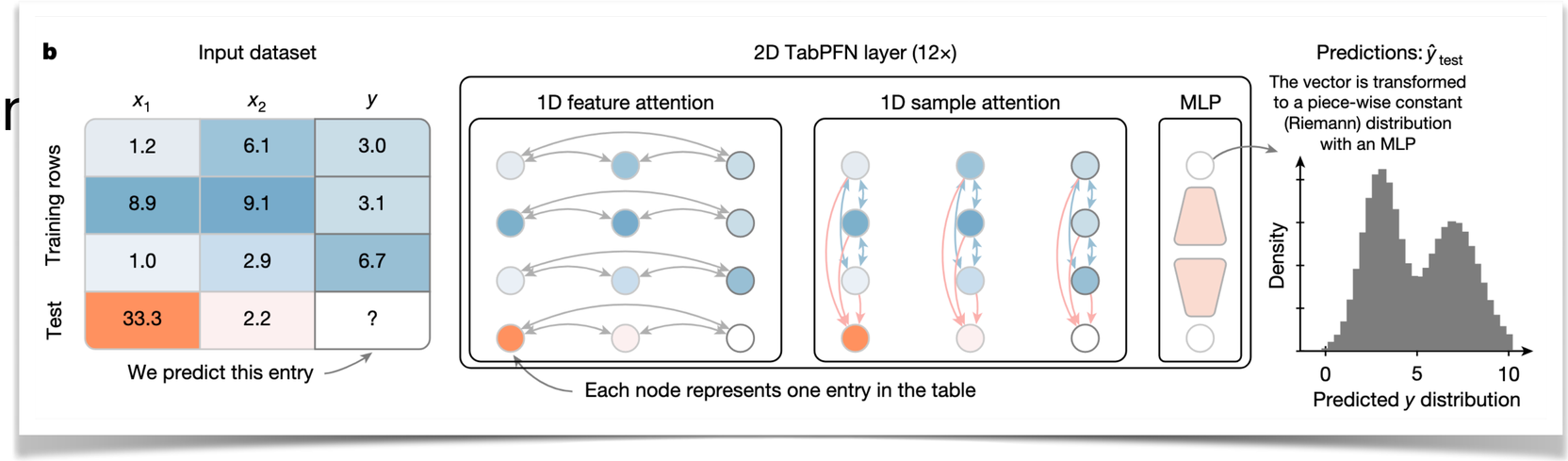
TabPFN-v1 architecture

Architecture

- Encode X_{train}, y_{train} with self-attention, decode X_{test} by attending on the contextualized tokens of X_{train}, y_{train}
- Model is trained only up to a number of features and target dimension (with zero-padding)
- Cannot perform inference or features/classes not seen!



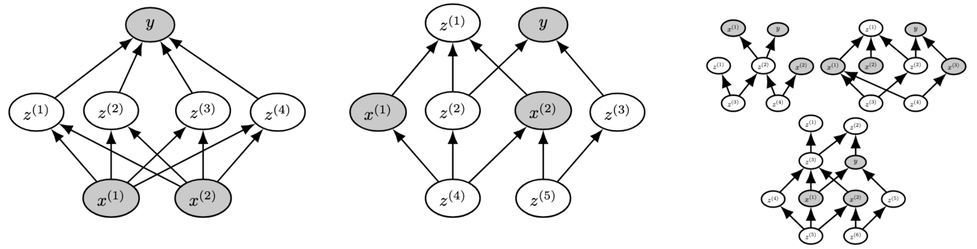
TabPFN-v1 architecture



TabPFN-v2 architecture

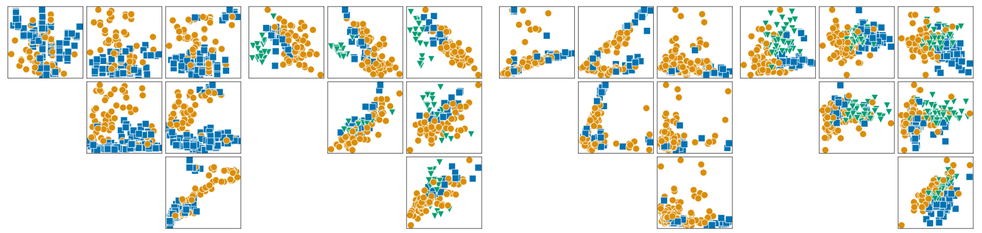
The Prior – Structural Causal Models

- The prior $p(\mathscr{D})$ is the **heart** of what makes PFNs work
- TabPFN uses Structural Causal Models (SCMs) to generate synthetic data:
  - SCM defines a directed acyclic graph
  - Each variable is a function of its parents
  - Nodes outputs are random MLPs
- Creates diverse synthetic datasets with realistic feature interactions



(a) A BNN (b) An SCM (c) SCMs sampled from the prior

Figure 2: Overview of graphs generating data in our prior. Inputs x are mapped to the output y through unobserved nodes z. Plots based on Müller et al. (2022).

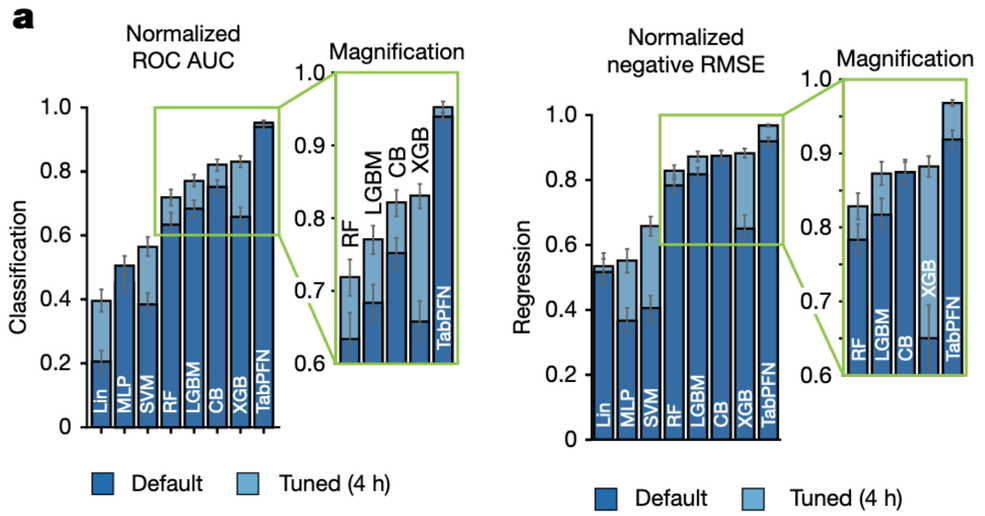


(a) Synthetic datasets (b) Actual datasets

Figure 3: Each point represents a sample, each sub-plot shows the value of two features for each sample, color indicates the class label. (a) Two synthetic datasets generated by our causal tabular data prior. Numeric SCM outputs are mapped to classes as described in Section 4.5. (b) Two datasets from our validation datasets: Parkinsons (Left) and Wine (Right).

Results

- TabPFN-v1: decent results on a small number of datasets against toyish baselines
- TabPFN-v2: outperforms other methods on small datasets (up to 10,000 samples)
- TabPFN-v2.5: outperforms **SOTA AutoML system** (at time of publication) on medium sized datasets



TabPFN-v2

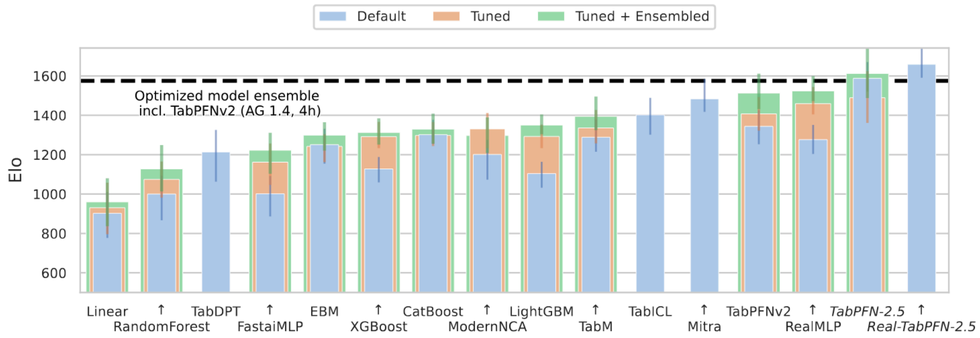


Figure 1: TabPFN-2.5 performance on the standard TabArena-lite benchmark [1], TabPFNv2 classification subset. TabPFN-2.5 outperforms any other model in a forward pass, and marks a strong leap from TabPFNv2. When fine-tuned on real data, Real-TabPFN-2.5 shows even stronger performance. The horizontal dotted line stands for AutoGluon 1.4 extreme mode tuned for 4 hours, an ensemble of models including TabPFNv2.

TabPFN-v2.5

You said PFN?

You said PFN?

- Many followup works!

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …

You said PFN?

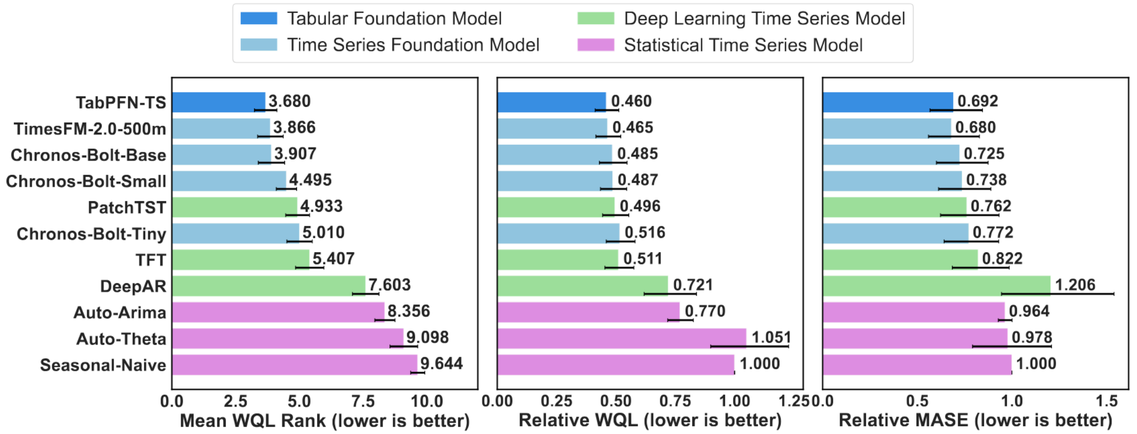
- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)

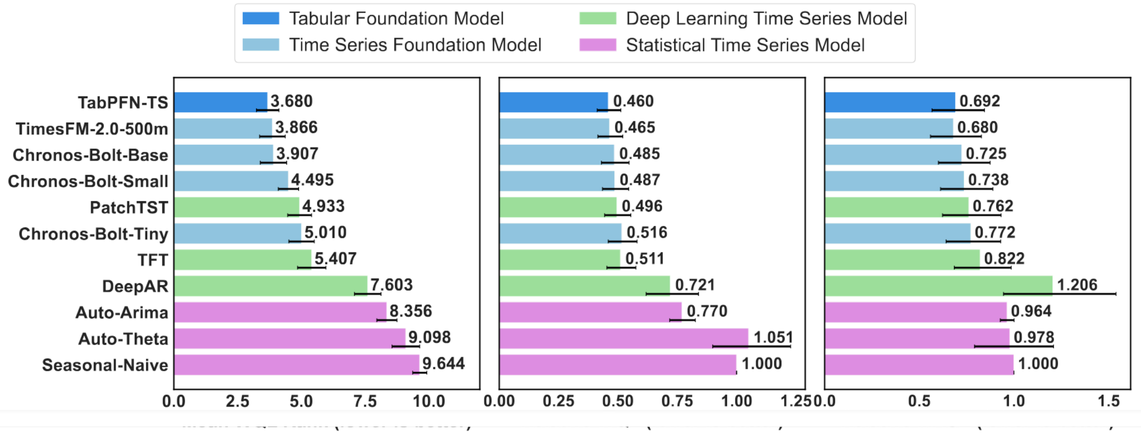
You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)



You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)

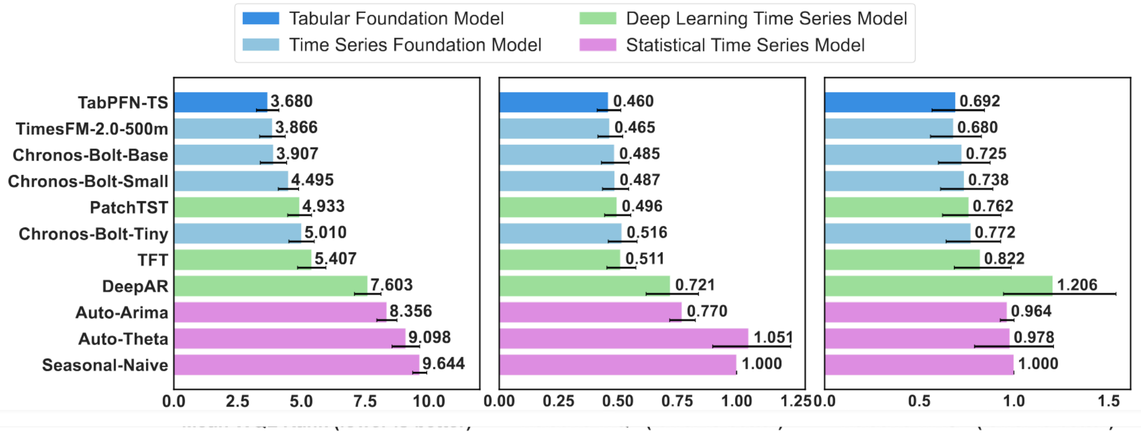


TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

Not the case anymore, also it was quite slow

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)



TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

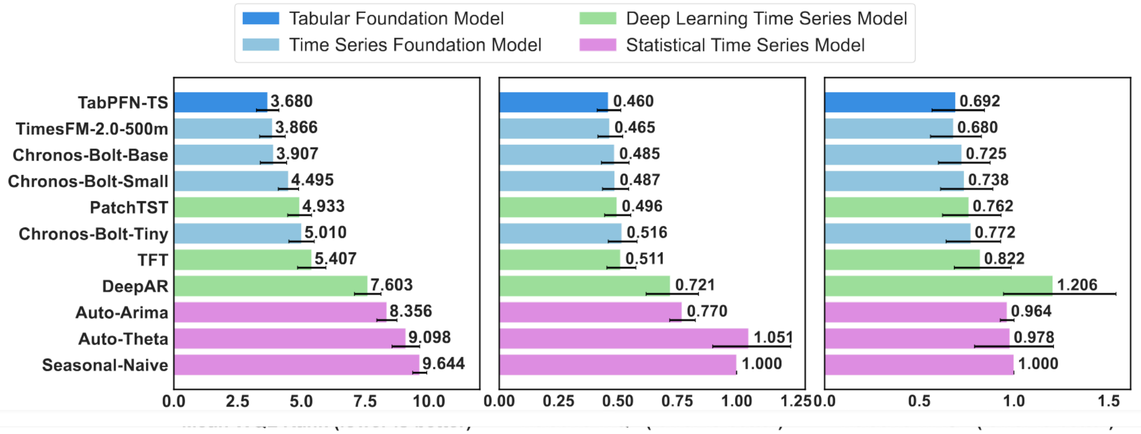
Not the case anymore, also it was quite slow

Model	# of Params.
Chronos-Bolt-Tiny	9M
TabPFN-TS	11M
Chronos-Bolt-Small	48M
Chronos-Bolt-Base	205M
TimesFM-2.0	500M

Table 1: Model size comparison of various time series foundation models. TabPFN-TS is among the smaller models, with a similar size to Chronos-Bolt-Tiny

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)
  - Statistics (Zhang 2025)



TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

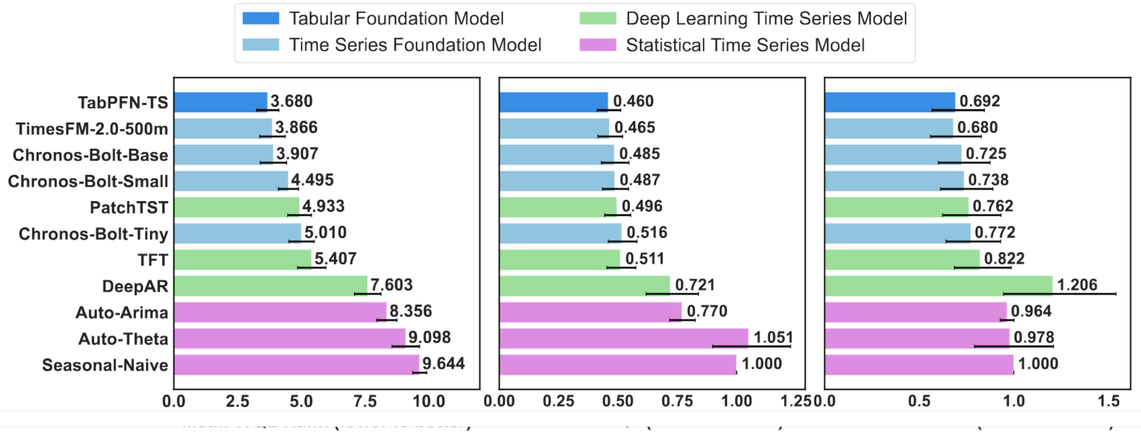
Not the case anymore, also it was quite slow

Model	# of Params.
Chronos-Bolt-Tiny	9M
TabPFN-TS	11M
Chronos-Bolt-Small	48M
Chronos-Bolt-Base	205M
TimesFM-2.0	500M

Table 1: Model size comparison of various time series foundation models. TabPFN-TS is among the smaller models, with a similar size to Chronos-Bolt-Tiny

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)
  - Statistics (Zhang 2025)
  - Causal discovery (Robertson 2025)



TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

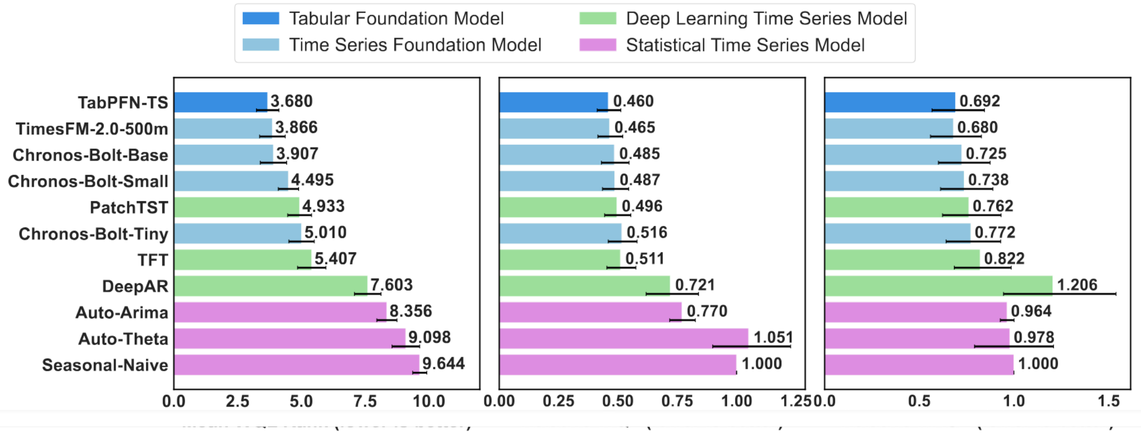
Not the case anymore, also it was quite slow

Model	# of Params.
Chronos-Bolt-Tiny	9M
TabPFN-TS	11M
Chronos-Bolt-Small	48M
Chronos-Bolt-Base	205M
TimesFM-2.0	500M

Table 1: Model size comparison of various time series foundation models. TabPFN-TS is among the smaller models, with a similar size to Chronos-Bolt-Tiny

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)
  - Statistics (Zhang 2025)
  - Causal discovery (Robertson 2025)
  - Hyperparameter optimization (Muller 2023)



TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

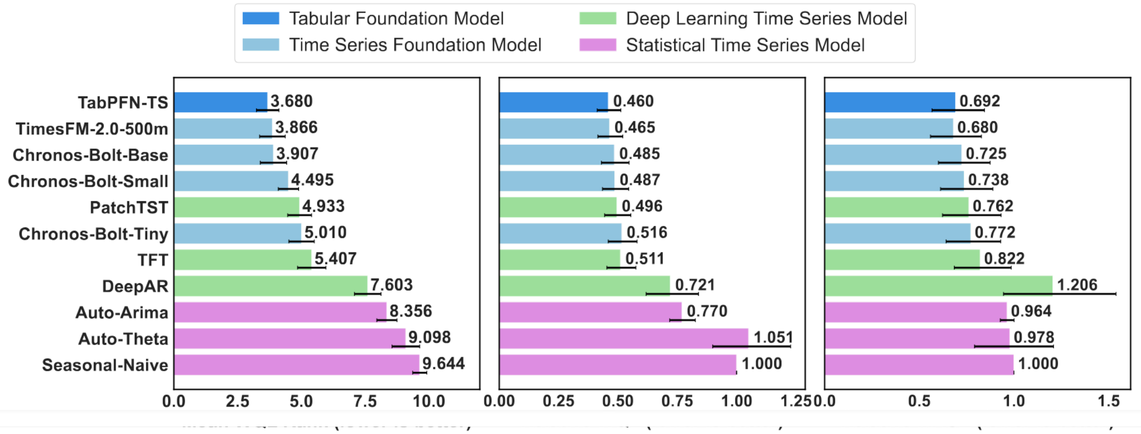
Not the case anymore, also it was quite slow

Model	# of Params.
Chronos-Bolt-Tiny	9M
TabPFN-TS	11M
Chronos-Bolt-Small	48M
Chronos-Bolt-Base	205M
TimesFM-2.0	500M

Table 1: Model size comparison of various time series foundation models. TabPFN-TS is among the smaller models, with a similar size to Chronos-Bolt-Tiny

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)
  - Statistics (Zhang 2025)
  - Causal discovery (Robertson 2025)
  - Hyperparameter optimization (Muller 2023)
  - …



TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

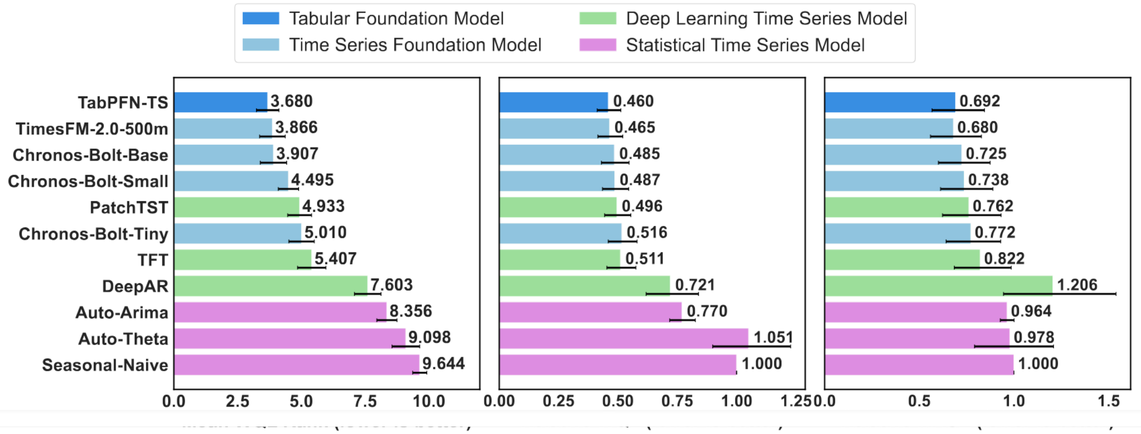
Not the case anymore, also it was quite slow

Model	# of Params.
Chronos-Bolt-Tiny	9M
TabPFN-TS	11M
Chronos-Bolt-Small	48M
Chronos-Bolt-Base	205M
TimesFM-2.0	500M

Table 1: Model size comparison of various time series foundation models. TabPFN-TS is among the smaller models, with a similar size to Chronos-Bolt-Tiny

You said PFN?

- Many followup works!
- Mothernet, Gamformer, TabForest, TabDPT, TabICL, ContextTab, Mitra, EquiTabPFN, …
- Some applications:
  - Time-series (TabPFN-TS Shi-Bin-Hoo 2025)
  - Statistics (Zhang 2025)
  - Causal discovery (Robertson 2025)
  - Hyperparameter optimization (Muller 2023)
  - …
  - From Benchmarks to Problems - A Perspective on Problem Finding in AI (Kyunghyun Cho - NeurIPS invited talk 2025)



TabPFN-TS was the top method on gift-eval for several month outperforming foundational time-series method 🤯

Not the case anymore, also it was quite slow

Model	# of Params.
Chronos-Bolt-Tiny	9M
TabPFN-TS	11M
Chronos-Bolt-Small	48M
Chronos-Bolt-Base	205M
TimesFM-2.0	500M

Table 1: Model size comparison of various time series foundation models. TabPFN-TS is among the smaller models, with a similar size to Chronos-Bolt-Tiny

EquiTabPFN, dealing with the lack of equivariance of PFNs

EquiTabPFN: A Target-Permutation Equivariant Prior Fitted Network

NeurIPS 2025

Michael Arbel[*1] **David Salinas**[*2,3] **Frank Hutter**[2,3,4]
[1]INRIA [2]University of Freiburg [3]ELLIS Institute Tübingen [4]PriorLabs
[*]Equal contribution

Michael Arbel[*1]

David Salinas[*2,3]

Frank Hutter[2,3,4]

EquiTabPFN

Did you say equivariant?

- In tabular tasks, the ordering of target components is **arbitrary**
- Models should give identical predictions under any permutation of target!

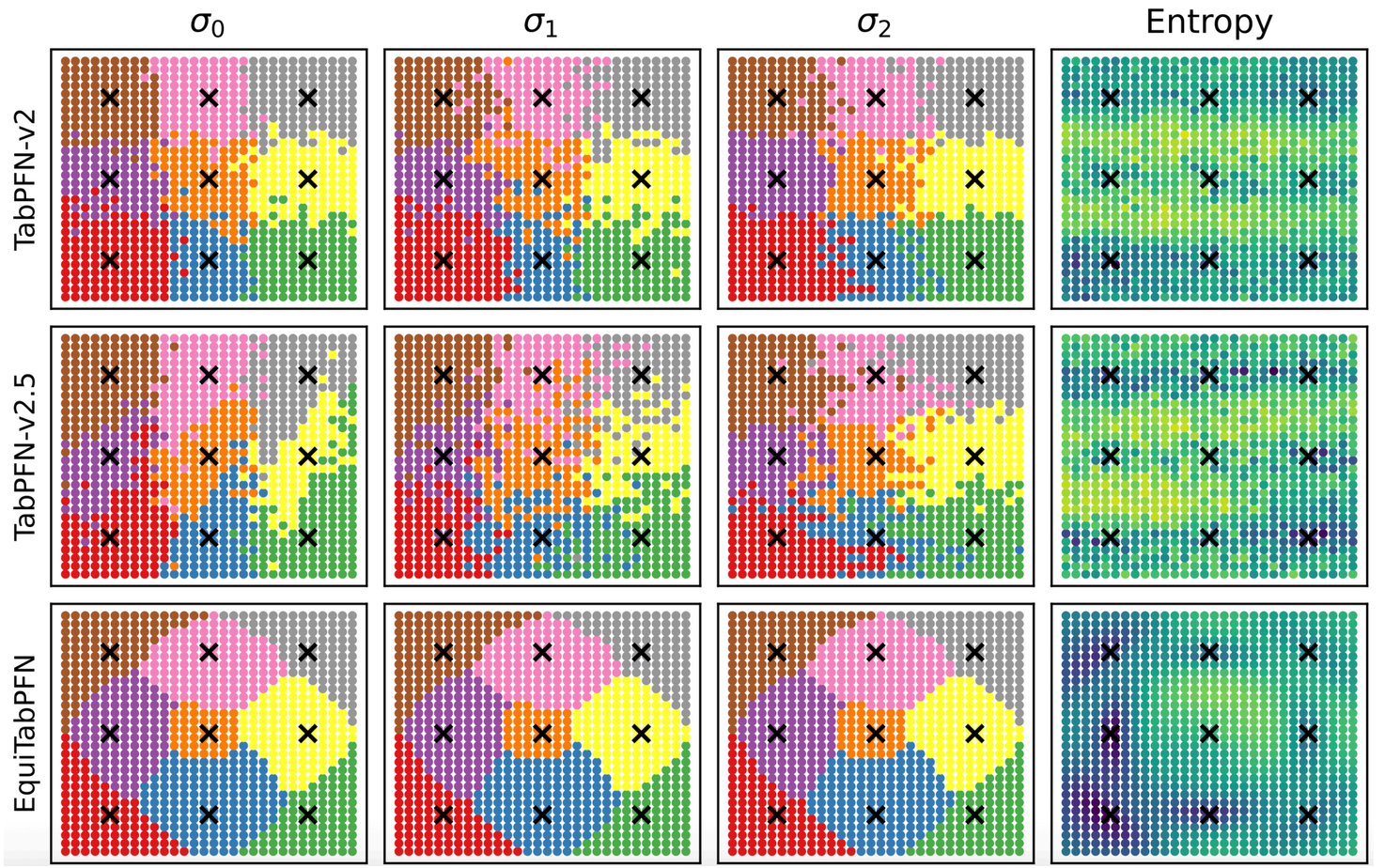
Definition

Denote $Y^* = f_{X,Y}(X^*)$ the predictions of a PFN on test features X^* given a training dataset X, Y.

A PFN is *target-equivariant* if $\sigma(Y^*) = f_{X,\sigma(Y)}(X^*)$ for all permutations σ

Prediction instabilities

- Training sets with 9 examples, each own class, features in $\mathbb{R}^2$
- Shows predictions of $\sigma^{-1}(f_{X,\sigma(Y)}(X^*))$
- Should be identical for different σ!



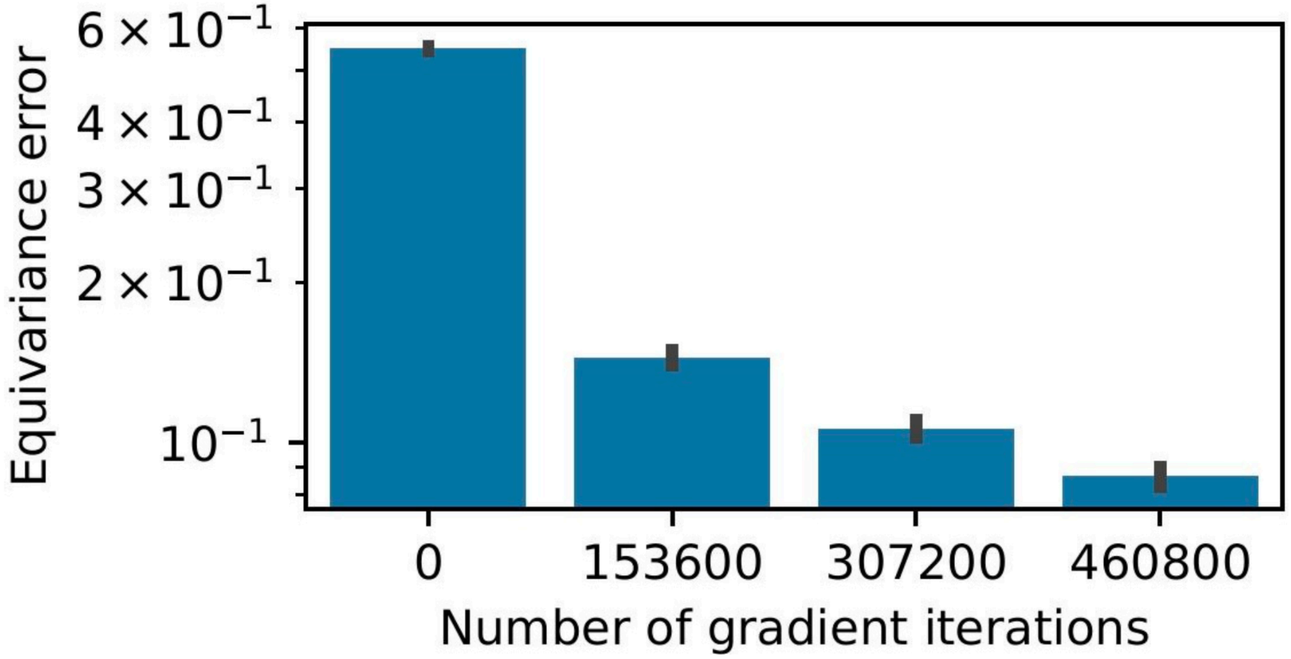
Predictions stable w.r.t. target class encoding

Entropy and probabilities smooth w.r.t. distance to features

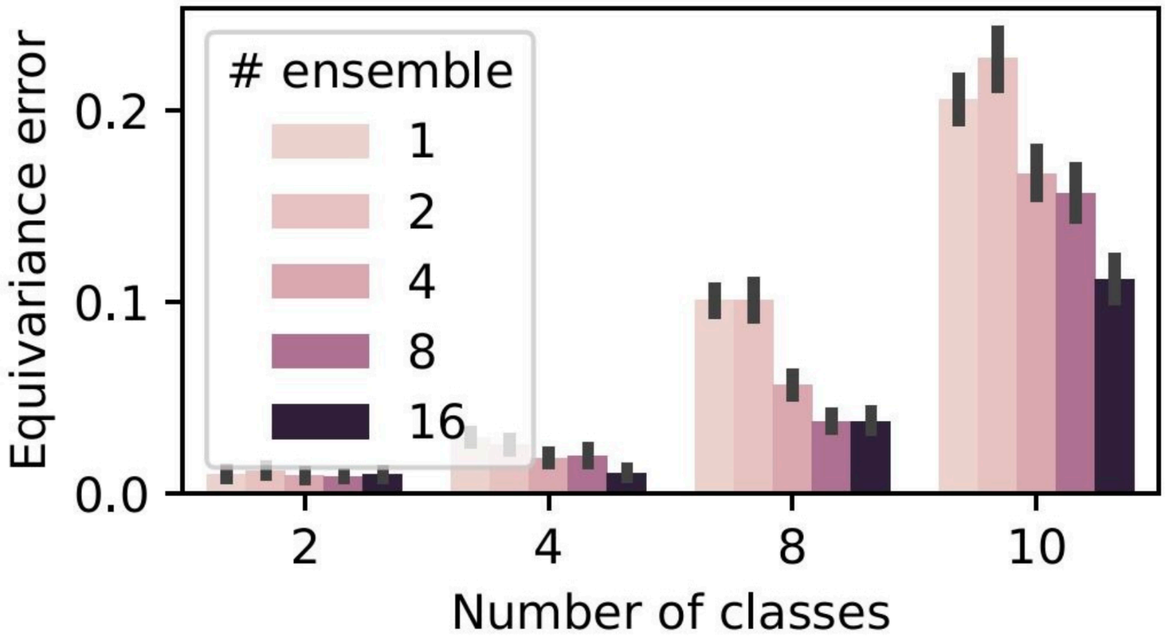
Can you just train longer? Or just ensemble more?

No…

Would get to zero with $\mathcal{O}(m!)$ ensembles



Equivariance error while training



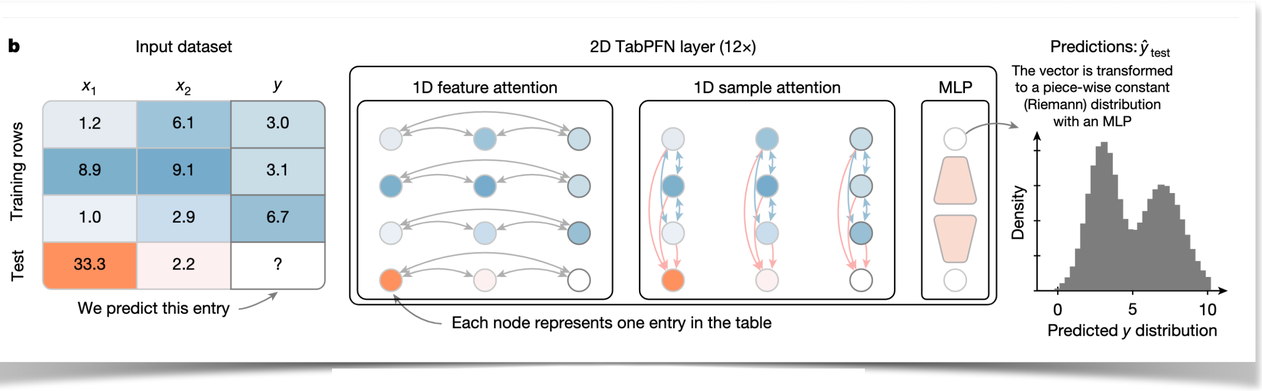
Equivariance error when ensembling

The cost of not being equivariant

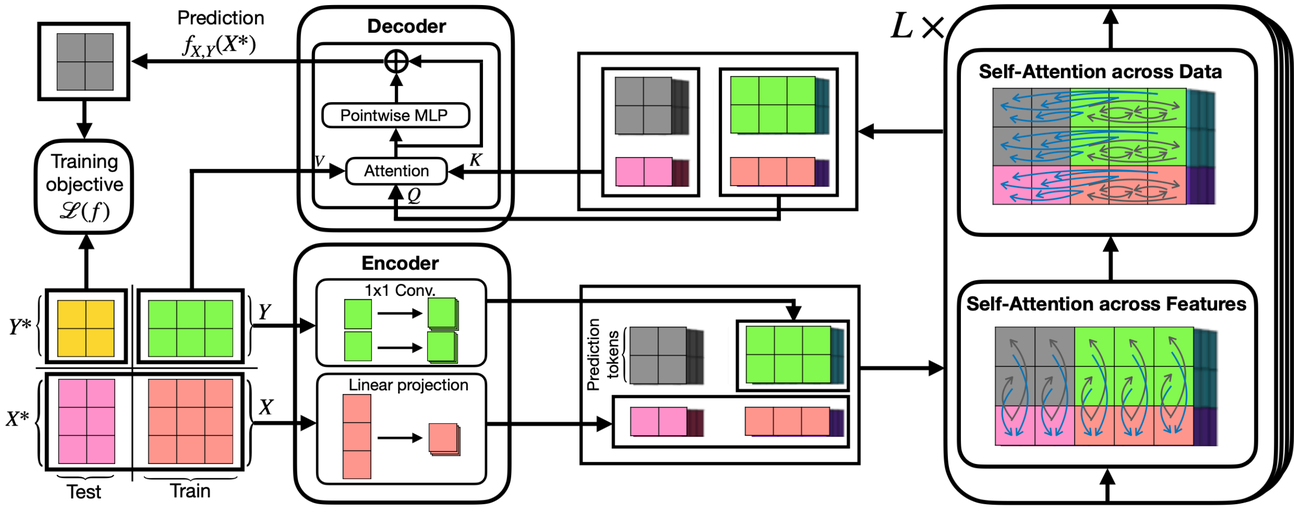
Proposition
Under mild assumptions—convex, permutation-invariant loss ℓ and a target permutations-invariant data distribution—the optimal solution to the PFN pre-training objective is necessarily target equivariant.

Proposed Architecture

- Alternate attention over rows and target dimensions
- Equivariant to target permutations
- Handle any number of target
- Output obtained by weighting input labels by similarity



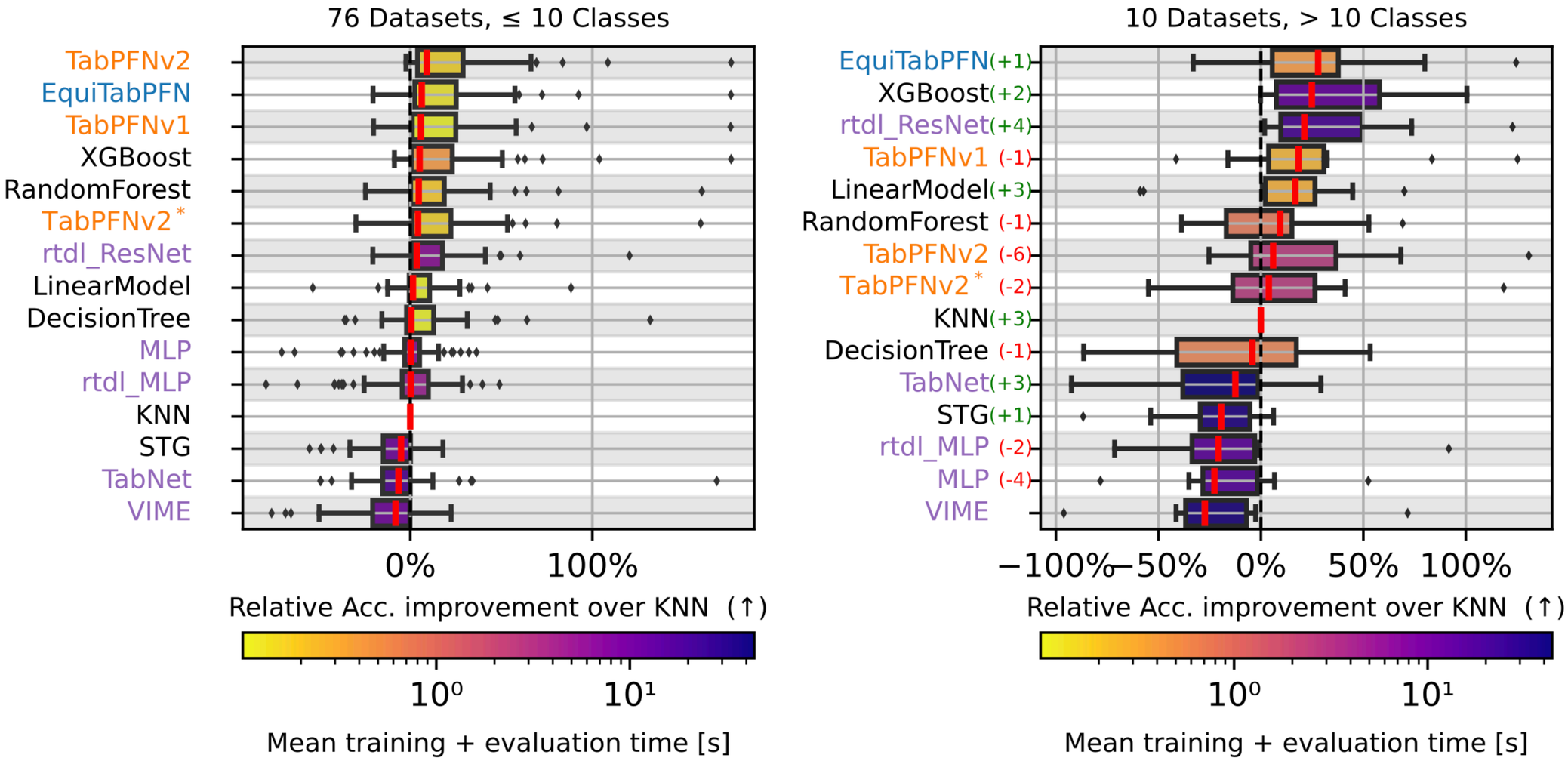
TabPFN-v2: alternate attention over rows / features



Ours: alternate attention over rows / target dimensions

+ non parametric output, weighted by similarity

Enabling ICL on data with unseen class counts



Conclusion

- Handling target equivariance allows to:
  - Obtain stable predictions with respect to target permutation
  - Perform inference on any number of classes, not just the ones seen in training
- Future work:
  - Handle multivariate regression
  - Equivariance to feature symmetry $x \to 1 - x$
  - Single model for regression and classification
- Code available: https://github.com/MichaelArbel/EquiTabPFN/

Tabular Benchmarking

What is the best Tabular method?

Cost of benchmarking

- AutoML Benchmark [Ginsberg et al 2023] considered 71 classification and 33 regression datasets

What is the best Tabular method?

Cost of benchmarking

- AutoML Benchmark [Ginsberg et al 2023] considered 71 classification and 33 regression datasets

Journal of Machine Learning Research 1 (2000) 1-48 Submitted 4/00; Published 10/00

AMLB: an AutoML Benchmark

Pieter Gijsbers[1] P.GIJSBERS@TUE.NL
Marcos L. P. Bueno[1,4] MARCOS.DEPAULABUENO@DONDERS.RU.NL
Stefan Coors[2] STEFAN.COORS@STAT.UNI-MUENCHEN.DE
Erin LeDell[3] ERIN@H2O.AI
Sébastien Poirier[3] SEBASTIEN@H2O.AI
Janek Thomas[2] JANEK.THOMAS@STAT.UNI-MUENCHEN.DE
Bernd Bischl[2] BERND.BISCHL@STAT.UNI-MUENCHEN.DE
Joaquin Vanschoren[1] J.VANSCHOREN@TUE.NL

[1] Eindhoven University of Technology, Eindhoven, The Netherlands
[2] Ludwig Maximilian University of Munich, Munich, Germany
[3] H2O.ai, Mountain View, CA, United States
[4] Radboud University, Nijmegen, The Netherlands

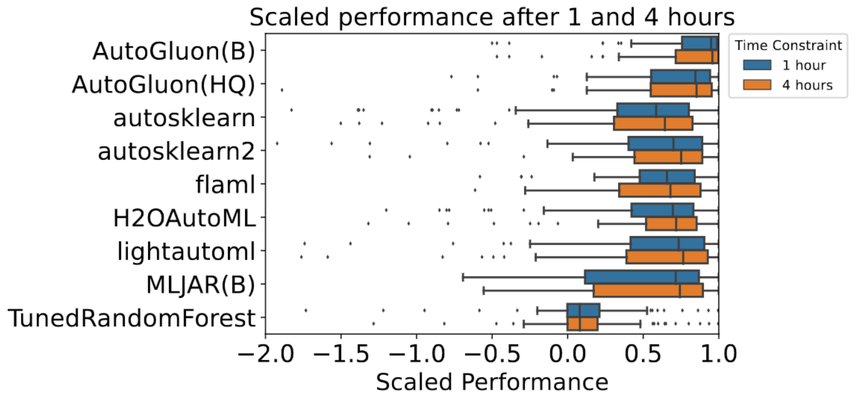


Figure 4: Scaled performance for each framework under different time constraints. Only frameworks which have evaluations on all tasks for both time constraints are shown. Performance generally does not improve much with more time.

What is the best Tabular method?

Cost of benchmarking

- AutoML Benchmark [Ginsberg et al 2023] considered 71 classification and 33 regression datasets

Journal of Machine Learning Research 1 (2000) 1-48 Submitted 4/00; Published 10/00

AMLB: an AutoML Benchmark

Pieter Gijsbers[1] P.GIJSBERS@TUE.NL
Marcos L. P. Bueno[1,4] MARCOS.DEPAULABUENO@DONDERS.RU.NL
Stefan Coors[2] STEFAN.COORS@STAT.UNI-MUENCHEN.DE
Erin LeDell[3] ERIN@H2O.AI
Sébastien Poirier[3] SEBASTIEN@H2O.AI
Janek Thomas[2] JANEK.THOMAS@STAT.UNI-MUENCHEN.DE
Bernd Bischl[2] BERND.BISCHL@STAT.UNI-MUENCHEN.DE
Joaquin Vanschoren[1] J.VANSCHOREN@TUE.NL

[1] Eindhoven University of Technology, Eindhoven, The Netherlands
[2] Ludwig Maximilian University of Munich, Munich, Germany
[3] H2O.ai, Mountain View, CA, United States
[4] Radboud University, Nijmegen, The Netherlands

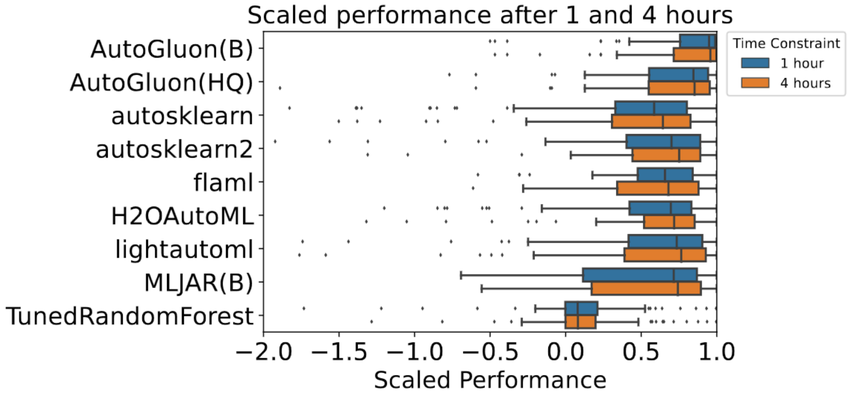


Figure 4: Scaled performance for each framework under different time constraints. Only frameworks which have evaluations on all tasks for both time constraints are shown. Performance generally does not improve much with more time.

Evaluating a single method costs 40K CPU hours of compute!

What is the best Tabular method?

Cost of benchmarking

- AutoML Benchmark [Ginsberg et al 2023] considered 71 classification and 33 regression datasets

Journal of Machine Learning Research 1 (2000) 1-48 Submitted 4/00; Published 10/00

AMLB: an AutoML Benchmark

Pieter Gijsbers[1] P.GIJSBERS@TUE.NL
Marcos L. P. Bueno[1,4] MARCOS.DEPAULABUENO@DONDERS.RU.NL
Stefan Coors[2] STEFAN.COORS@STAT.UNI-MUENCHEN.DE
Erin LeDell[3] ERIN@H2O.AI
Sébastien Poirier[3] SEBASTIEN@H2O.AI
Janek Thomas[2] JANEK.THOMAS@STAT.UNI-MUENCHEN.DE
Bernd Bischl[2] BERND.BISCHL@STAT.UNI-MUENCHEN.DE
Joaquin Vanschoren[1] J.VANSCHOREN@TUE.NL

[1] Eindhoven University of Technology, Eindhoven, The Netherlands
[2] Ludwig Maximilian University of Munich, Munich, Germany
[3] H2O.ai, Mountain View, CA, United States
[4] Radboud University, Nijmegen, The Netherlands

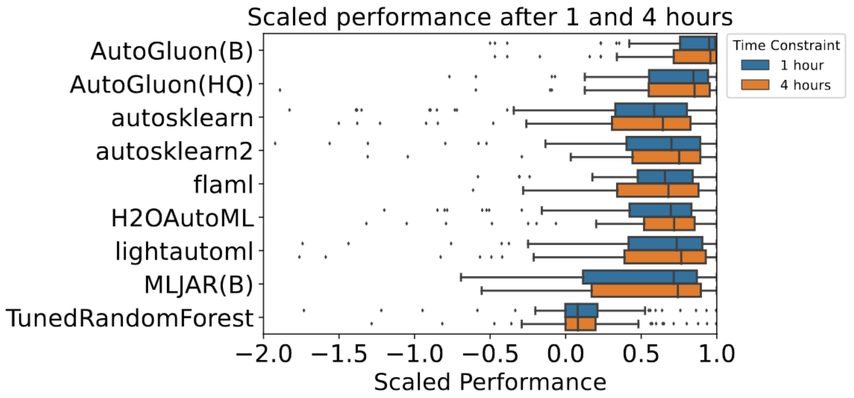


Figure 4: Scaled performance for each framework under different time constraints. Only frameworks which have evaluations on all tasks for both time constraints are shown. Performance generally does not improve much with more time.

Evaluating a single method costs 40K CPU hours of compute!

Can we limit this cost? 🤔

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

TabRepo

- Goals:

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:
  - 200 datasets from regression, classification, multi-class (thanks OpenML 🥰)

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:
  - 200 datasets from regression, classification, multi-class (thanks OpenML 🥰)
  - 200 random configurations of models used in AutoGluon (CatBoost, MLP, LightGBM, RandomForest, …) on all datasets with 3 seeds

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:
  - 200 datasets from regression, classification, multi-class (thanks OpenML 🥰)
  - 200 random configurations of models used in AutoGluon (CatBoost, MLP, LightGBM, RandomForest, …) on all datasets with 3 seeds
- Performance metrics (latency, accuracy, …) **and predictions** available for every dataset, model, seed

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:
  - 200 datasets from regression, classification, multi-class (thanks OpenML 🥰)
  - 200 random configurations of models used in AutoGluon (CatBoost, MLP, LightGBM, RandomForest, …) on all datasets with 3 seeds
- Performance metrics (latency, accuracy, …) **and predictions** available for every dataset, model, seed
- ~100GB of data, ~200K CPU hours of compute

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:
  - 200 datasets from regression, classification, multi-class (thanks OpenML 🥰)
  - 200 random configurations of models used in AutoGluon (CatBoost, MLP, LightGBM, RandomForest, …) on all datasets with 3 seeds
- Performance metrics (latency, accuracy, …) **and predictions** available for every dataset, model, seed
- ~100GB of data, ~200K CPU hours of compute

💡 **Storing predictions** and target labels allows to obtain the performance of **any ensemble** on the fly!

TabRepo

TabRepo: A Large Scale Repository of Tabular Model Evaluations and its AutoML Applications

David Salinas[1,*] Nick Erickson[1,*]

AutoML Conf 2023

- Goals:
  - 1) reduce cost of evaluation
  - 2) improve over the default hyperparameters of AutoGluon/AutoML systems
- Precomputed evaluations and results on:
  - 200 datasets from regression, classification, multi-class (thanks OpenML 🥰)
  - 200 random configurations of models used in AutoGluon (CatBoost, MLP, LightGBM, RandomForest, …) on all datasets with 3 seeds
- Performance metrics (latency, accuracy, …) **and predictions** available for every dataset, model, seed
- ~100GB of data, ~200K CPU hours of compute

💡 **Storing predictions** and target labels allows to obtain the performance of **any ensemble** on the fly!

🥳 The dataset combined with **portfolio learning** allows to outperform Autogluon!

TabRepo

Studying the effect of HPO and ensembling

TabRepo

Studying the effect of HPO and ensembling

💡 **Storing predictions** and target labels allows to obtain the performance of **any ensemble** on the fly!

TabRepo

Studying the effect of HPO and ensembling

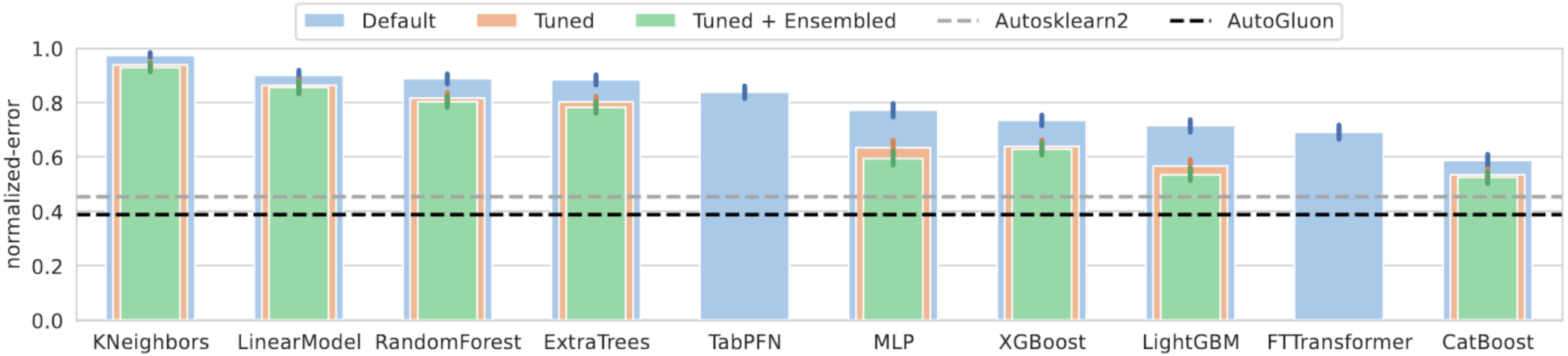


Figure 2: Normalized error for all model families when using default hyperparameters, tuned hyperparameters, and ensembling after tuning. All methods are run with a 4h budget.

💡 **Storing predictions** and target labels allows to obtain the performance of **any ensemble** on the fly!

TabRepo

Studying the effect of HPO and ensembling

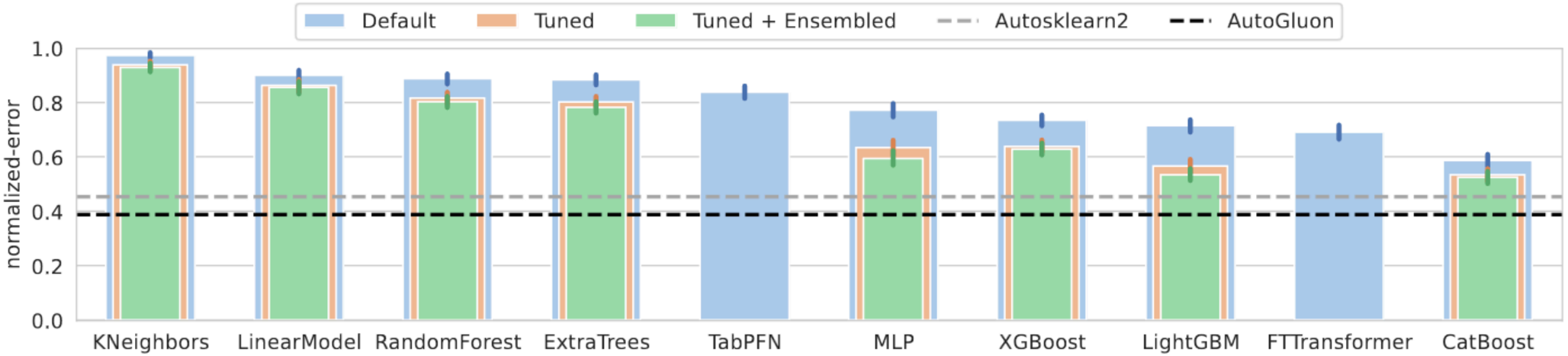


Figure 2: Normalized error for all model families when using default hyperparameters, tuned hyperparameters, and ensembling after tuning. All methods are run with a 4h budget.

💡 **Storing predictions** and target labels allows to obtain the performance of **any ensemble** on the fly!

Doing this analysis just costs a few minutes on a laptop (as opposed to days on a cluster!)

TabRepo

Studying the effect of HPO and ensembling

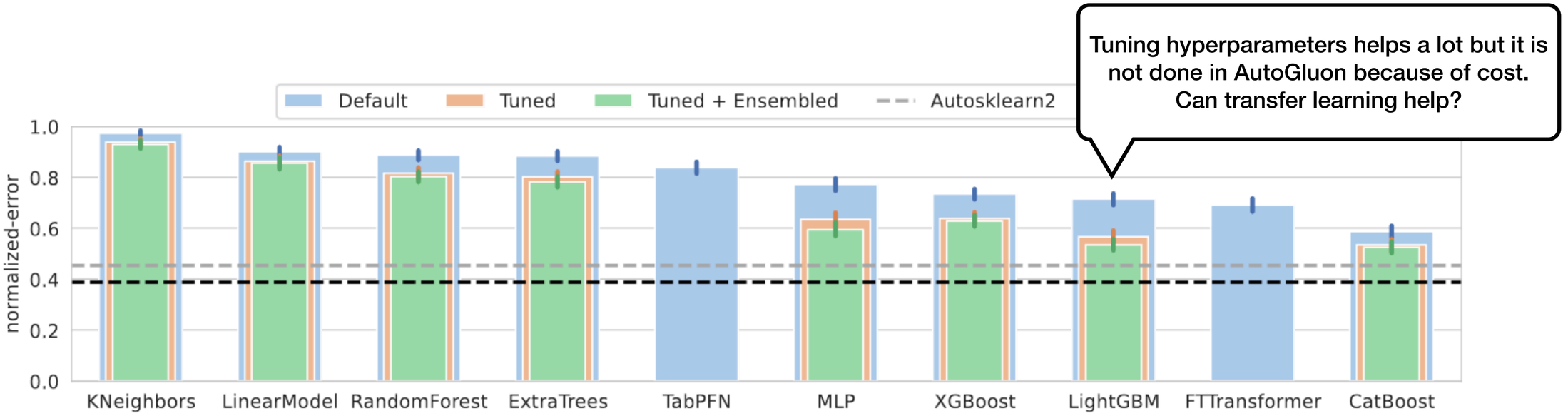


Figure 2: Normalized error for all model families when using default hyperparameters, tuned hyperparameters, and ensembling after tuning. All methods are run with a 4h budget.

💡 **Storing predictions** and target labels allows to obtain the performance of **any ensemble** on the fly!

Doing this analysis just costs a few minutes on a laptop (as opposed to days on a cluster!)

Portfolio learning

Reaping the benefits of evaluations

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem:

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem:

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem:

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \operatorname{argmin}_{(j_1,\ldots,j_k) \in [m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization proble

With best avg. performance across datasets

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization proble

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k) \in [m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem:

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k) \in [m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation
- Greedy algorithm:

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization probl

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation
- Greedy algorithm:

$$j_1 = \text{argmin}_{j_1\in[m]} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_{ij_1}, \qquad j_n = \text{argmin}_{j_n\in[m]} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_n})$$

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation
- Greedy algorithm:

$$j_1 = \text{argmin}_{j_1\in[m]} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_{ij_1}, \qquad j_n = \text{argmin}_{j_n\in[m]} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_n})$$

Start by the model performing best on average

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization problem:

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation
- Greedy algorithm:

$$j_1 = \text{argmin}_{j_1\in[m]} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_{ij_1}, \qquad j_n = \text{argmin}_{j_n\in[m]} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_n})$$

Start by the model performing best on average

Greedily pick the model performing best on average when combined with the ones previously selected

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization probl

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation
- Greedy algorithm:

$$j_1 = \text{argmin}_{j_1\in[m]} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_{ij_1}, \qquad j_n = \text{argmin}_{j_n\in[m]} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_n})$$

Start by the model performing best on average

Greedily pick the model performing best on average when combined with the ones previously selected

Benefits 👍:

- Approximation guarantees from the original (sub-modular) problem
- Tractable
- Works **extremely well** in practice

Portfolio learning

Reaping the benefits of evaluations

- Assume we have access to error metrics of n datasets on m models, denoted as $\varepsilon \in \mathbb{R}^{n \times m}$
- How can we select the best set of k default models for an average dataset?
- Solve the optimization probl

With best avg. performance across datasets

… when using the best performing model on a given dataset

Select among all possible sets of k models

$$(j_1, \ldots, j_k) = \text{argmin}_{(j_1,\ldots,j_k)\in[m]^k} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_k})$$

- NP-hard [Feurer 2022], but admits an approximation
- Greedy algorithm:

$$j_1 = \text{argmin}_{j_1\in[m]} \frac{1}{n} \sum_{i=1}^{n} \varepsilon_{ij_1}, \qquad j_n = \text{argmin}_{j_n\in[m]} \frac{1}{n} \sum_{i=1}^{n} \min(\varepsilon_{ij_1}, \ldots, \varepsilon_{ij_n})$$

Start by the model performing best on average

Greedily pick the model performing best on average when combined with the ones previously selected

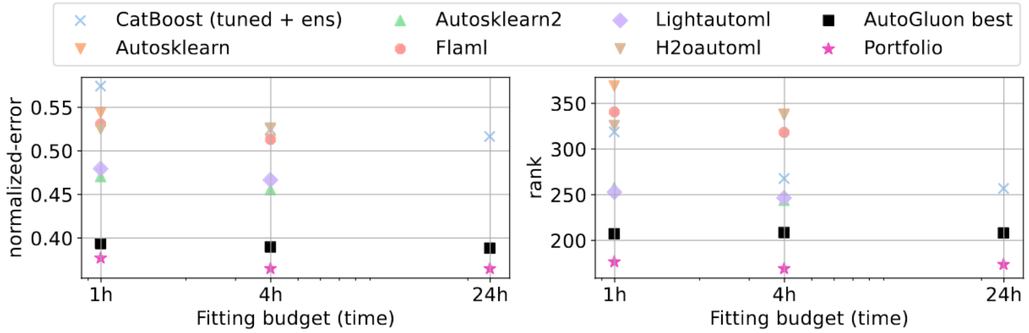
Benefits 👍:

- Approximation guarantees from the original (sub-modular) problem
- Tractable
- Works **extremely well** in practice

Disadvantage 👎: needs a grid or a surrogate

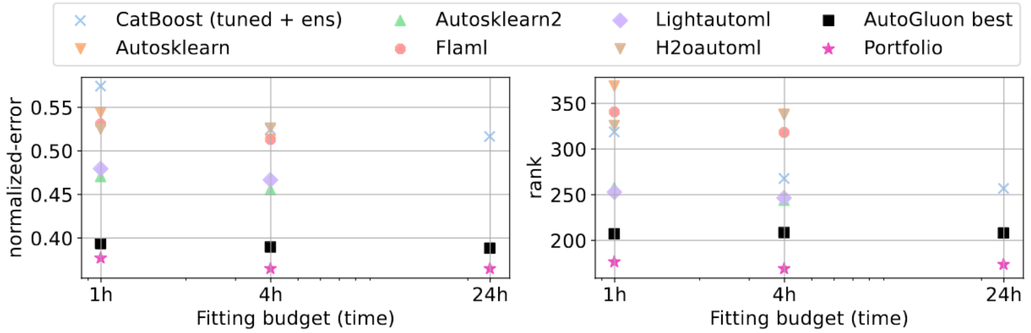
Results

Results


CatBoost (tuned + ens)
Autosklearn
Autosklearn2
Flaml
Lightautoml
H2oautoml
AutoGluon best
Portfolio
normalized-error
0.55
0.50
0.45
0.40
1h
4h
24h
Fitting budget (time)
rank
350
300
250
200
1h
4h
24h
Fitting budget (time)

Results

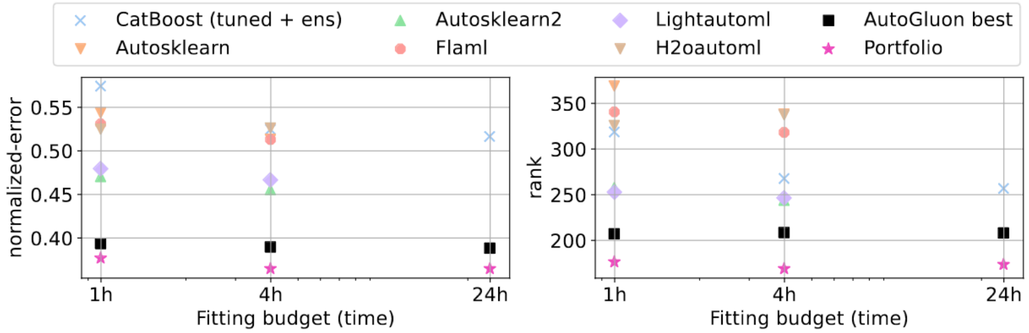
- Just fitting portfolio configuration on evaluations of TabRepo outperforms all SOTA AutoML methods studied



Results

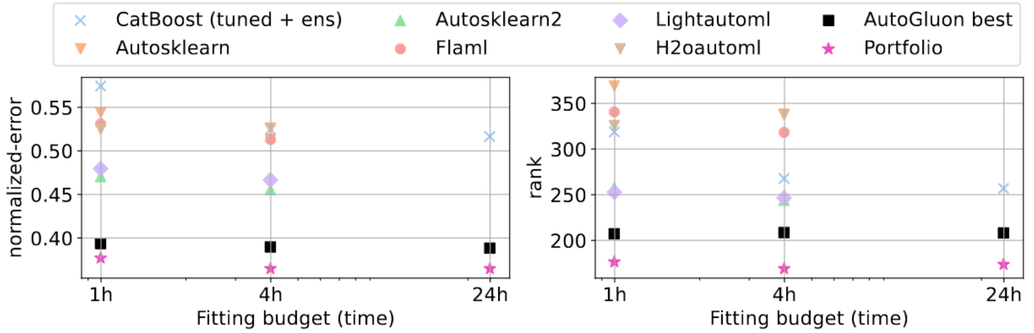
- Just fitting portfolio configuration on evaluations of TabRepo outperforms all SOTA AutoML methods studied

- We can analyse the performance of various components: #ensemble, #configurations, #datasets



Results

- Just fitting portfolio configuration on evaluations of TabRepo outperforms all SOTA AutoML methods studied
- We can analyse the performance of various components: #ensemble, #configurations, #datasets



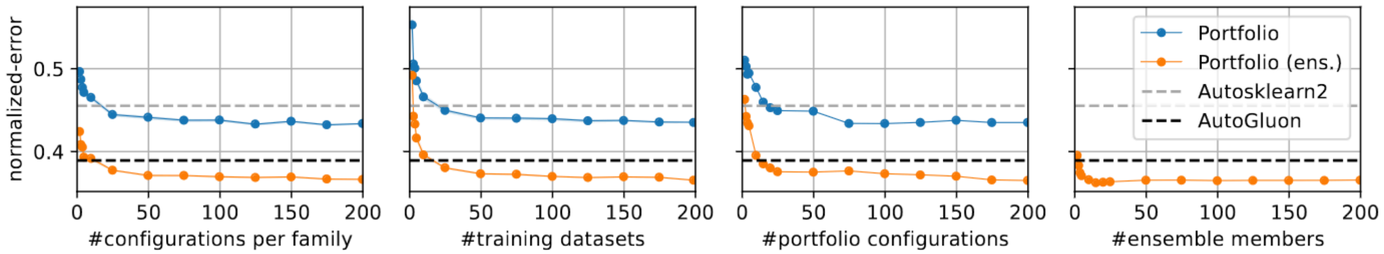
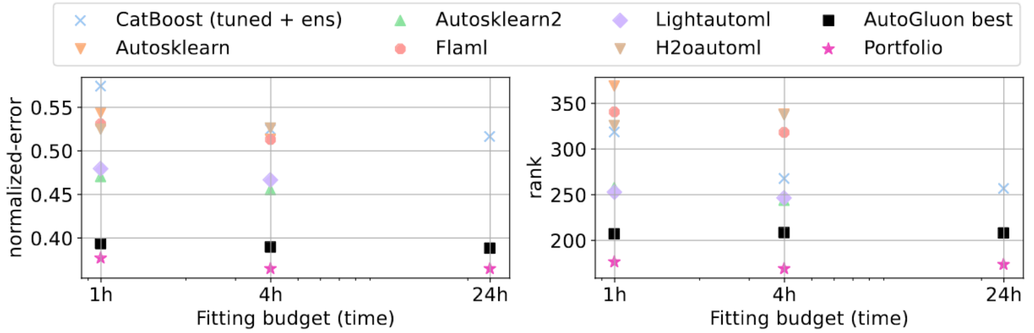


Figure 4: Impact on normalized error when varying the (a) number of configurations per family, (b) number of training datasets, (c) portfolio size and (d) number of ensemble members.

Results

- Just fitting portfolio configuration on evaluations of TabRepo outperforms all SOTA AutoML methods studied
- We can analyse the performance of various components: #ensemble, #configurations, #datasets
- Portfolio configurations has replaced the manually configured defaults and improved significantly AutoGluon



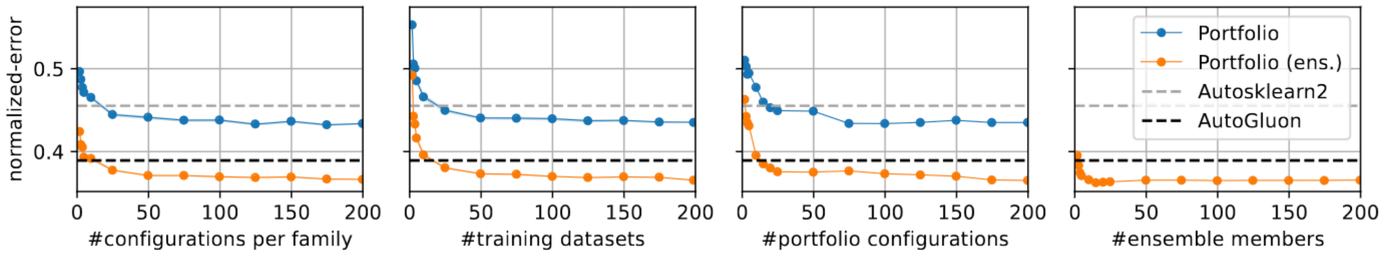
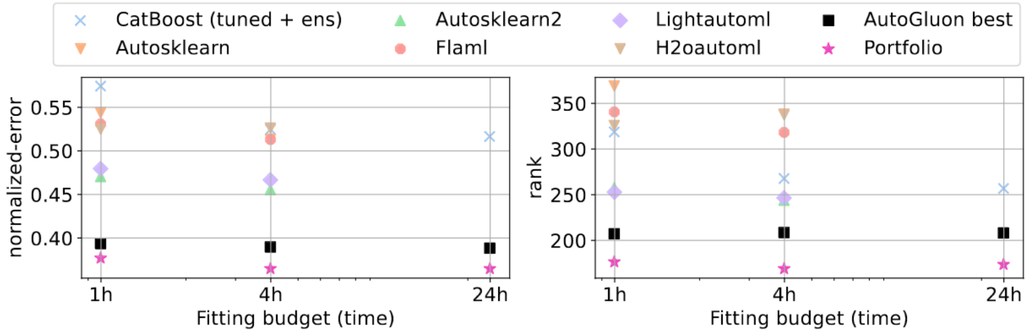


Figure 4: Impact on normalized error when varying the (a) number of configurations per family, (b) number of training datasets, (c) portfolio size and (d) number of ensemble members.

Results

- Just fitting portfolio configuration on evaluations of TabRepo outperforms all SOTA AutoML methods studied
- We can analyse the performance of various components: #ensemble, #configurations, #datasets
- Portfolio configurations has replaced the manually configured defaults and improved significantly AutoGluon



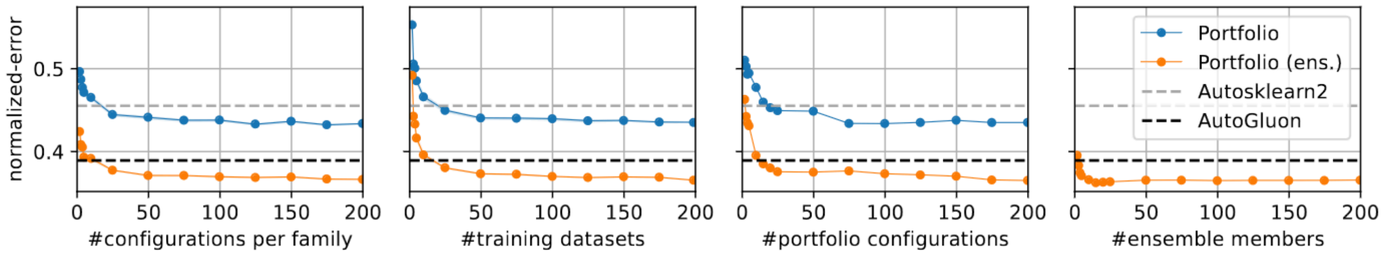


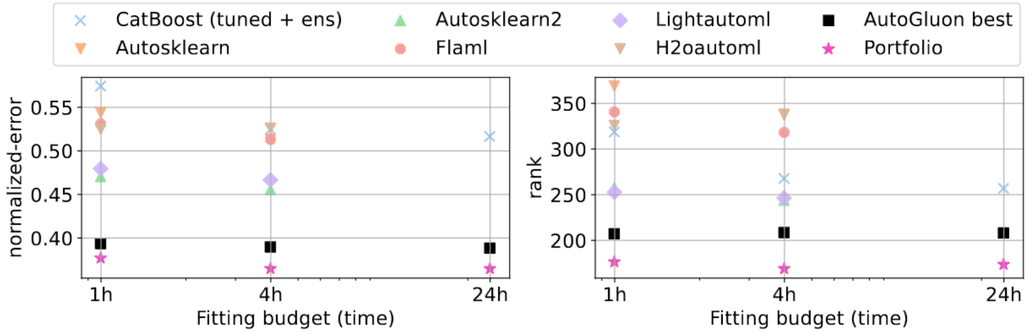
Figure 4: Impact on normalized error when varying the (a) number of configurations per family, (b) number of training datasets, (c) portfolio size and (d) number of ensemble members.

Table 2: Performance of AutoGluon combined with portfolios on AMLB.

method	win-rate	loss reduc.
AG + Portfolio (ours)	-	**0%**
AG	67%	2.8%
MLJAR	81%	22.5%
lightautoml	83%	11.7%
GAMA	86%	15.5%
FLAML	87%	16.3%
autosklearn	89%	11.8%
H2OAutoML	92%	10.3%
CatBoost	94%	18.1%
TunedRandomForest	94%	22.9%
RandomForest	97%	25.0%
XGBoost	98%	20.9%
LightGBM	98%	23.6%

Results

- Just fitting portfolio configuration on evaluations of TabRepo outperforms all SOTA AutoML methods studied
- We can analyse the performance of various components: #ensemble, #configurations, #datasets
- Portfolio configurations has replaced the manually configured defaults and improved significantly AutoGluon



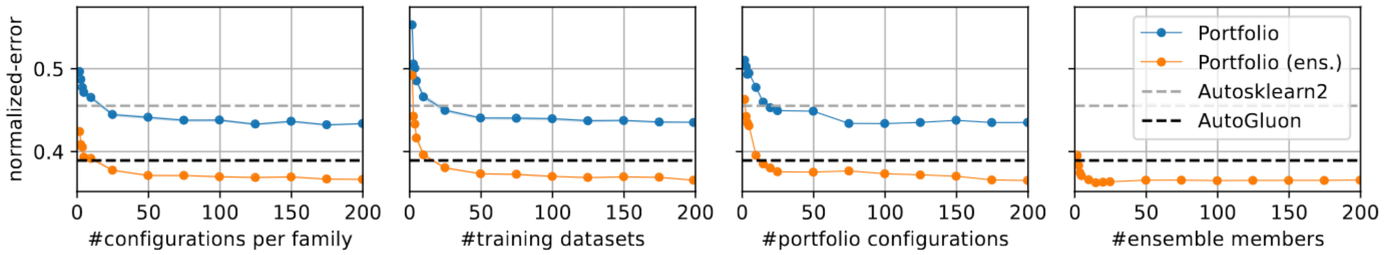


Figure 4: Impact on normalized error when varying the (a) number of configurations per family, (b) number of training datasets, (c) portfolio size and (d) number of ensemble members.

Table 2: Performance of AutoGluon combined with portfolios on AMLB.

method	win-rate	loss reduc.
AG + Portfolio (ours)	-	**0%**
AG	67%	2.8%
MLJAR	81%	22.5%
lightautoml	83%	11.7%
GAMA	86%	15.5%
FLAML	87%	16.3%
autosklearn	89%	11.8%
H2OAutoML	92%	10.3%
CatBoost	94%	18.1%
TunedRandomForest	94%	22.9%
RandomForest	97%	25.0%
XGBoost	98%	20.9%
LightGBM	98%	23.6%

Limitations

Limitations

- Easy to rerun paper analysis but hard to compare your own method

Limitations

- Easy to rerun paper analysis but hard to compare your own method
- Large collections of datasets (216) but mostly grabbed everything we could

Limitations

- Easy to rerun paper analysis but hard to compare your own method
- Large collections of datasets (216) but mostly grabbed everything we could
- No good control on quality, duplication, domain

Limitations

- Easy to rerun paper analysis but hard to compare your own method
- Large collections of datasets (216) but mostly grabbed everything we could
- No good control on quality, duplication, domain
- Only TabPFN-v1 as In Context Learning (ICL) method

TabArena: A Living Benchmark for Machine Learning on Tabular Data

Nick Erickson[1] **Lennart Purucker**[2] **Andrej Tschalzev**[3] **David Holzmüller**[4,5,6]
Prateek Mutalik Desai[1] **David Salinas**[8,2] **Frank Hutter**[7,8,2]
[1]Amazon Web Services [2]University of Freiburg [3]University of Mannheim [4]INRIA Paris
[5]Ecole Normale Supérieure [6]PSL Research University [7]Prior Labs [8]ELLIS Institute Tübingen
mail@tabarena.ai

NeurIPS 2025 Spotlight

TabArena

TabArena

Nick Erickson

Lennart Purucker

Andrej Tschalzev

David Holzmüller

Prateek Mutalik Desai

David Salinas

Frank Hutter

aws

ufr

small data
MORE FROM LESS
CRC 1597

IN OMNIBUS VERITAS
UNIVERSITÄT MANNHEIM

Inria

1794
ENS
PSL
UNIVERSITÉ PARIS

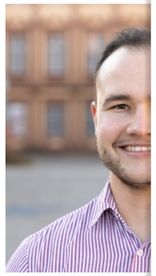
ellis
INSTITUTE
TÜBINGEN

PRIOR
LABS

TabArena

Nick Erickson

Lennart Purucker



And
Tscha

Prateek Mutalik Desai

David Salinas

Competing interests

D.H. is one of the authors of RealMLP and one of the authors of TabICL.
D.S. and N.E. are the authors of TabRepo.
N.E., L.P., and P.M.D. are developers of AutoGluon, and in extension, the current maintainers of FastAI MLP and Torch MLP.
L.P. and F.H. are a subset of the authors of TabPFNv2.
L.P. is an OpenML core contributor.
F.H. is affiliated with PriorLabs, a company focused on developing tabular foundation models.
The authors declare no other competing interests.

Hutter

small data
MORE FROM LESS
CRC 1597

1794
ENS
PSL
UNIVERSITÉ PARIS

PRIOR
LABS

Motivation 1: Unreliable Baselines

How to become SOTA on the highly used benchmark by McElfresh et al. (2023):

Model	Avg. Rank	Avg. norm. logloss	Avg. logloss
XGBoost	5.56	0.1	0.39
CatBoost	5.84	0.12	0.45
LightGBM	6.85	0.17	0.45
ResNet	8.12	0.22	0.49
SAINT	8.77	0.23	0.52
...			
MLP	10.79	0.39	0.96
...			
KNN	15.68	0.71	0.88

Tschalzev, Andrej, et al. "Unreflected use of tabular data repositories can undermine research quality." (2025).

Motivation 1: Unreliable Baselines

How to become SOTA on the highly used benchmark by McElfresh et al. (2023):

Model	Avg. Rank	Avg. norm. logloss	Avg. logloss
XGBoost (ours, holdout)	4.13	0.06	0.36
XGBoost	5.56	0.1	0.39
CatBoost	5.84	0.12	0.45
MLP (ours, holdout)	6.09	0.15	0.4
LightGBM	6.85	0.17	0.45
ResNet	8.12	0.22	0.49
SAINT	8.77	0.23	0.52
...			
MLP	10.79	0.39	0.96
...			
KNN	15.68	0.71	0.88

Tschalzev, Andrej, et al. "Unreflected use of tabular data repositories can undermine research quality." (2025).

Motivation 1: Unreliable Baselines

How to become SOTA on the highly used benchmark by McElfresh et al. (2023):

Model	Avg. Rank	Avg. norm. logloss	Avg. logloss
XGBoost (ours, holdout)	4.13	0.06	0.36
XGBoost	5.56	0.1	0.39
CatBoost	5.84	0.12	0.45
MLP (ours, holdout)	6.09	0.15	0.4
LightGBM	6.85	0.17	0.45
ResNet	8.12	0.22	0.49
SAINT	8.77	0.23	0.52
...			
MLP	10.79	0.39	0.96
...			
KNN	15.68	0.71	0.88

Accepted ICML and NeurIPS papers (that claim SOTA)

Tschalzev, Andrej, et al. "Unreflected use of tabular data repositories can undermine research quality." (2025).

Motivation 1: Unreliable Baselines

How to become SOTA on the highly used benchmark by McElfresh et al. (2023):

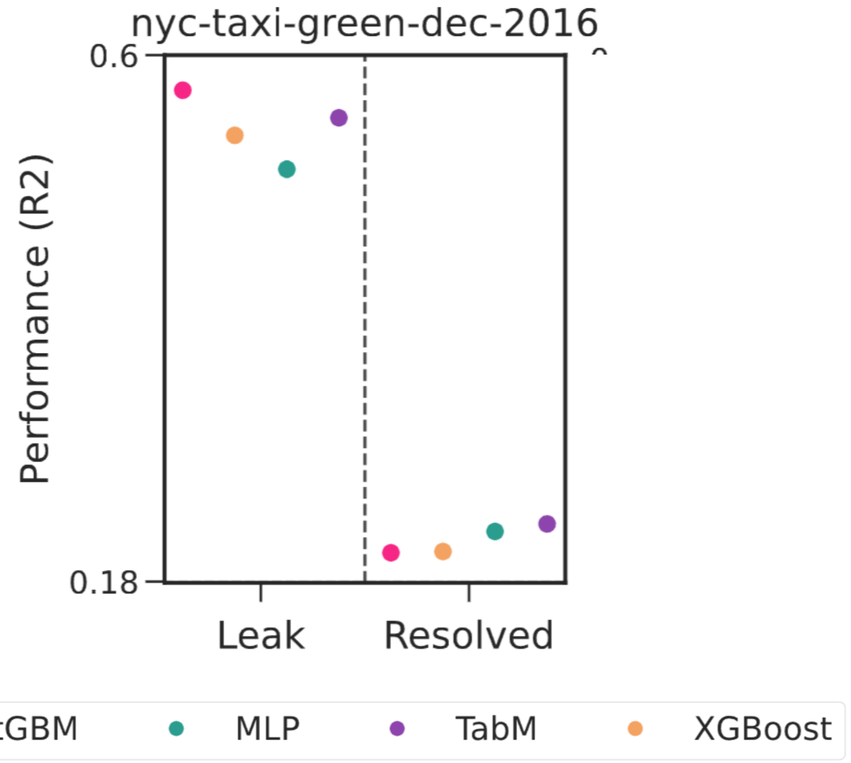
Model	Avg. Rank	Avg. norm. logloss	Avg. logloss
XGBoost (ours, 5CV)	1.77	0.03	0.34
MLP (ours, 5CV)	2.1	0.08	0.34
XGBoost (ours, holdout)	4.13	0.06	0.36
XGBoost	5.56	0.1	0.39
CatBoost	5.84	0.12	0.45
MLP (ours, holdout)	6.09	0.15	0.4
LightGBM	6.85	0.17	0.45
ResNet	8.12	0.22	0.49
SAINT	8.77	0.23	0.52
...			
MLP	10.79	0.39	0.96
...			
KNN	15.68	0.71	0.88

Accepted ICML and NeurIPS papers (that claim SOTA)

Tschalzev, Andrej, et al. "Unreflected use of tabular data repositories can undermine research quality." (2025).

Motivation 2: Insufficient Dataset Curation

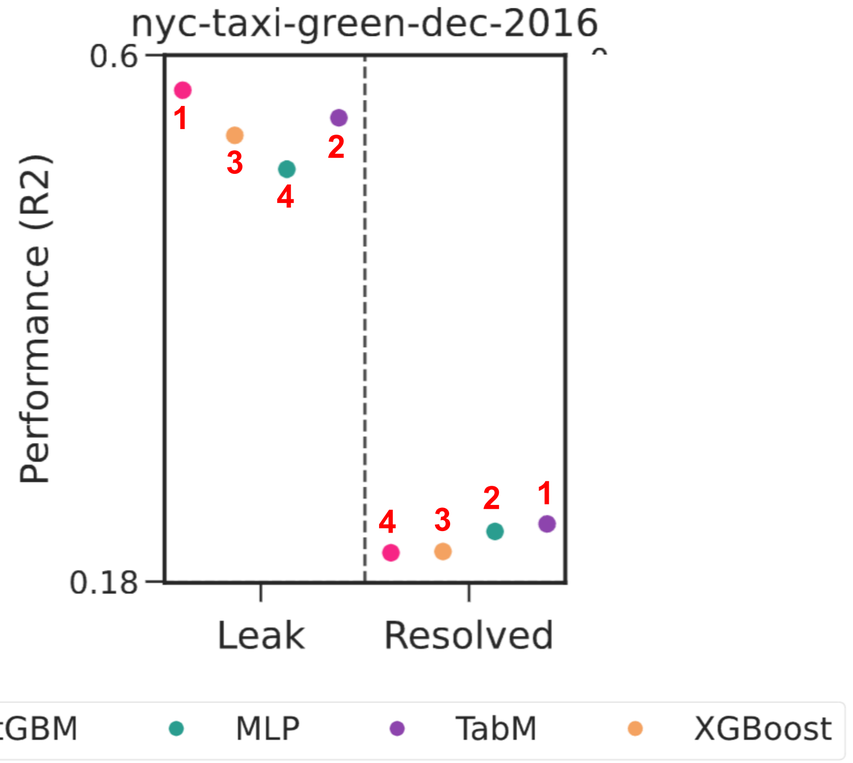
Faulty data influences the results:



Tschalzev, Andrej, et al. "Unreflected use of tabular data repositories can undermine research quality." (2025).

Motivation 2: Insufficient Dataset Curation

Faulty data influences the results:



Tschalzev, Andrej, et al. "Unreflected use of tabular data repositories can undermine research quality." (2025).

Motivation 3: Inappropriate Evaluation Protocols

Splits must be appropriate for the data:

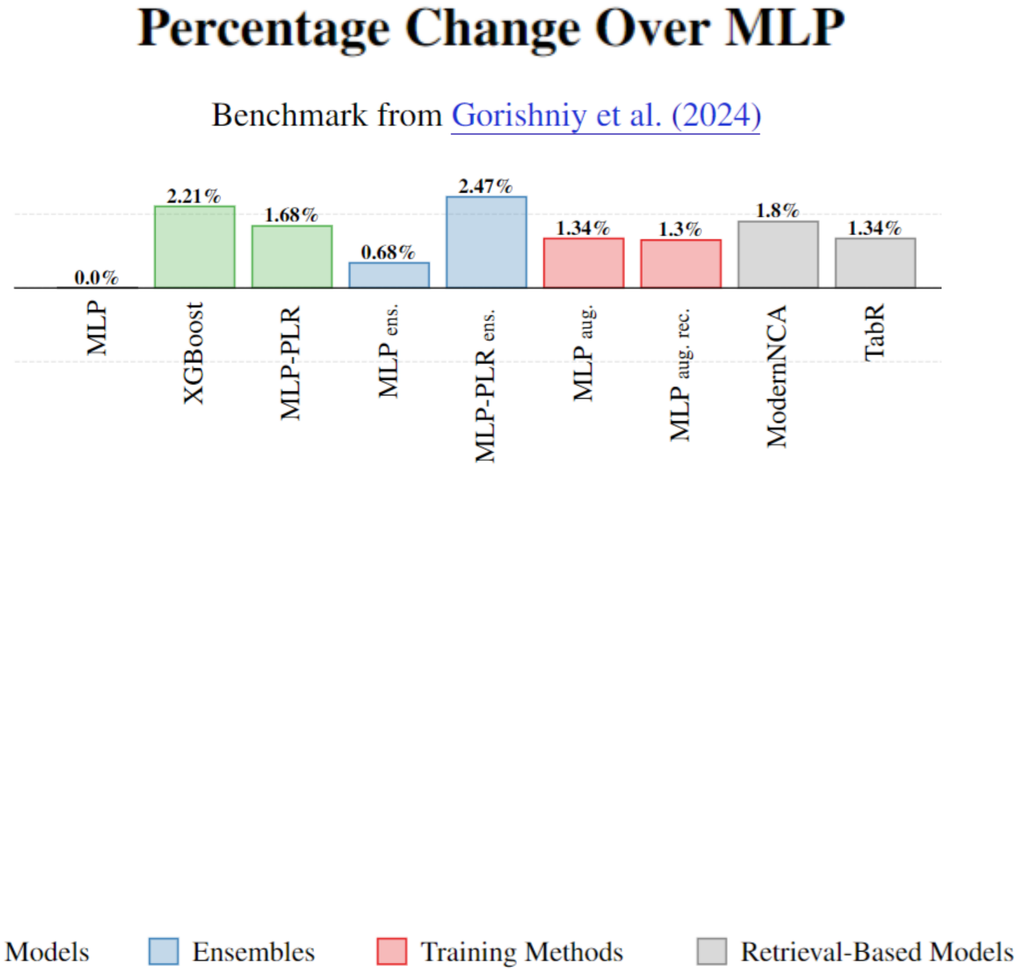
Benchmark	Time-split		
	Needed	Possible	Used
Grinsztajn et al. (2022)	22	5	
Tabzilla (McElfresh et al., 2023)	12	0	
WildTab (Kolesnikov, 2023)	1	1	✗
TableShift (Gardner et al., 2023)	15	8	
Gorishniy et al. (2024)	7	1	

Rubachev, Ivan, et al. "TabReD: Analyzing Pitfalls and Filling the Gaps in Tabular Deep Learning Benchmarks.“ (2024)

Motivation 3: Inappropriate Evaluation Protocols

Splits must be appropriate for the data:

Benchmark	Time-split		
	Needed	Possible	Used
Grinsztajn et al. (2022)	22	5	
Tabzilla (McElfresh et al., 2023)	12	0	
WildTab (Kolesnikov, 2023)	1	1	✗
TableShift (Gardner et al., 2023)	15	8	
Gorishniy et al. (2024)	7	1	



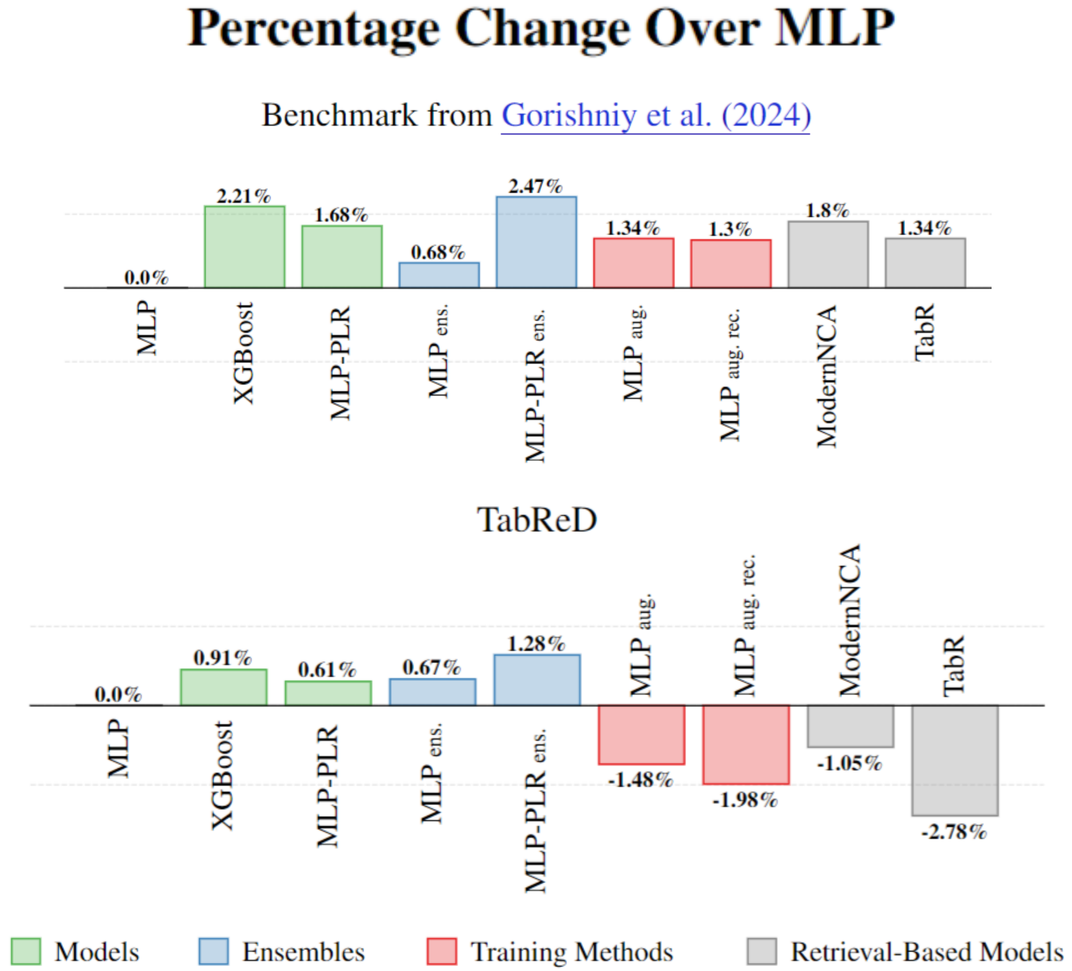
Rubachev, Ivan, et al. "TabReD: Analyzing Pitfalls and Filling the Gaps in Tabular Deep Learning Benchmarks.“ (2024)

Motivation 3: Inappropriate Evaluation Protocols

Splits must be appropriate for the data:

Benchmark	Time-split		
	Needed	Possible	Used
Grinsztajn et al. (2022)	22	5	
Tabzilla (McElfresh et al., 2023)	12	0	
WildTab (Kolesnikov, 2023)	1	1	✗
TableShift (Gardner et al., 2023)	15	8	
Gorishniy et al. (2024)	7	1	

Rubachev, Ivan, et al. "TabReD: Analyzing Pitfalls and Filling the Gaps in Tabular Deep Learning Benchmarks.“ (2024)



Motivation Summary

(Partial) Overview of Tabular Benchmarks

Bischl et al. [28, 29]
Gorishniy et al. [30]
Shwartz-Ziv and Armon [31]
Grinsztajn et al. [32]
McElfresh et al. [33]
Fischer et al. [34]
Gijsbers et al. [35]
Kohli et al. [7]
Tschalzev et al. [8]
Holzmüller et al. [20]
Ye et al. [36]
Rubachev et al. [10]
Salinas and Erickson [37]

Erickson, Nick, et al. "TabArena: A Living Benchmark for Machine Learning on Tabular Data." (2025).

Motivation Summary

(Partial) Overview of Tabular Benchmarks

Bischl et al. [28, 29]
Gorishniy et al. [30]
Shwartz-Ziv and Armon [31]
Grinsztajn et al. [32]
McElfresh et al. [33]
Fischer et al. [34]
Gijsbers et al. [35]
Kohli et al. [7]
Tschalzev et al. [8]
Holzmüller et al. [20]
Ye et al. [36]
Rubachev et al. [10]
Salinas and Erickson [37]

One more benchmark should fix it!

Erickson, Nick, et al. "TabArena: A Living Benchmark for Machine Learning on Tabular Data." (2025).

Motivation Summary

(Partial) Overview of Tabular Benchmarks

Bischl et al. [28, 29]
Gorishniy et al. [30]
Shwartz-Ziv and Armon [31]
Grinsztajn et al. [32]
McElfresh et al. [33]
Fischer et al. [34]
Gijsbers et al. [35]
Kohli et al. [7]
Tschalzev et al. [8]
Holzmüller et al. [20]
Ye et al. [36]
Rubachev et al. [10]
Salinas and Erickson [37]

Erickson, Nick, et al. "TabArena: A Living Benchmark for Machine Learning on Tabular Data." (2025).

Motivation Summary

(Partial) Overview of Tabular Benchmarks

Bischl et al. [28, 29]
Gorishniy et al. [30]
Shwartz-Ziv and Armon [31]
Grinsztajn et al. [32]
McElfresh et al. [33]
Fischer et al. [34]
Gijsbers et al. [35]
Kohli et al. [7]
Tschalzev et al. [8]
Holzmüller et al. [20]
Ye et al. [36]
Rubachev et al. [10]
Salinas and Erickson [37]

Benchmarks require continuous updates!

Erickson, Nick, et al. "TabArena: A Living Benchmark for Machine Learning on Tabular Data." (2025).

TabArena-v0.1

Models

Models

Models, Hyperparameters, and Tuning

1. **SOTA** tree-based, neural networks, and foundation **models**.
2. Implemented **with authors**
3. Good, **optimized** search spaces

Model	Short Name	Search Space	Type
Random Forests [12]	RandomForest	Prior Work + Us	🌳
Extremely Randomized Trees [13]	ExtraTrees	Prior Work + Us	🌳
XGBoost [14]	XGBoost	Prior Work + Us	🌳
LightGBM [15]	LightGBM	Prior Work + Us	🌳
CatBoost [16]	CatBoost	Prior Work + Us	🌳
Explainable Boosting Machine [17, 18]	EBM	Authors	🌳
FastAI MLP [19]	FastaiMLP	Authors	🧠
Torch MLP [19]	TorchMLP	Authors	🧠
RealMLP [20]	RealMLP	Authors	🧠
$\text{TabM}^{\dagger}_{\text{mini}}$ [9]	TabM	Authors	🧠
ModernNCA [21]	ModernNCA	Authors	🧠
TabPFNv2 [5]	TabPFNv2	Authors	🤖
TabICL [22]	TabICL	-	🤖
TabDPT [23]	TabDPT	-	🤖
Linear / Logistic Regression	Linear	TabRepo	📏
K-Nearest Neighbors	KNN	TabRepo	📏

tree-based (🌳), neural network (🧠), pretrained foundation models (🤖), and baseline (📏)

Models, Hyperparameters, and Tuning

Models

Benchmark	#splits inner
Bischl et al. [28, 29]	1
Gorishniy et al. [30]	1
Shwartz-Ziv and Armon [31]	1
Grinsztajn et al. [32]	1
McElfresh et al. [33]	1
Fischer et al. [34]	{1, 3, 10}
Gijsbers et al. [35]	-
Kohli et al. [7]	1
Tschalzev et al. [8]	10
Holzmüller et al. [20]	1
Ye et al. [36]	1
Rubachev et al. [10]	1
Salinas and Erickson [37]	8
TabArena (Ours)	8

Peak Performance by:

- Proper (inner) **cross-validation to avoid overfitting**

Models, Hyperparameters, and Tuning

Models

Benchmark	#splits inner	Ensembling
Bischl et al. [28, 29]	1	✗
Gorishniy et al. [30]	1	(✓)
Shwartz-Ziv and Armon [31]	1	(✓)
Grinsztajn et al. [32]	1	✗
McElfresh et al. [33]	1	✗
Fischer et al. [34]	{1, 3, 10}	✗
Gijsbers et al. [35]	-	(✓)
Kohli et al. [7]	1	✗
Tschalzev et al. [8]	10	(✓)
Holzmüller et al. [20]	1	(✓)
Ye et al. [36]	1	✗
Rubachev et al. [10]	1	(✓)
Salinas and Erickson [37]	8	✓
TabArena (Ours)	8	✓

Peak Performance by:

- Proper (inner) **cross-validation to avoid overfitting**
- Model-wise **post-hoc ensembling** (Caruana et al.)

Models, Hyperparameters, and Tuning

Models

Benchmark	#splits inner	Ensembling	HPO Limit #confs.	HPO Limit #hours
Bischl et al. [28, 29]	1	✗	1	-
Gorishniy et al. [30]	1	(✓)	100	6
Shwartz-Ziv and Armon [31]	1	(✓)	1000	-
Grinsztajn et al. [32]	1	✗	400	-
McElfresh et al. [33]	1	✗	30	10
Fischer et al. [34]	{1, 3, 10}	✗	{-, 500}	-
Gijsbers et al. [35]	-	(✓)	-	4
Kohli et al. [7]	1	✗	100	{3, -}
Tschalzev et al. [8]	10	(✓)	100	-
Holzmüller et al. [20]	1	(✓)	50	-
Ye et al. [36]	1	✗	100	-
Rubachev et al. [10]	1	(✓)	100	-
Salinas and Erickson [37]	8	✓	200	200
TabArena (Ours)	8	✓	200	200

Peak Performance by:

- Proper (inner) **cross-validation to avoid overfitting**
- Model-wise **post-hoc ensembling** (Caruana et al.)
- **Extensive HPO** (200 configs, 1 hour per config)

TabArena-v0.1

Datasets

Datasets Curation

Datasets

1053 datasets with unique names from 13 tabular benchmarks

Deduplication
- 254 alternative version
- 167 same but other names
- 63 regex + sanity check
- 7 similar tasks

= **562**

Other domain
- 66 image
- 39 forecasting
- 13 audio
- 12 text
- 5 control

= **427**

Real predictive task
- 49 scientific discovery
- 44 deterministic
- 30 artificial or simulated

= **304**

Other
- 142 tiny data
- 32 quality issues
- 9 License

= **121**

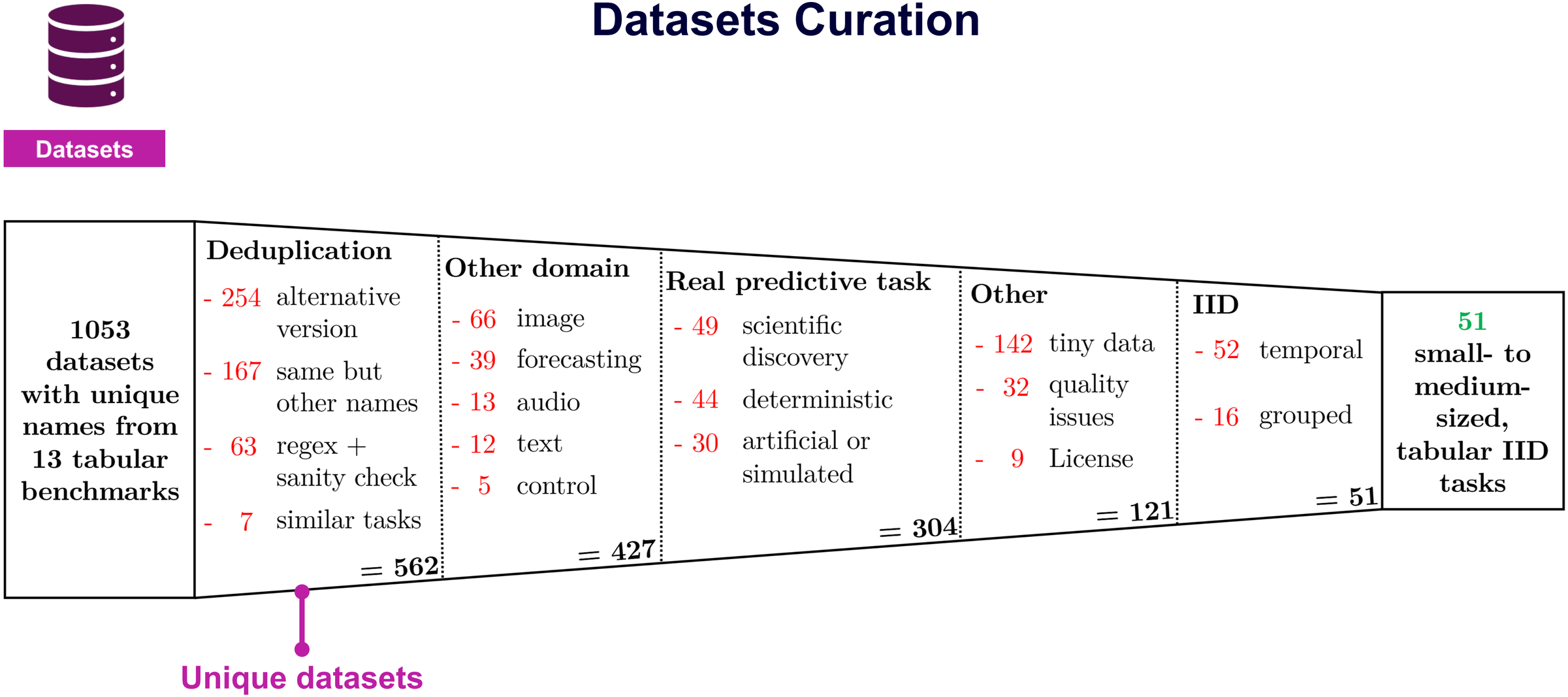
IID
- 52 temporal
- 16 grouped

= **51**

51 small- to medium-sized, tabular IID tasks

Results of our *manual* curation: *51 out of 1053*

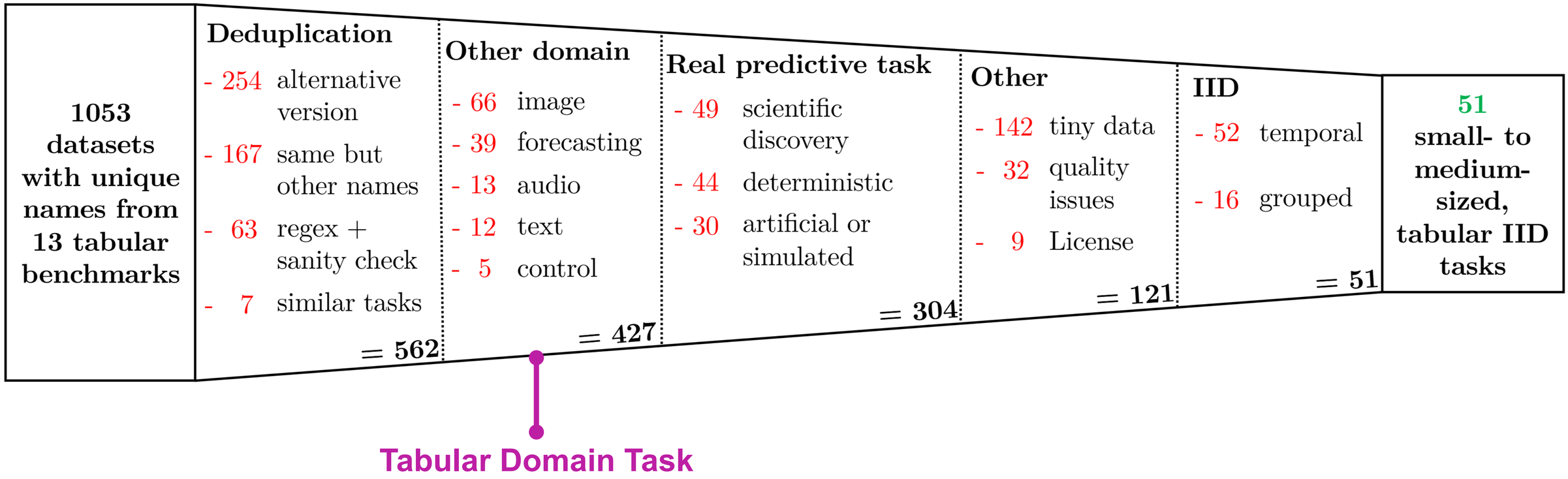
Datasets Curation



- Many surprising duplicates (e.g., AutoML competition datasets)
- Very similar tasks (e.g., 5 datasets from one paper, same features different targets)

Datasets Curation

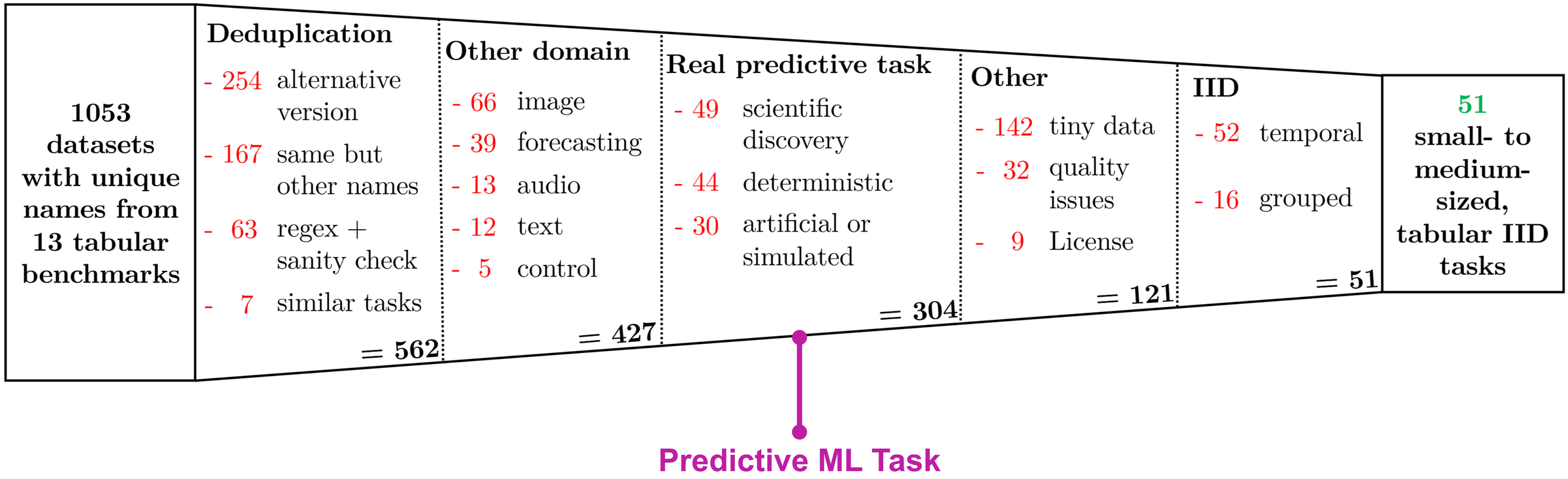
Datasets



- Many datasets that treat images as tables (often very outdated)
- Often, only the original source described the data

Datasets Curation

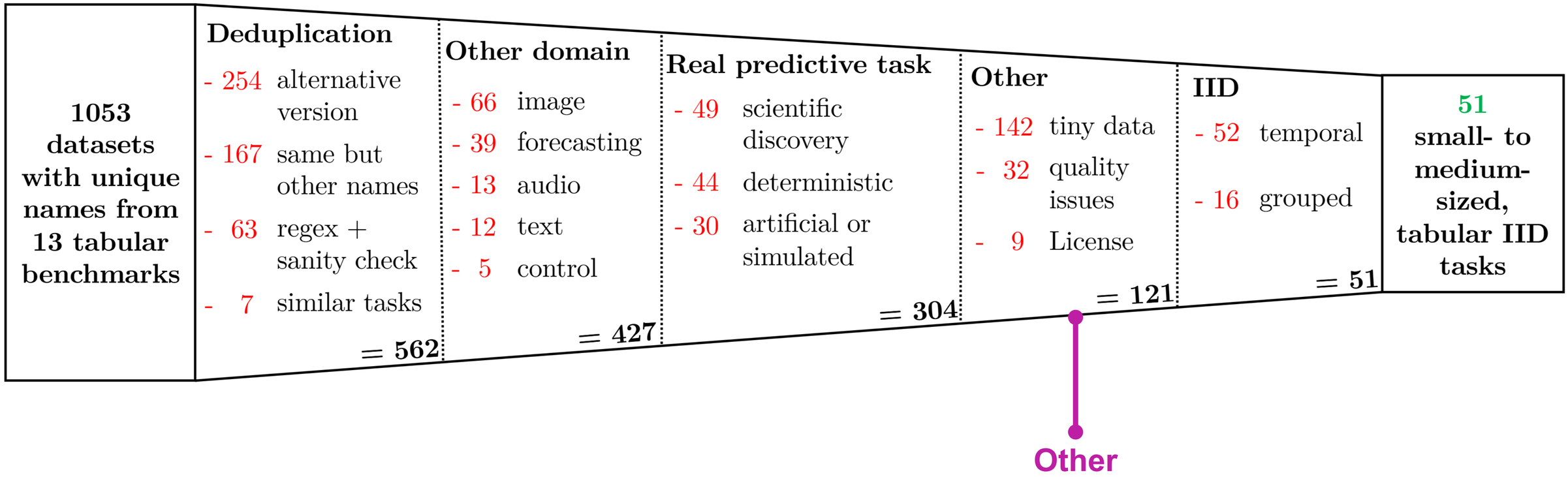
Datasets



- Scientific discovery (why/how questions) vs. predictive task
- Real-world data: not deterministic, not artificial, not simulated

Datasets Curation

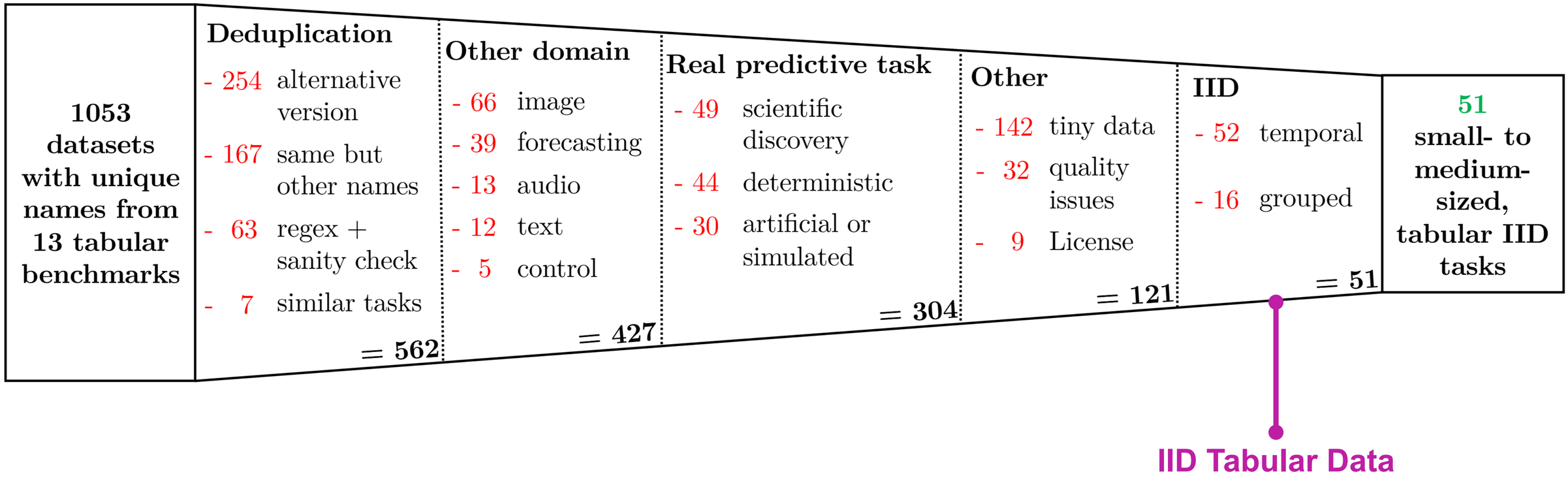
Datasets



- Many tiny (often old) datasets
- Datasets with preprocessing errors (PCA data leakage), missing source information, and target leakage

Datasets Curation

Datasets



- Tasks that require non-random splits
- Temporal-dependent features / grouped data (e.g., algorithm selection)
- Many borderline cases

Datasets Curation

Datasets

1053 datasets with unique names from 13 tabular benchmarks

Deduplication

- 254 alternative version
- 167 same but other names
- 63 regex + sanity check
- 7 similar tasks

= 562

Other domain

- 66 image
- 39 forecasting
- 13 audio
- 12 text
- 5 control

= 427

Real predictive task

- 49 scientific discovery
- 44 deterministic
- 30 artificial or simulated

= 304

Other

- 142 tiny data
- 32 quality issues
- 9 License

= 121

IID

- 52 temporal
- 16 grouped

= 51

51 small- to medium-sized, tabular IID tasks

Check for yourself and verify our curation:
https://tabarena.ai/dataset-curation

Datasets Curation

Datasets

1053 datasets with unique names from 13 tabular benchmarks

Deduplication
- 254 alternative version
- 167 same but other names
- 63 regex + sanity check
- 7 similar tasks

= 562

Other domain
- 66 image
- 39 forecasting
- 13 audio
- 12 text
- 5 control

= 427

Real predictive task
- 49 scientific discovery
- 44 deterministic
- 30 artificial or simulated

= 304

Other
- 142 tiny data
- 32 quality issues
- 9 License

= 121

IID
- 52 temporal
- 16 grouped

= 51

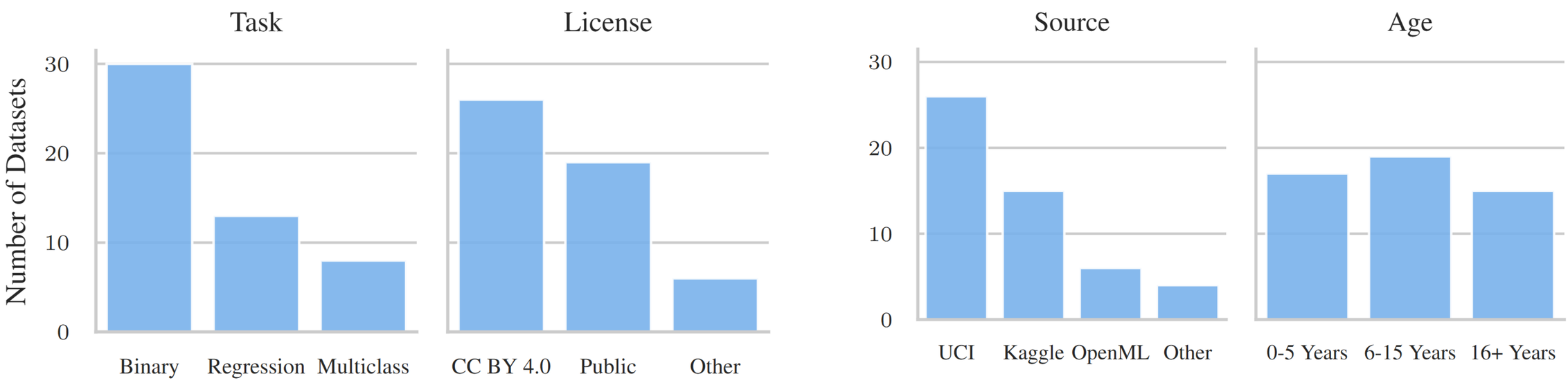
51 small- to medium-sized, tabular IID tasks

Check for yourself and verify our curation:
https://tabarena.ai/dataset-curation

Smaller is better!
Sometimes at least…

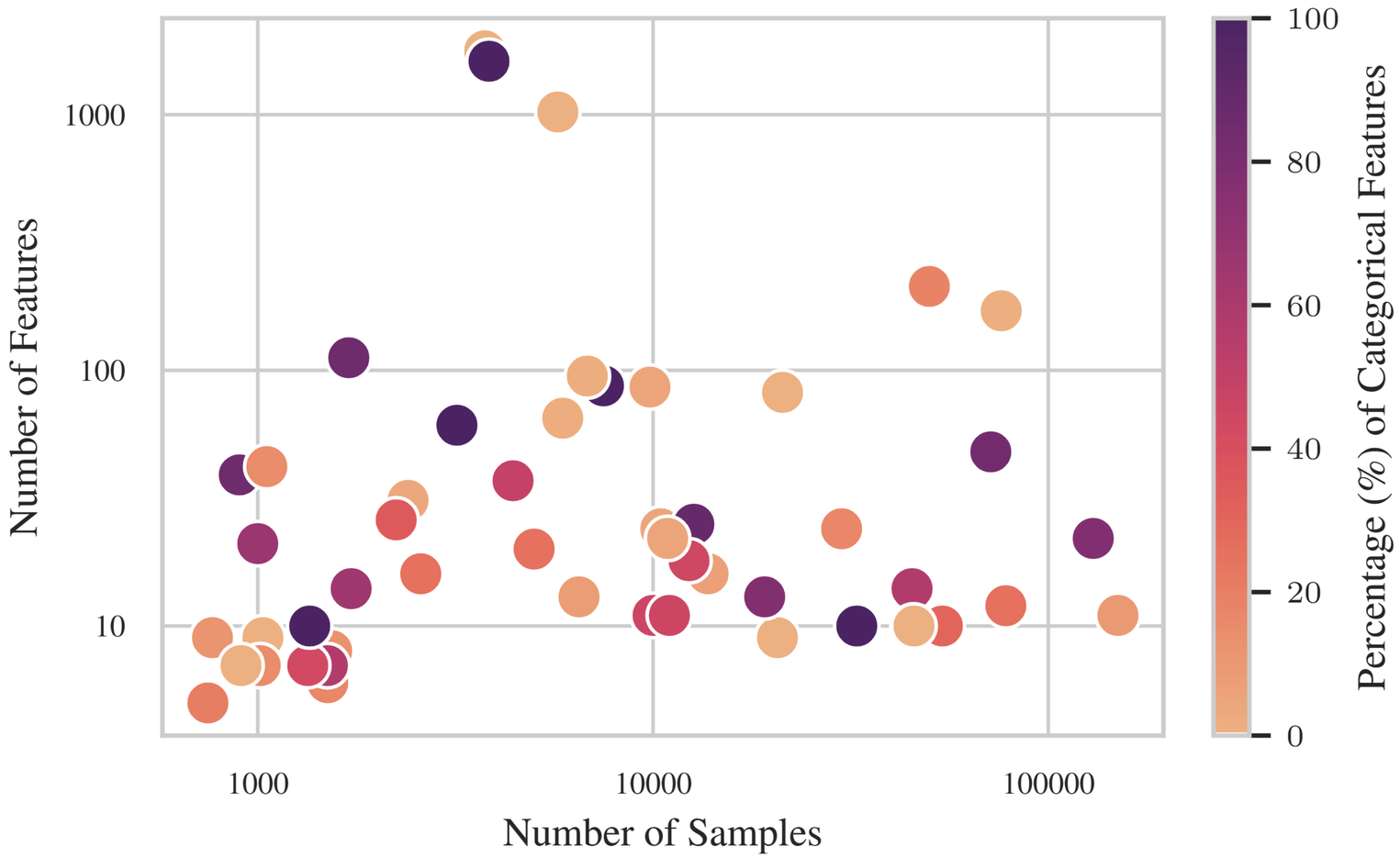
Datasets Overview

Datasets



Datasets Overview

Datasets



Datasets

Compared to Prior Benchmarks

Benchmark	Manual curation	#datasets remaining
Bischl et al. [28, 29]	✗	9/72
Gorishniy et al. [30]	✓	1/11
Shwartz-Ziv and Armon [31]	✗	1/11
Grinsztajn et al. [32]	✓	12/47
McElfresh et al. [33]	✗	13/196
Fischer et al. [34]	✓	8/35
Gijsbers et al. [35]	✓	15/104
Kohli et al. [7]	✓	17/187
Tschalzev et al. [8]	✓	1/10
Holzmüller et al. [20]	✓	10/118
Ye et al. [36]	✗	39/300
Rubachev et al. [10]	✓	0/8
Salinas and Erickson [37]	✗	19/200
TabArena (Ours)	✓	51/51

Focus

Models

Datasets

TabArena-v0.1

Evaluations

Evaluation Design

Evaluations

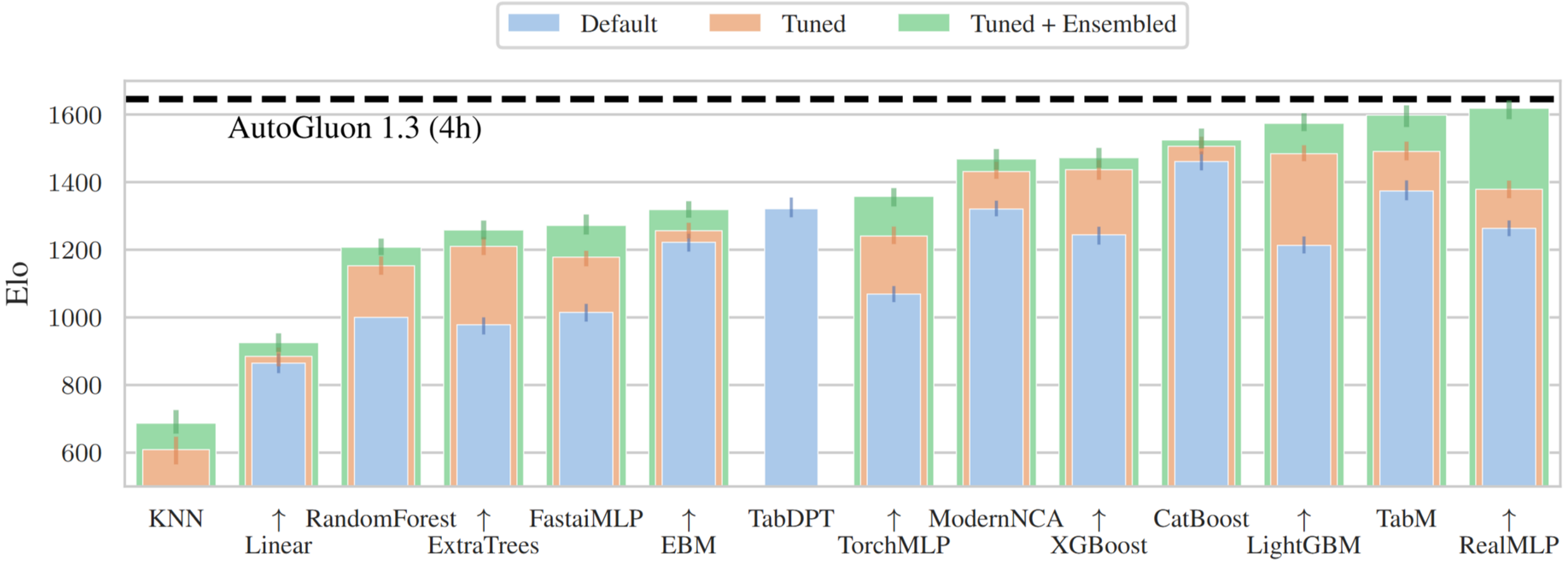
1. **Repeat experiments per dataset:**
   - 30 times for data with less than 2500 samples (10-repeated 3-fold cv)
   - 9 times for all other data (3-repeated 3-fold cv)
2. **Using the Elo rating system**
   - pairwise model comparison
   - 400-point Elo Gap corresponds to a 10 to 1 (91%) win rate
3. **Robust metrics appropriate for benchmarking**
   - Binary: ROC AUC
   - Multiclass: Log Loss
   - Regression: RMSE
4. **Realistic reference pipeline for practitioners**
   - A pipeline practitioners can easily use
   - SOTA AutoML, AutoGluon trained for 4 hours
5. **Store and share extensive metadata**
   - such as: validation predictions (per-fold), test predictions, training time, inference time, precomputed results on various metrics, hyperparameters – “**TabRepo 2.0**”

Evaluations

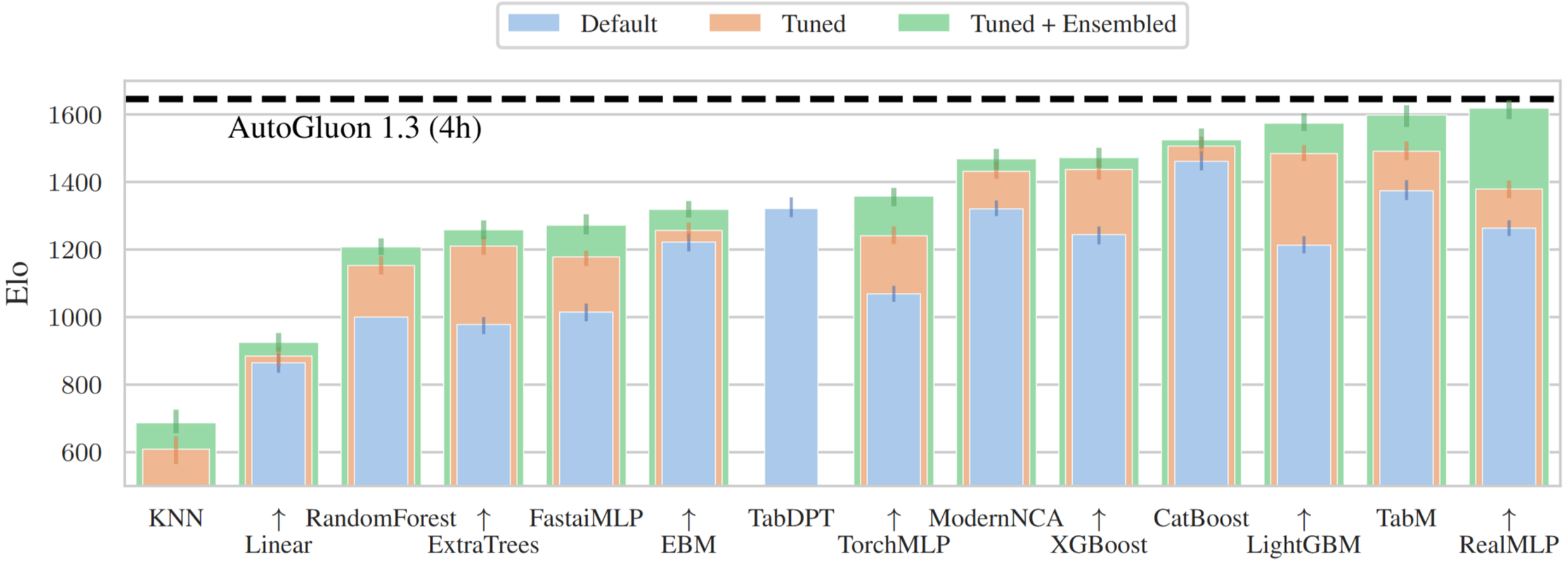
Evaluation Design

Benchmark	#splits inner	#splits outer	Results available
Bischl et al. [28, 29]	1	10	(✓)
Gorishniy et al. [30]	1	1	✗
Shwartz-Ziv and Armon [31]	1	{1, 3}	✗
Grinsztajn et al. [32]	1	{1, 2, 3, 5}	(✓)
McElfresh et al. [33]	1	10	(✓)
Fischer et al. [34]	{1, 3, 10}	{1, 10, 100}	(✓)
Gijsbers et al. [35]	-	10	(✓)
Kohli et al. [7]	1	1	✗
Tschalzev et al. [8]	10	1	✗
Holzmüller et al. [20]	1	10	✓
Ye et al. [36]	1	1	(✓)
Rubachev et al. [10]	1	1	(✓)
Salinas and Erickson [37]	8	3	✓
TabArena (Ours)	8	{9, 30}	✓

Main Results

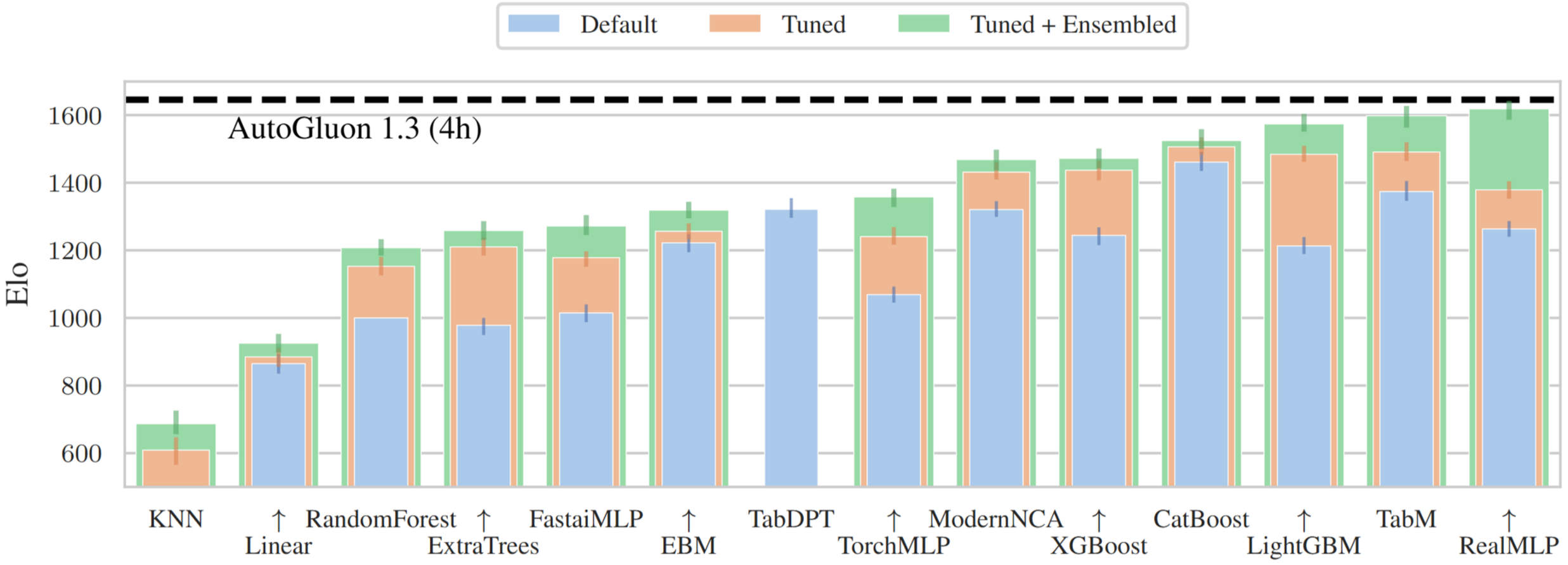


Main Results



CatBoost is best by default and with tuning.

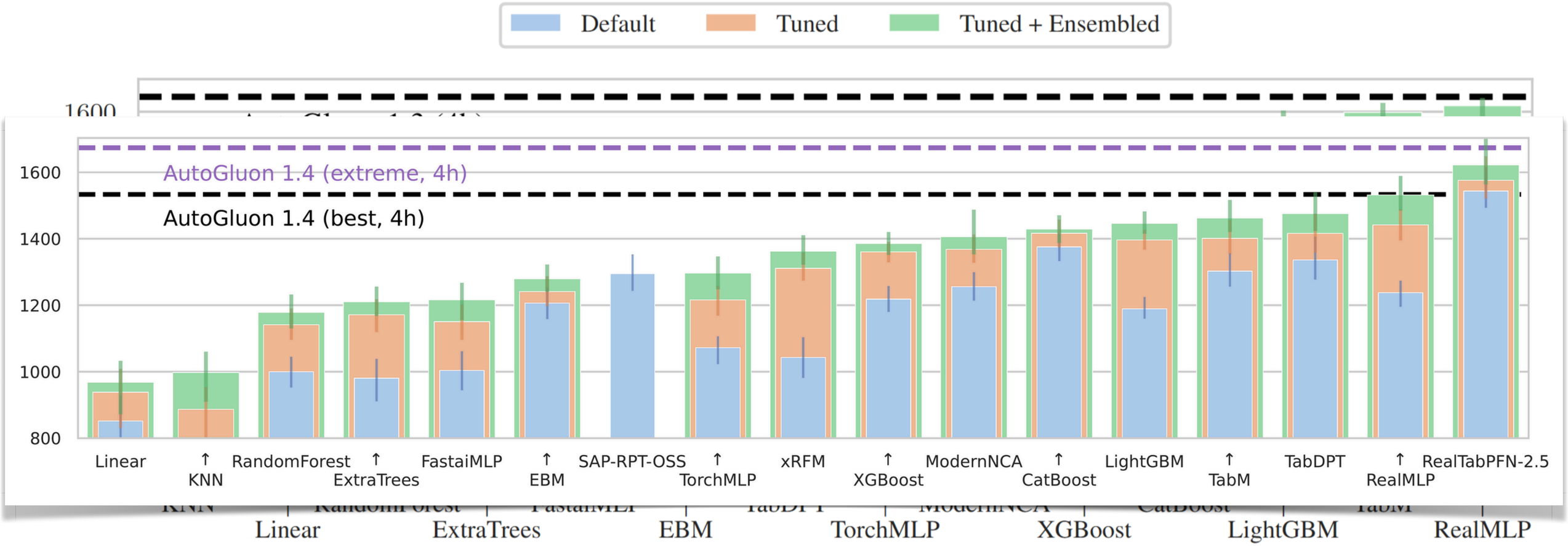
Main Results



CatBoost is best by default and with tuning.

Deep learning models dominate with ensembling.

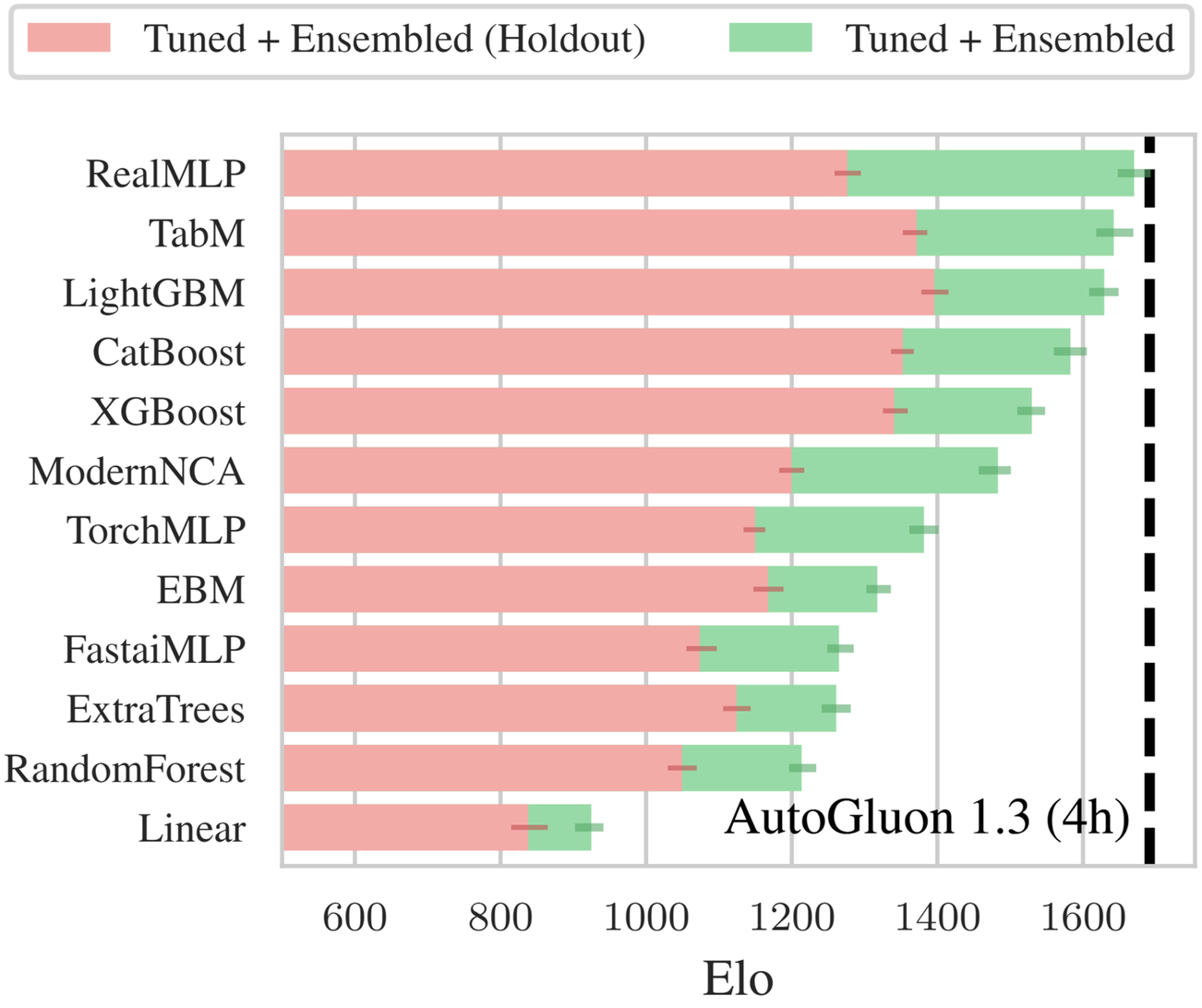
Main Results



CatBoost is best by default and with tuning.

Deep learning models dominate with ensembling.

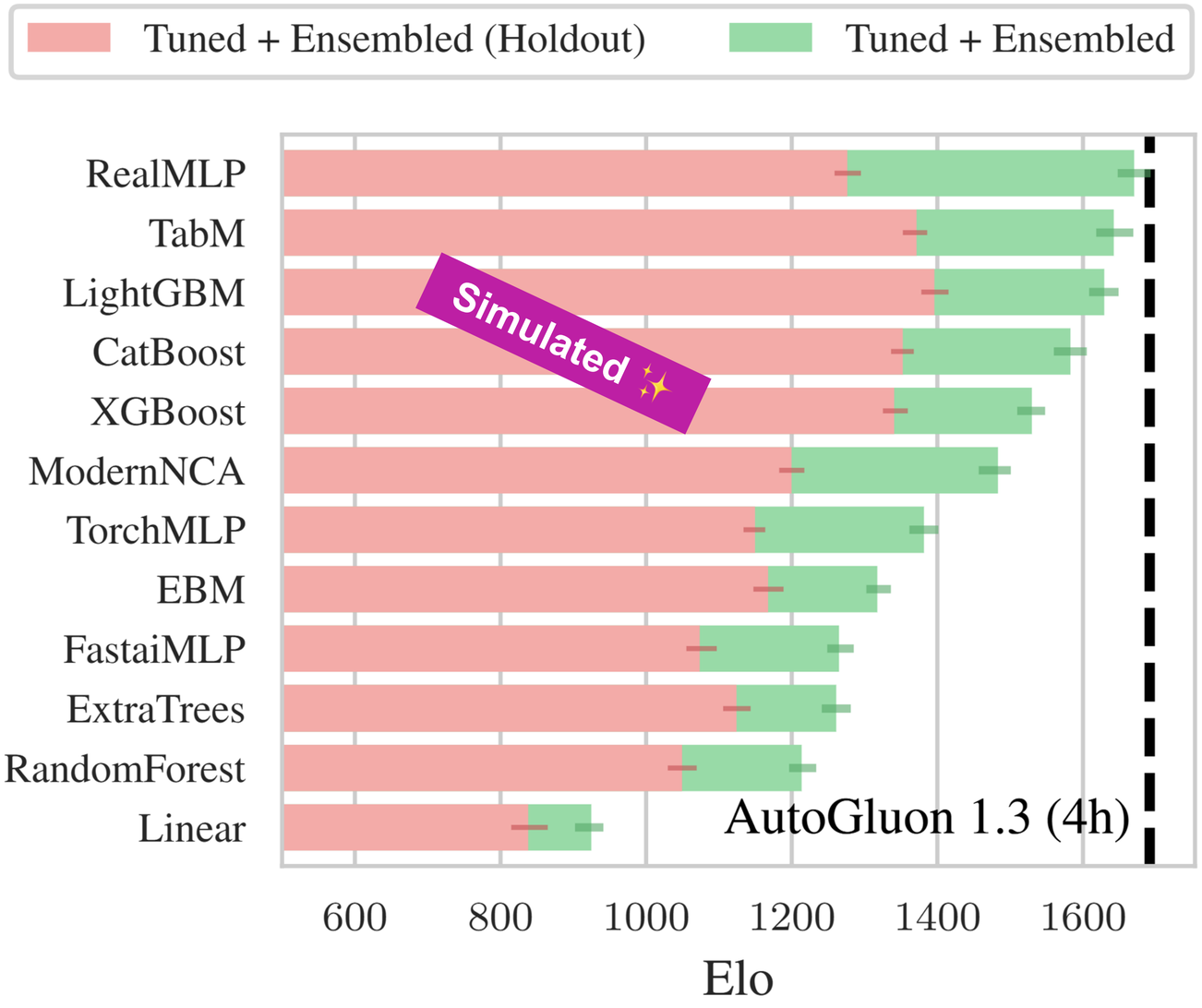
Additional Results: Hold Holdout!



Do not use holdout validation!

- **Worse peak performance** (after HPO + Ensembling)
- Relative **model ranking changes**
- **Unreliable for post-hoc analysis** (e.g., meta-feature analysis)

Additional Results: Hold Holdout!



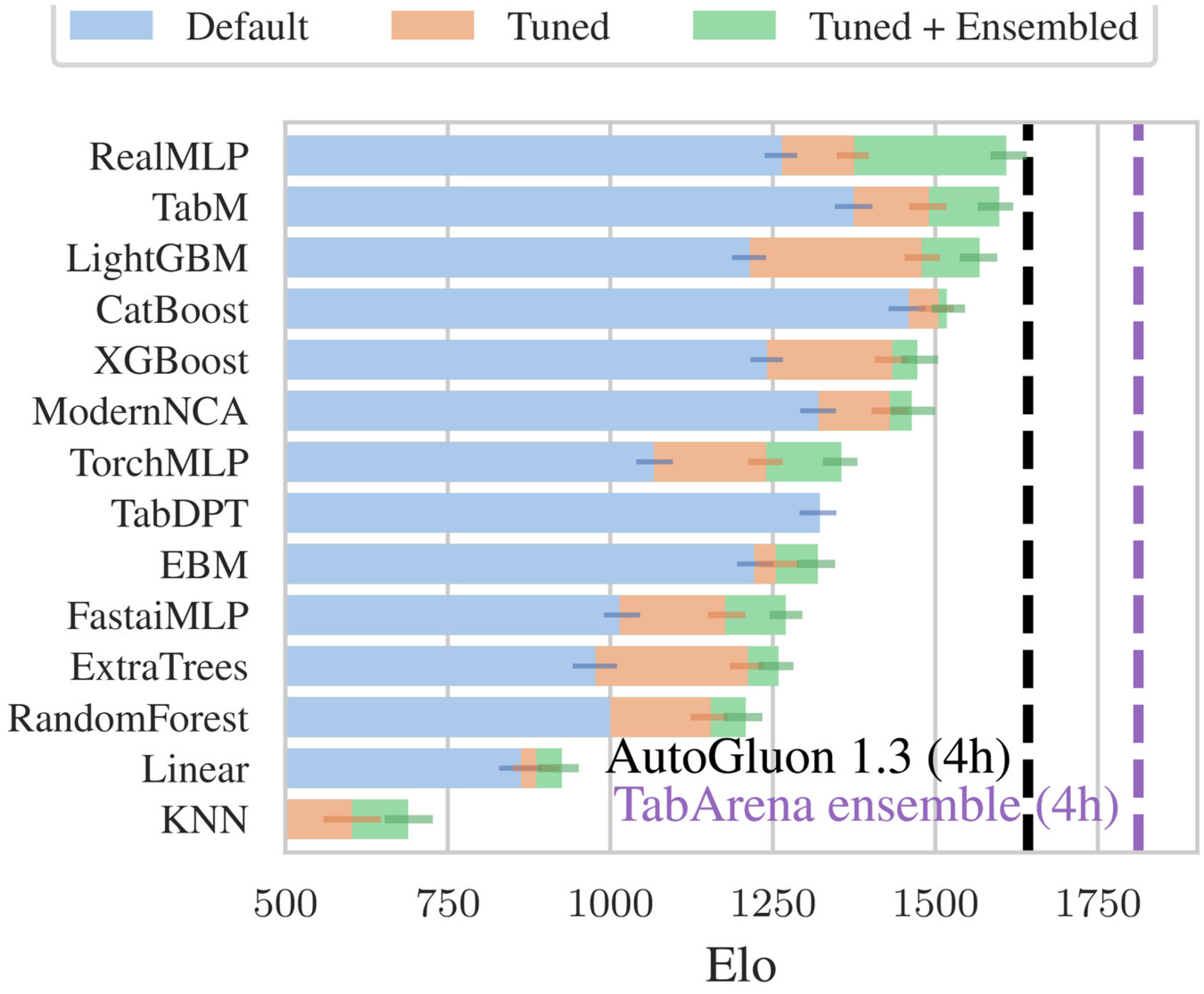
Do not use holdout validation!

- **Worse peak performance** (after HPO + Ensembling)
- Relative **model ranking changes**
- **Unreliable for post-hoc analysis** (e.g., meta-feature analysis)

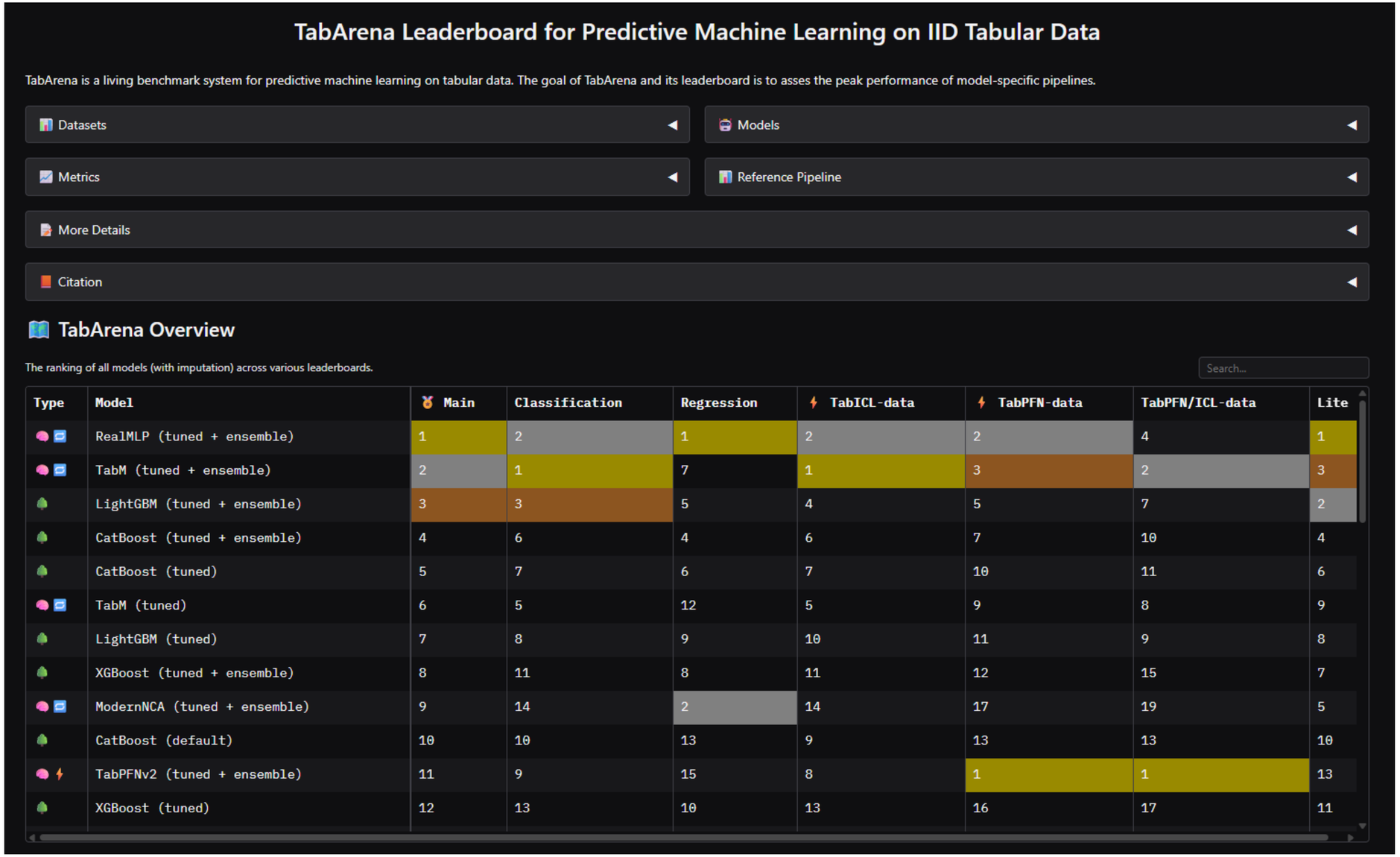
Additional Results: Ensembling

SOTA model-agnostic ensembles!

- Fully **simulated ✨ AutoML system** (AutoGluon-like)
- **Significantly better**, even with 4 hours instead of 200 configs
- **The real research goal**; GBDT vs. Deep learning is “just” framing



Hugging Face Leaderboard: https://tabarena.ai/



TabArena Leaderboard for Predictive Machine Learning on IID Tabular Data

TabArena is a living benchmark system for predictive machine learning on tabular data. The goal of TabArena and its leaderboard is to asses the peak performance of model-specific pipelines.

Datasets ◀ | Models ◀

Metrics ◀ | Reference Pipeline ◀

More Details ◀

Citation ◀

TabArena Overview

The ranking of all models (with imputation) across various leaderboards.

Search...

Type	Model	Main	Classification	Regression	TabICL-data	TabPFN-data	TabPFN/ICL-data	Lite
	RealMLP (tuned + ensemble)	1	2	1	2	2	4	1
	TabM (tuned + ensemble)	2	1	7	1	3	2	3
	LightGBM (tuned + ensemble)	3	3	5	4	5	7	2
	CatBoost (tuned + ensemble)	4	6	4	6	7	10	4
	CatBoost (tuned)	5	7	6	7	10	11	6
	TabM (tuned)	6	5	12	5	9	8	9
	LightGBM (tuned)	7	8	9	10	11	9	8
	XGBoost (tuned + ensemble)	8	11	8	11	12	15	7
	ModernNCA (tuned + ensemble)	9	14	2	14	17	19	5
	CatBoost (default)	10	10	13	9	13	13	10
	TabPFNv2 (tuned + ensemble)	11	9	15	8	1	1	13
	XGBoost (tuned)	12	13	10	13	16	17	11

Living Benchmark: First Steps

[WIP][New Model] TabFlex ✓

#171 opened 4 days ago by LennartPurucker updated 4 days ago

new model

Mitra

#161 opened last month by xiyuanzh updated last week

update to EBM hyperparameters

#158 opened on May 30 by paulbkoch • 1

[WIP][New Model] PerpetualBoosting ✓

#170 opened 4 days ago by LennartPurucker updated 4 days ago

new model

[WIP][New Model] BETA-TabPFN ✓

#172 opened 4 days ago by LennartPurucker

new model

[WIP][New Model] Dynamic Programming Decision Trees

#176 opened 3 days ago by KohlerHECTOR updated 3 days ago 4 tasks

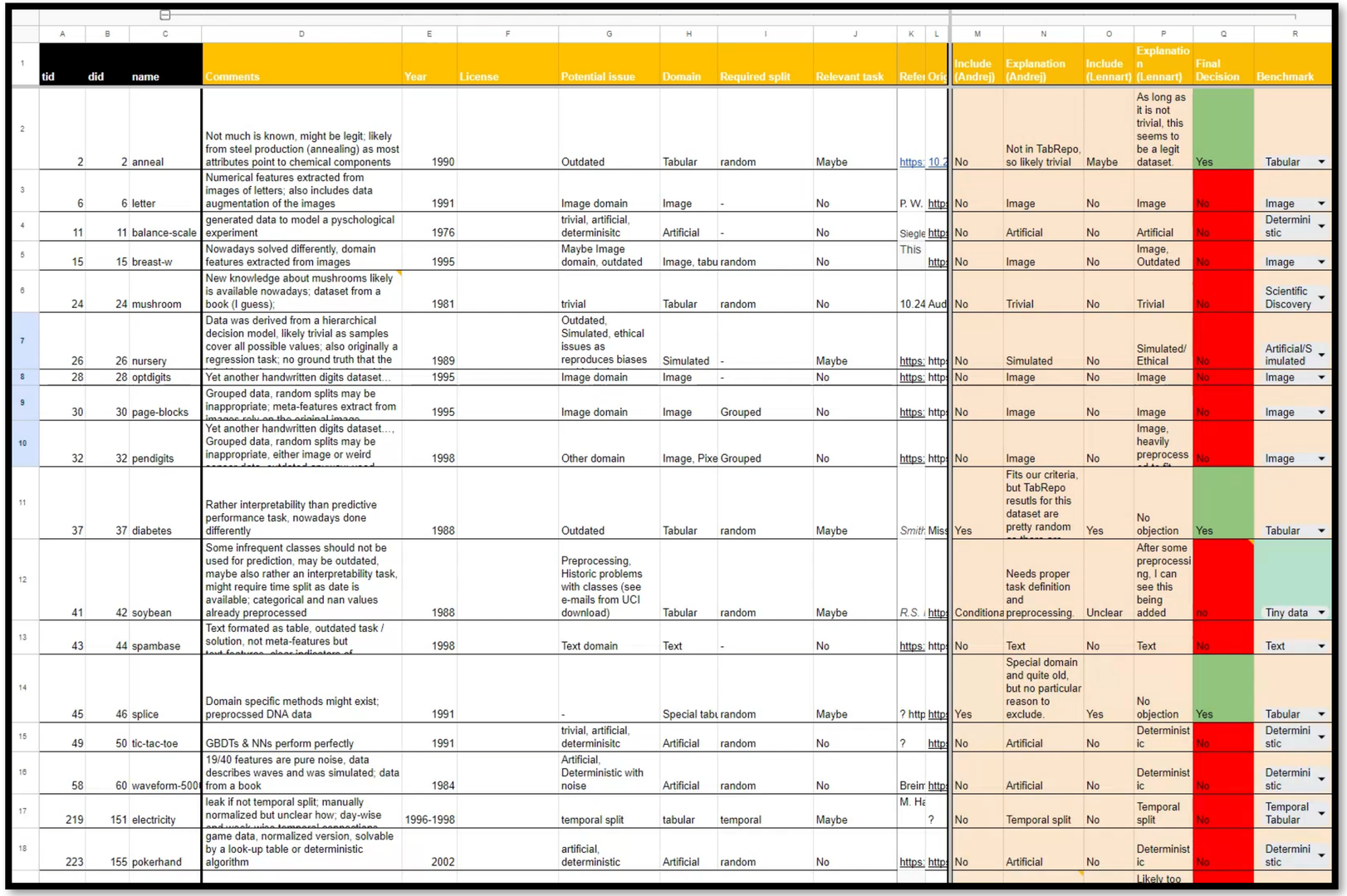
new model

Using all our models – or with the next version of AutoGluon :)

```python
from autogluon.core.data import LabelCleaner
from autogluon.features.generators import AutoMLPipelineFeatureGenerator
from sklearn.datasets import load_breast_cancer
from sklearn.metrics import roc_auc_score
from sklearn.model_selection import train_test_split

# Import a TabArena model
from tabrepo.benchmark.models.ag.realmlp.realmlp_model import RealMLPModel

# Get Data
X, y = load_breast_cancer(return_X_y=True, as_frame=True)
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.5, random_state=42
)
# Preprocessing
feature_generator, label_cleaner = (
    AutoMLPipelineFeatureGenerator(),
    LabelCleaner.construct(problem_type="binary", y=y),
)
X_train, y_train = (
    feature_generator.fit_transform(X_train),
    label_cleaner.transform(y_train),
)
X_test, y_test = feature_generator.transform(X_test), label_cleaner.transform(y_test)

# Train TabArena Model
clf = RealMLPModel()
clf.fit(X=X_train, y=y_train)

# Predict and score
prediction_probabilities = clf.predict_proba(X=X_test)
print("ROC AUC:", roc_auc_score(y_test, prediction_probabilities))
```

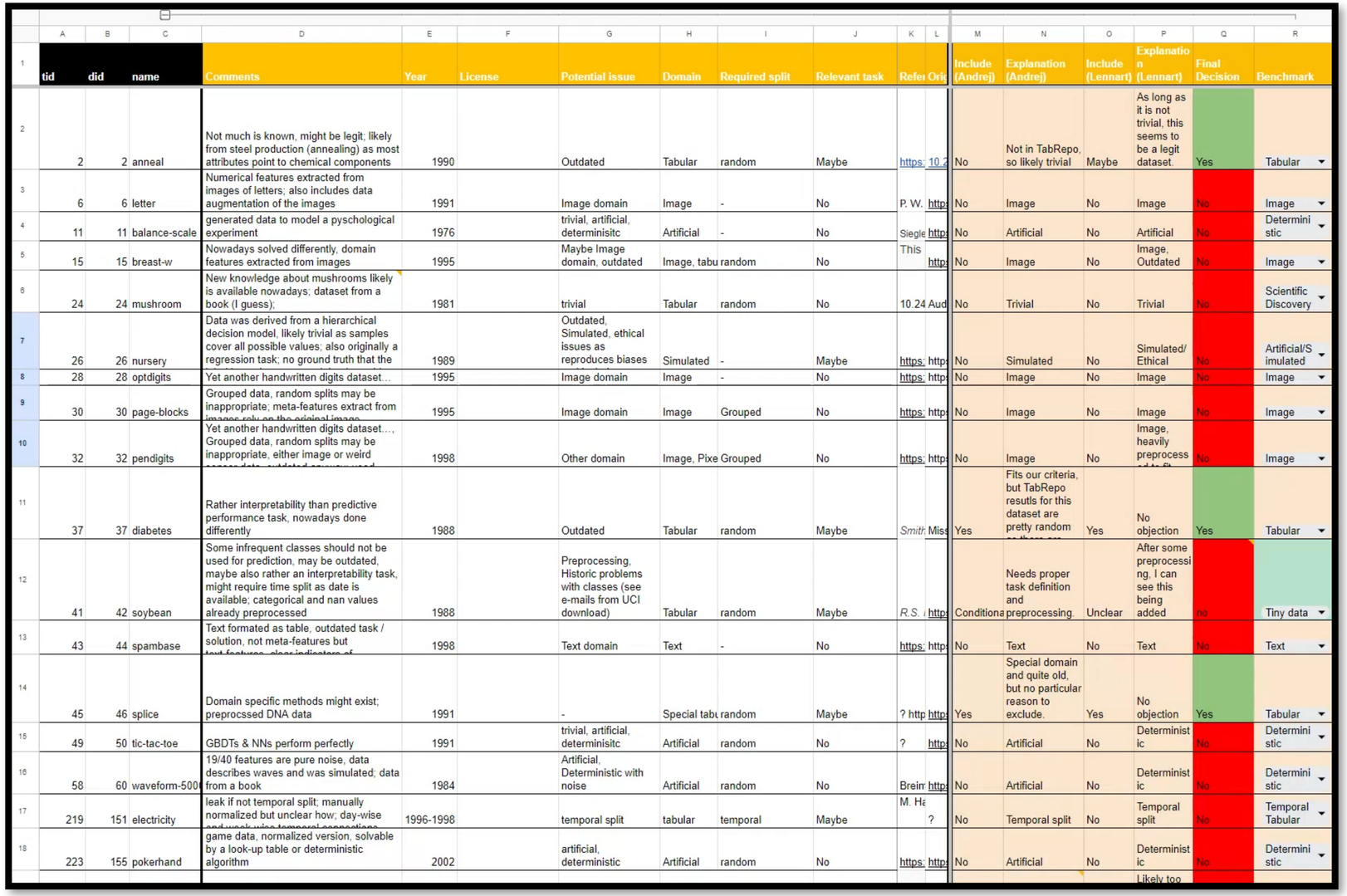
https://tabarena.ai/code-examples

Public Dataset Curation: https://tabarena.ai/dataset-curation



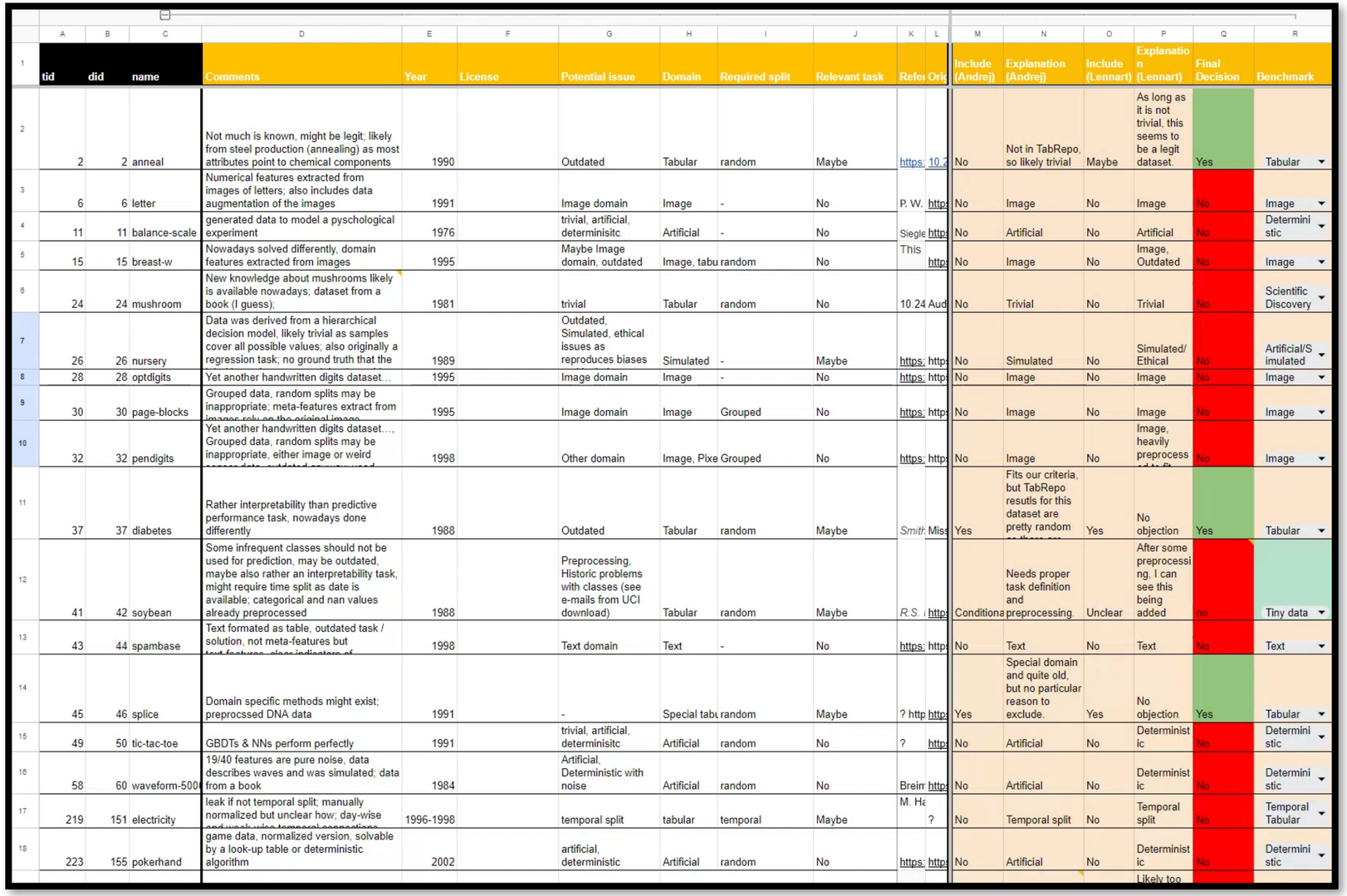
tid	did	name	Comments	Year	License	Potential issue	Domain	Required split	Relevant task	Refer	Orig	Include (Andrej)	Explanation (Andrej)	Include (Lennart)	Explanation (Lennart)	Final Decision	Benchmark
2	2	anneal	Not much is known, might be legit; likely from steel production (annealing) as most attributes point to chemical components	1990		Outdated	Tabular	random	Maybe	https:	10.2	No	Not in TabRepo, so likely trivial	Maybe	As long as it is not trivial, this seems to be a legit dataset.	Yes	Tabular
3	6	letter	Numerical features extracted from images of letters; also includes data augmentation of the images	1991		Image domain	Image	-	No	P. W.	http:	No	Image	No	Image	No	Image
4	11	balance-scale	generated data to model a pyschological experiment	1976		trivial, artificial, determinisitc	Artificial	-	No	Siegle	http:	No	Artificial	No	Artificial	No	Determinstic
5	15	breast-w	Nowadays solved differently, domain features extracted from images	1995		Maybe Image domain, outdated	Image, tabu	random	No	This	http:	No	Image	No	Image, Outdated	No	Image
6	24	mushroom	New knowledge about mushrooms likely is available nowadays; dataset from a book (I guess);	1981		trivial	Tabular	random	No	10.24	Aud	No	Trivial	No	Trivial	No	Scientific Discovery
7	26	nursery	Data was derived from a hierarchical decision model, likely trivial as samples cover all possible values; also originally a regression task; no ground truth that the	1989		Outdated, Simulated, ethical issues as reproduces biases	Simulated	-	Maybe	https:	http:	No	Simulated	No	Simulated/ Ethical	No	Artificial/Simulated
8	28	optdigits	Yet another handwritten digits dataset…	1995		Image domain	Image	-	No	https:	http:	No	Image	No	Image	No	Image
9	30	page-blocks	Grouped data, random splits may be inappropriate; meta-features extract from images rely on the original image	1995		Image domain	Image	Grouped	No	https:	http:	No	Image	No	Image	No	Image
10	32	pendigits	Yet another handwritten digits dataset…, Grouped data, random splits may be inappropriate, either image or weird	1998		Other domain	Image, Pixe	Grouped	No	https:	http:	No	Image	No	Image, heavily preprocess	No	Image
11	37	diabetes	Rather interpretability than predictive performance task, nowadays done differently	1988		Outdated	Tabular	random	Maybe	*Smith*	Miss	Yes	Fits our criteria, but TabRepo resutls for this dataset are pretty random	Yes	No objection	Yes	Tabular
12	41	42 soybean	Some infrequent classes should not be used for prediction, may be outdated, maybe also rather an interpretability task, might require time split as date is available; categorical and nan values already preprocessed	1988		Preprocessing, Historic problems with classes (see e-mails from UCI download)	Tabular	random	Maybe	*R.S.*	http:	Conditiona	Needs proper task definition and preprocessing.	Unclear	After some preprocessing, I can see this being added	no	Tiny data
13	43	44 spambase	Text formated as table, outdated task / solution, not meta-features but	1998		Text domain	Text	-	No	https:	http:	No	Text	No	Text	No	Text
14	45	46 splice	Domain specific methods might exist; preprocssed DNA data	1991		-	Special tabu	random	Maybe	? http	http:	Yes	Special domain and quite old, but no particular reason to exclude.	Yes	No objection	Yes	Tabular
15	49	50 tic-tac-toe	GBDTs & NNs perform perfectly	1991		trivial, artificial, determinisitc	Artificial	random	No	?	http:	No	Artificial	No	Deterministic	No	Determinstic
16	58	60 waveform-500	19/40 features are pure noise, data describes waves and was simulated; data from a book	1984		Artificial, Deterministic with noise	Artificial	random	No	Breim	http:	No	Artificial	No	Deterministic	No	Determinstic
17	219	151 electricity	leak if not temporal split; manually normalized but unclear how; day-wise	1996-1998		temporal split	tabular	temporal	Maybe	M. Ha	?	No	Temporal split	No	Temporal split	No	Temporal Tabular
18	223	155 pokerhand	game data, normalized version, solvable by a look-up table or deterministic algorithm	2002		artificial, deterministic	Artificial	random	No	https:	http:	No	Artificial	No	Deterministic	No	Determinstic
															Likely too		

Public Dataset Curation: https://tabarena.ai/dataset-curation



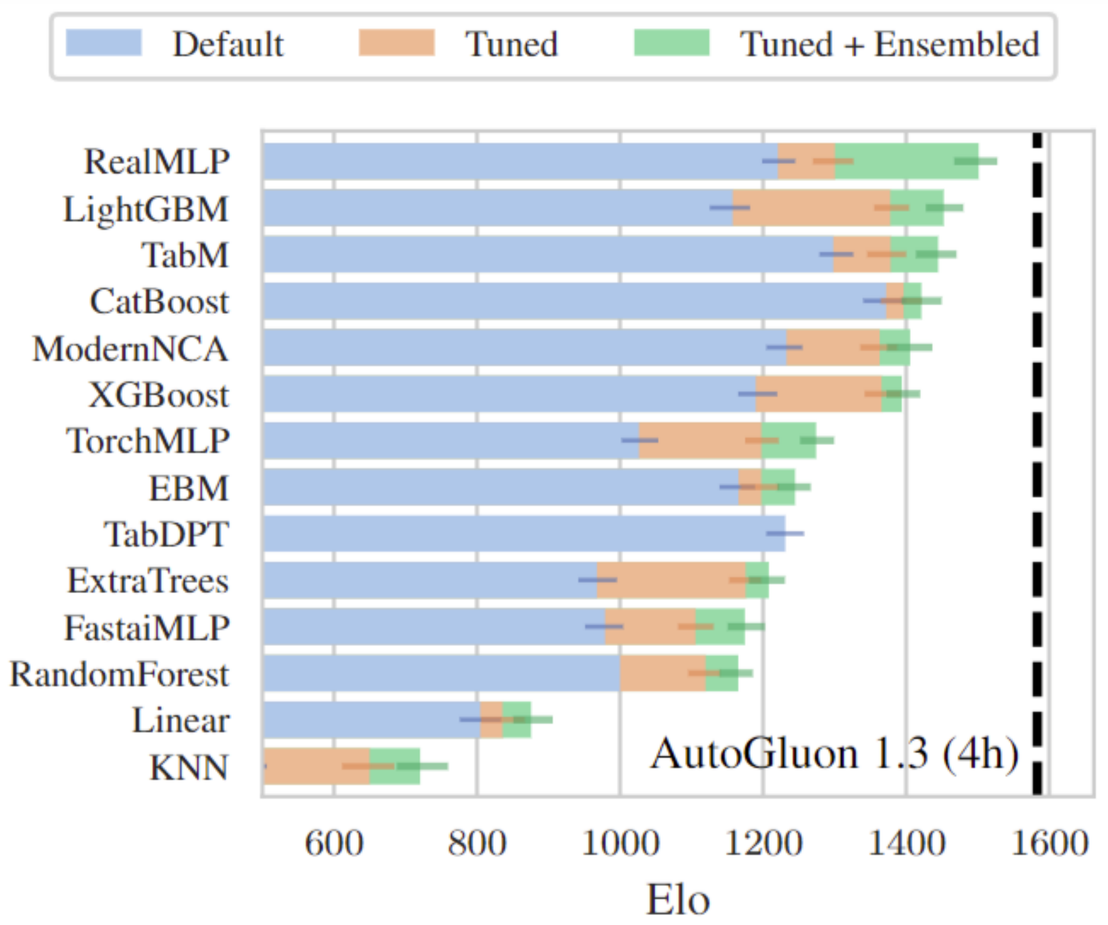
tid	did	name	Comments	Year	License	Potential issue	Domain	Required split	Relevant task	Refer	Orig	Include (Andrej)	Explanation (Andrej)	Include (Lennart)	Explanation (Lennart)	Final Decision	Benchmark
2	2	anneal	Not much is known, might be legit; likely from steel production (annealing) as most attributes point to chemical components	1990		Outdated	Tabular	random	Maybe	https:	10.2	No	Not in TabRepo, so likely trivial	Maybe	As long as it is not trivial, this seems to be a legit dataset.	Yes	Tabular
3	6	letter	Numerical features extracted from images of letters; also includes data augmentation of the images	1991		Image domain	Image	-	No	P. W.	http:	No	Image	No	Image	No	Image
4	11	balance-scale	generated data to model a pyschological experiment	1976		trivial, artificial, determinisitc	Artificial	-	No	Siegle	http:	No	Artificial	No	Artificial	No	Determinstic
5	15	breast-w	Nowadays solved differently, domain features extracted from images	1995		Maybe Image domain, outdated	Image, tabu	random	No	This	http:	No	Image	No	Image, Outdated	No	Image
6	24	mushroom	New knowledge about mushrooms likely is available nowadays; dataset from a book (I guess);	1981		trivial	Tabular	random	No	10.24	Aud	No	Trivial	No	Trivial	No	Scientific Discovery
7	26	nursery	Data was derived from a hierarchical decision model, likely trivial as samples cover all possible values; also originally a regression task; no ground truth that the	1989		Outdated, Simulated, ethical issues as reproduces biases	Simulated	-	Maybe	https:	http:	No	Simulated	No	Simulated/ Ethical	No	Artificial/Simulated
8	28	optdigits	Yet another handwritten digits dataset…	1995		Image domain	Image	-	No	https:	http:	No	Image	No	Image	No	Image
9	30	page-blocks	Grouped data, random splits may be inappropriate; meta-features extract from images rely on the original image	1995		Image domain	Image	Grouped	No	https:	http:	No	Image	No	Image	No	Image
10	32	pendigits	Yet another handwritten digits dataset…, Grouped data, random splits may be inappropriate, either image or weird	1998		Other domain	Image, Pixe	Grouped	No	https:	http:	No	Image	No	Image, heavily preprocess	No	Image
11	37	diabetes	Rather interpretability than predictive performance task, nowadays done differently	1988		Outdated	Tabular	random	Maybe	*Smith*	Miss	Yes	Fits our criteria, but TabRepo resutls for this dataset are pretty random	Yes	No objection	Yes	Tabular
12	41	42 soybean	Some infrequent classes should not be used for prediction, may be outdated, maybe also rather an interpretability task, might require time split as date is available; categorical and nan values already preprocessed	1988		Preprocessing, Historic problems with classes (see e-mails from UCI download)	Tabular	random	Maybe	*R.S.*	http:	Conditiona	Needs proper task definition and preprocessing.	Unclear	After some preprocessing, I can see this being added	no	Tiny data
13	43	44 spambase	Text formated as table, outdated task / solution, not meta-features but	1998		Text domain	Text	-	No	https:	http:	No	Text	No	Text	No	Text
14	45	46 splice	Domain specific methods might exist; preprocssed DNA data	1991		-	Special tabu	random	Maybe	? http	http:	Yes	Special domain and quite old, but no particular reason to exclude.	Yes	No objection	Yes	Tabular
15	49	50 tic-tac-toe	GBDTs & NNs perform perfectly	1991		trivial, artificial, determinisitc	Artificial	random	No	?	http:	No	Artificial	No	Deterministic	No	Determinstic
16	58	60 waveform-500	19/40 features are pure noise, data describes waves and was simulated; data from a book	1984		Artificial, Deterministic with noise	Artificial	random	No	Breim	http:	No	Artificial	No	Deterministic	No	Determinstic
17	219	151 electricity	leak if not temporal split; manually normalized but unclear how; day-wise	1996-1998		temporal split	tabular	temporal	Maybe	M. Ha	?	No	Temporal split	No	Temporal split	No	Temporal Tabular
18	223	155 pokerhand	game data, normalized version, solvable by a look-up table or deterministic algorithm	2002		artificial, deterministic	Artificial	random	No	https:	http:	No	Artificial	No	Deterministic	No	Determinstic
															Likely too		

Public Dataset Curation: https://tabarena.ai/dataset-curation



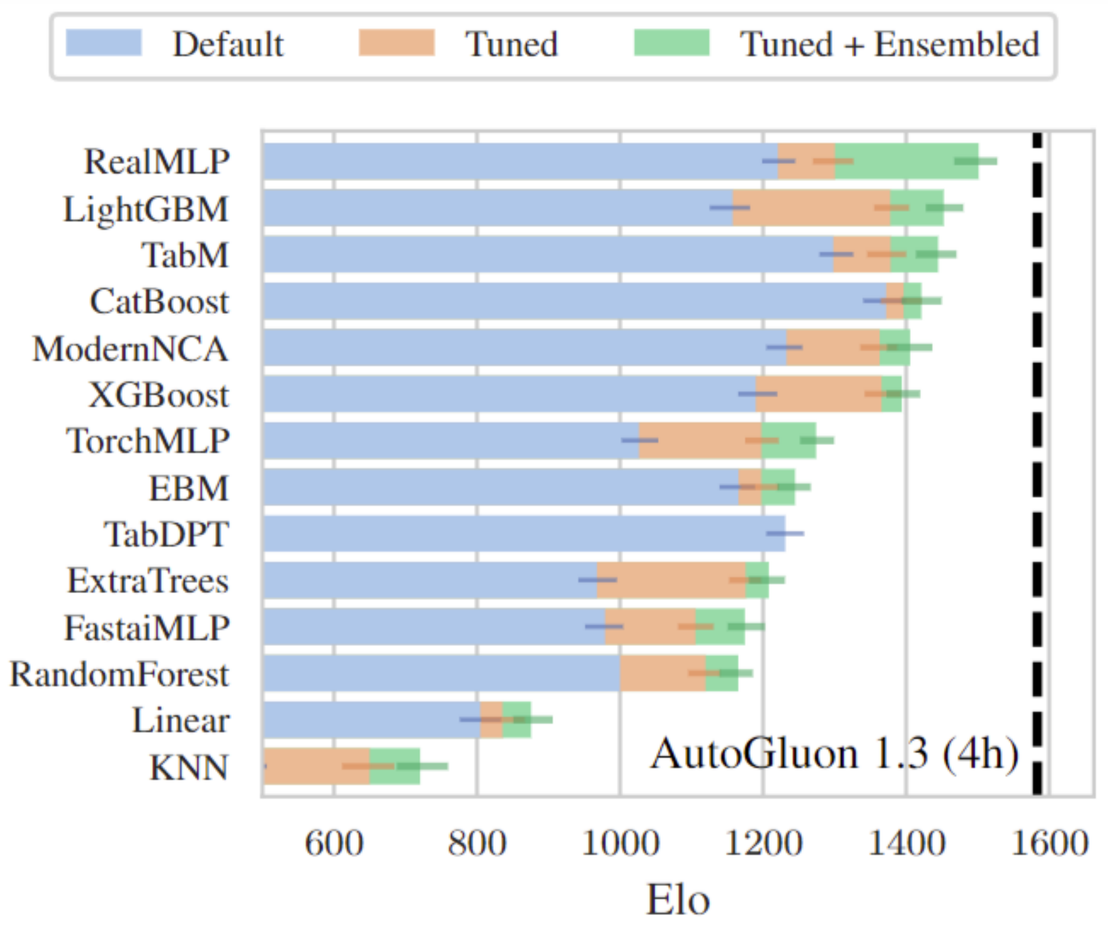
tid	did	name	Comments	Year	License	Potential issue	Domain	Required split	Relevant task	Refer	Orig	Include (Andrej)	Explanation (Andrej)	Include (Lennart)	Explanation (Lennart)	Final Decision	Benchmark
2	2	anneal	Not much is known, might be legit; likely from steel production (annealing) as most attributes point to chemical components	1990		Outdated	Tabular	random	Maybe	https:	10.2	No	Not in TabRepo, so likely trivial	Maybe	As long as it is not trivial, this seems to be a legit dataset.	Yes	Tabular
3	6	letter	Numerical features extracted from images of letters; also includes data augmentation of the images	1991		Image domain	Image	-	No	P. W.	http:	No	Image	No	Image	No	Image
4	11	balance-scale	generated data to model a pyschological experiment	1976		trivial, artificial, determinisitc	Artificial	-	No	Siegle	http:	No	Artificial	No	Artificial	No	Determinstic
5	15	breast-w	Nowadays solved differently, domain features extracted from images	1995		Maybe Image domain, outdated	Image, tabu	random	No	This	http:	No	Image	No	Image, Outdated	No	Image
6	24	mushroom	New knowledge about mushrooms likely is available nowadays; dataset from a book (I guess);	1981		trivial	Tabular	random	No	10.24	Aud	No	Trivial	No	Trivial	No	Scientific Discovery
7	26	nursery	Data was derived from a hierarchical decision model, likely trivial as samples cover all possible values; also originally a regression task; no ground truth that the	1989		Outdated, Simulated, ethical issues as reproduces biases	Simulated	-	Maybe	https:	http:	No	Simulated	No	Simulated/ Ethical	No	Artificial/Simulated
8	28	optdigits	Yet another handwritten digits dataset…	1995		Image domain	Image	-	No	https:	http:	No	Image	No	Image	No	Image
9	30	page-blocks	Grouped data, random splits may be inappropriate; meta-features extract from images rely on the original image	1995		Image domain	Image	Grouped	No	https:	http:	No	Image	No	Image	No	Image
10	32	pendigits	Yet another handwritten digits dataset…, Grouped data, random splits may be inappropriate, either image or weird	1998		Other domain	Image, Pixe	Grouped	No	https:	http:	No	Image	No	Image, heavily preprocess	No	Image
11	37	diabetes	Rather interpretability than predictive performance task, nowadays done differently	1988		Outdated	Tabular	random	Maybe	*Smith*	Miss	Yes	Fits our criteria, but TabRepo resutls for this dataset are pretty random	Yes	No objection	Yes	Tabular
12	41	42 soybean	Some infrequent classes should not be used for prediction, may be outdated, maybe also rather an interpretability task, might require time split as date is available; categorical and nan values already preprocessed	1988		Preprocessing, Historic problems with classes (see e-mails from UCI download)	Tabular	random	Maybe	*R.S.*	http:	Conditiona	Needs proper task definition and preprocessing.	Unclear	After some preprocessing, I can see this being added	no	Tiny data
13	43	44 spambase	Text formated as table, outdated task / solution, not meta-features but	1998		Text domain	Text	-	No	https:	http:	No	Text	No	Text	No	Text
14	45	46 splice	Domain specific methods might exist; preprocssed DNA data	1991		-	Special tabu	random	Maybe	? http	http:	Yes	Special domain and quite old, but no particular reason to exclude.	Yes	No objection	Yes	Tabular
15	49	50 tic-tac-toe	GBDTs & NNs perform perfectly	1991		trivial, artificial, determinisitc	Artificial	random	No	?	http:	No	Artificial	No	Deterministic	No	Determinstic
16	58	60 waveform-500	19/40 features are pure noise, data describes waves and was simulated; data from a book	1984		Artificial, Deterministic with noise	Artificial	random	No	Breim	http:	No	Artificial	No	Deterministic	No	Determinstic
17	219	151 electricity	leak if not temporal split; manually normalized but unclear how; day-wise	1996-1998		temporal split	tabular	temporal	Maybe	M. Ha	?	No	Temporal split	No	Temporal split	No	Temporal Tabular
18	223	155 pokerhand	game data, normalized version, solvable by a look-up table or deterministic algorithm	2002		artificial, deterministic	Artificial	random	No	https:	http:	No	Artificial	No	Deterministic	No	Determinstic
															Likely too		

Cheaper Evaluation For Papers: TabArena Lite



Only one repeat: 816× fewer jobs

Cheaper Evaluation For Papers: TabArena Lite



Benchmarking TabFlex with TabArena Lite takes about 20 minutes

Only one repeat: 816× fewer jobs

Takeaways

Benchmarks ❤️	TabArena is a truly representative benchmark for machine learning on small- to medium sized IID tabular data.
SOTA with Ensembling 📈	CatBoost shines. Deep learning with ensembling dominates. Promising future for foundation models!
Living benchmark baby!	TabArena will be updated and support more (non-IID) data, models, and tasks.

Thank you, any questions?

Leaderboard: https://tabarena.ai
Paper: https://arxiv.org/abs/2506.16791
Code: https://tabarena.ai/code

Nick Erickson

Lennart Purucker

Andrej Tschalzev

David Holzmüller

Prateek Mutalik Desai

David Salinas

Frank Hutter

Thank you, any questions?

- **EquiTabPFN**
  - Paper: https://neurips.cc/virtual/2025/poster/118521
  - Code: https://github.com/MichaelArbel/EquiTabPFN
- **TabRepo**
  - Paper: https://proceedings.mlr.press/v256/salinas24a.html
  - Code: https://github.com/autogluon/tabrepo
- **TabPFN-TS**
  - Paper: https://arxiv.org/abs/2501.02945
  - Code: https://github.com/PriorLabs/tabpfn-time-series
- **TabArena**
  - Leaderboard: https://tabarena.ai
  - Paper: https://arxiv.org/abs/2506.16791
  - Code: https://tabarena.ai/code