Tabular Foundation Models for particle physics with FAIR Universe HiggsML Dataset

Ahmad Asfour[1,2] Ragansu Chakkappai[1,3],

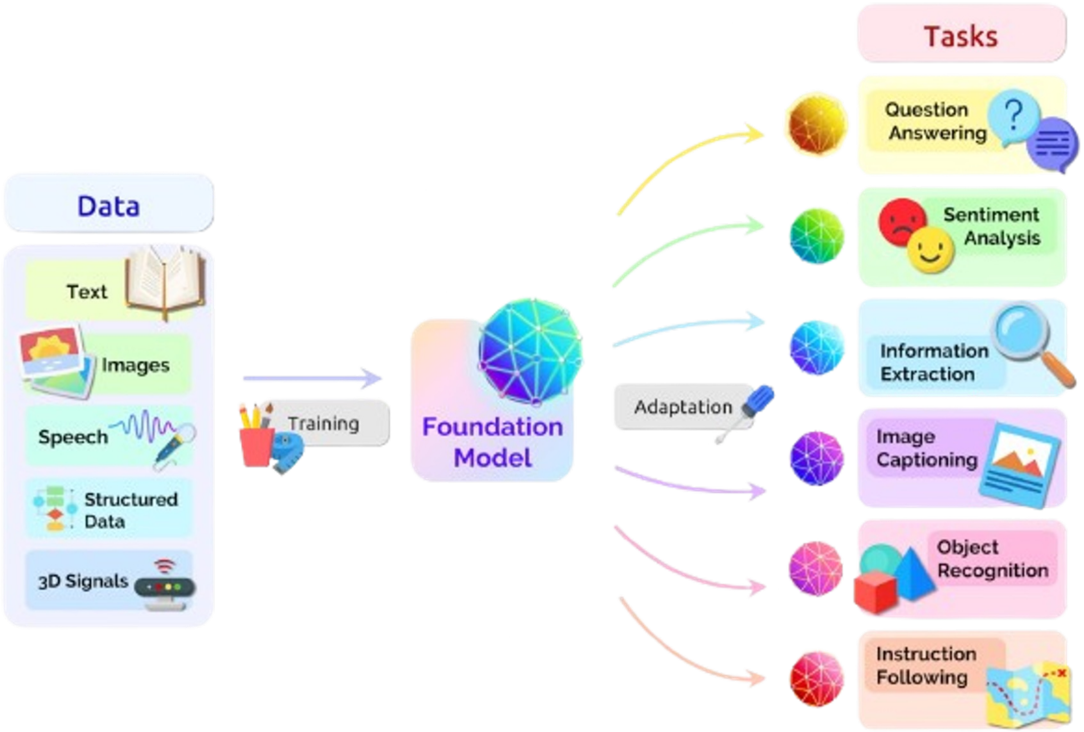
David Rousseau[1,3]

1. CNRS/IN2P3, IJCLab, France
2. An-Najah National University, Palestine
3. Université Paris-Saclay, France

ML4Jets, Sep 2026, Vienna

Foundational Model

See Anna Hallin's talk



Trained on large unrelated collections of data

Used for many tasks unrelated to original training

Foundation Model for Tabular Data

nature

Explore content ⌄ **About the journal** ⌄ **Publish with us** ⌄

nature › articles › article

TabPFN @Prior Labs, bought in Jul 2026 by SAP
(huge Entreprise Resource-Planning company) for ~1 billion dollar

Article | Open access | Published: 08 January 2025

Accurate predictions on small data with a tabular foundation model

Noah Hollmann, Samuel Müller, Lennart Purucker, Arju
Robin Tibor Schirrmeister & Frank Hutter

Nature **637**, 319–326 (2025) | Cite this article

← Go to **ICML 2025 Conference** homepage

2 Versions ▾

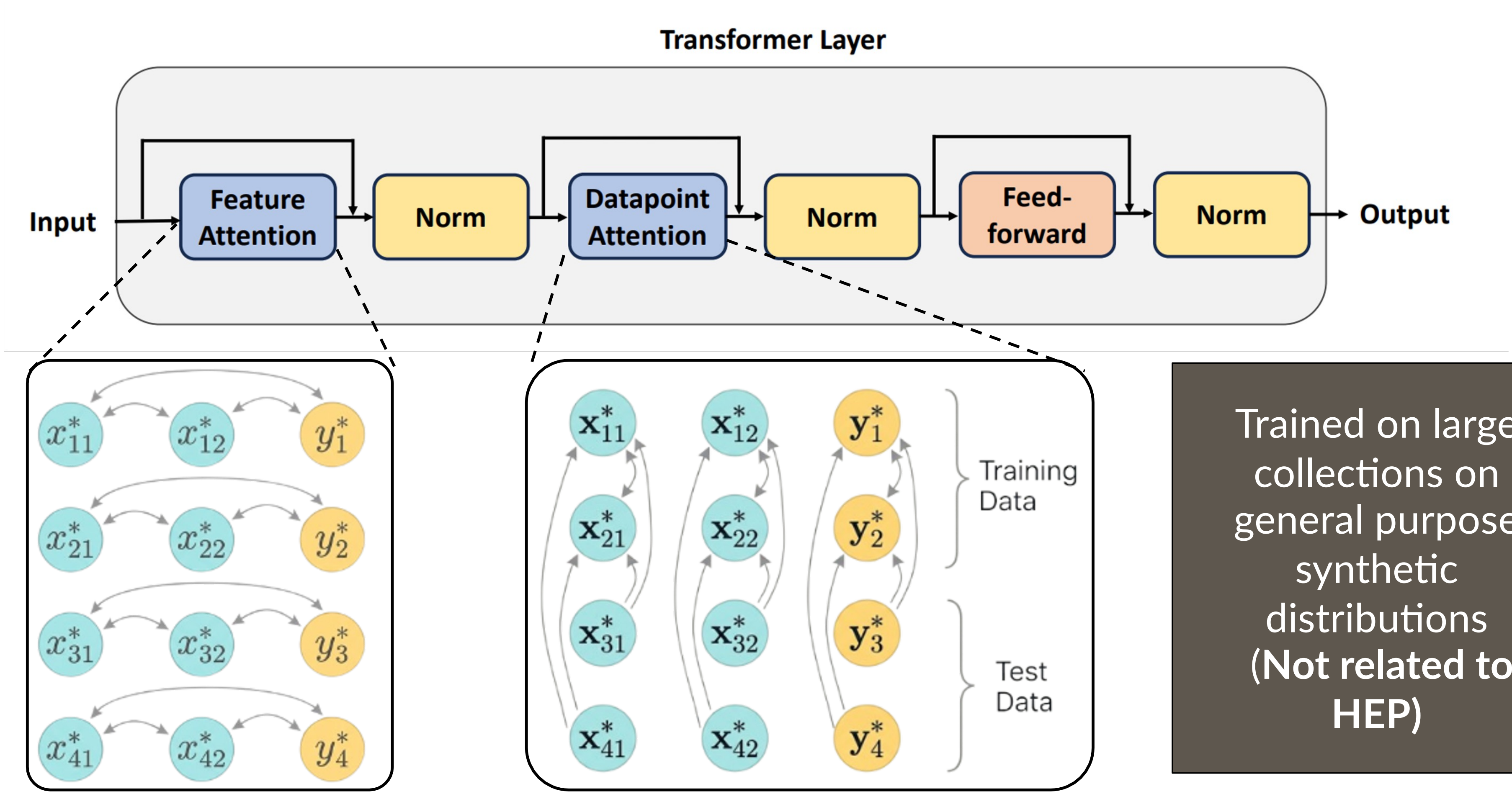
@INRIA Saclay, academics

TabICL: A Tabular Foundation Model for In-Context Learning on Large Data

Jingang QU, David Holzmüller, Gaël Varoquaux, Marine Le Morvan

Published: 01 May 2025, Last Modified: 23 Jul 2025 ICML 2025 poster Everyone
Revisions BibTeX CC BY 4.0

Foundation Model for Tabular Data



Trained on large collections on general purpose synthetic distributions **(Not related to HEP)**

Benchmarking Tabular Foundation Models (Jun 2026)

SPOTLIGHT POSTER | Thu, Dec 4, 2025 · 11:00 AM – 2:00 PM PST

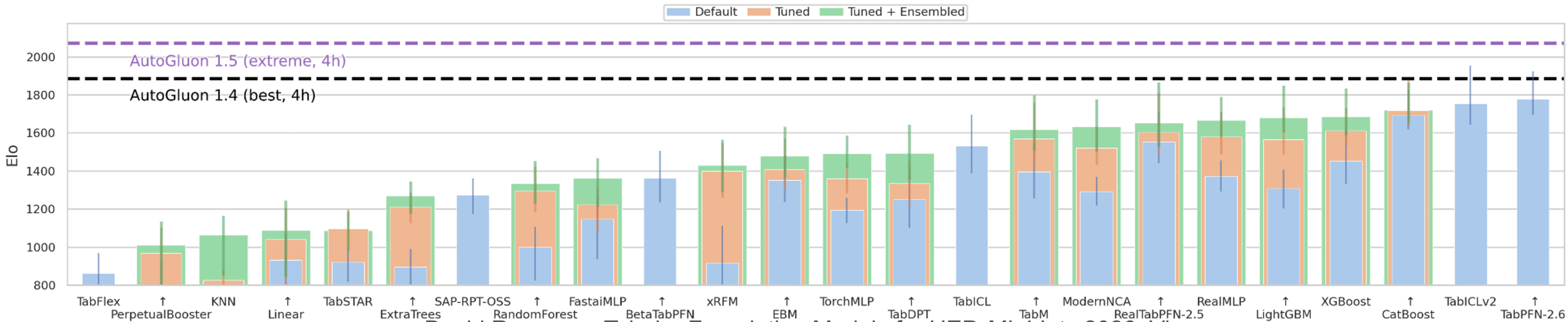
TabArena: A Living Benchmark for Machine Learning on Tabular Data

Nick Erickson · Lennart Purucker · Andrej Tschalzev · David Holzmüller · Prateek Desai · David Salinas · Frank Hutter

NEURAL INFORMATION PROCESSING SYSTEMS

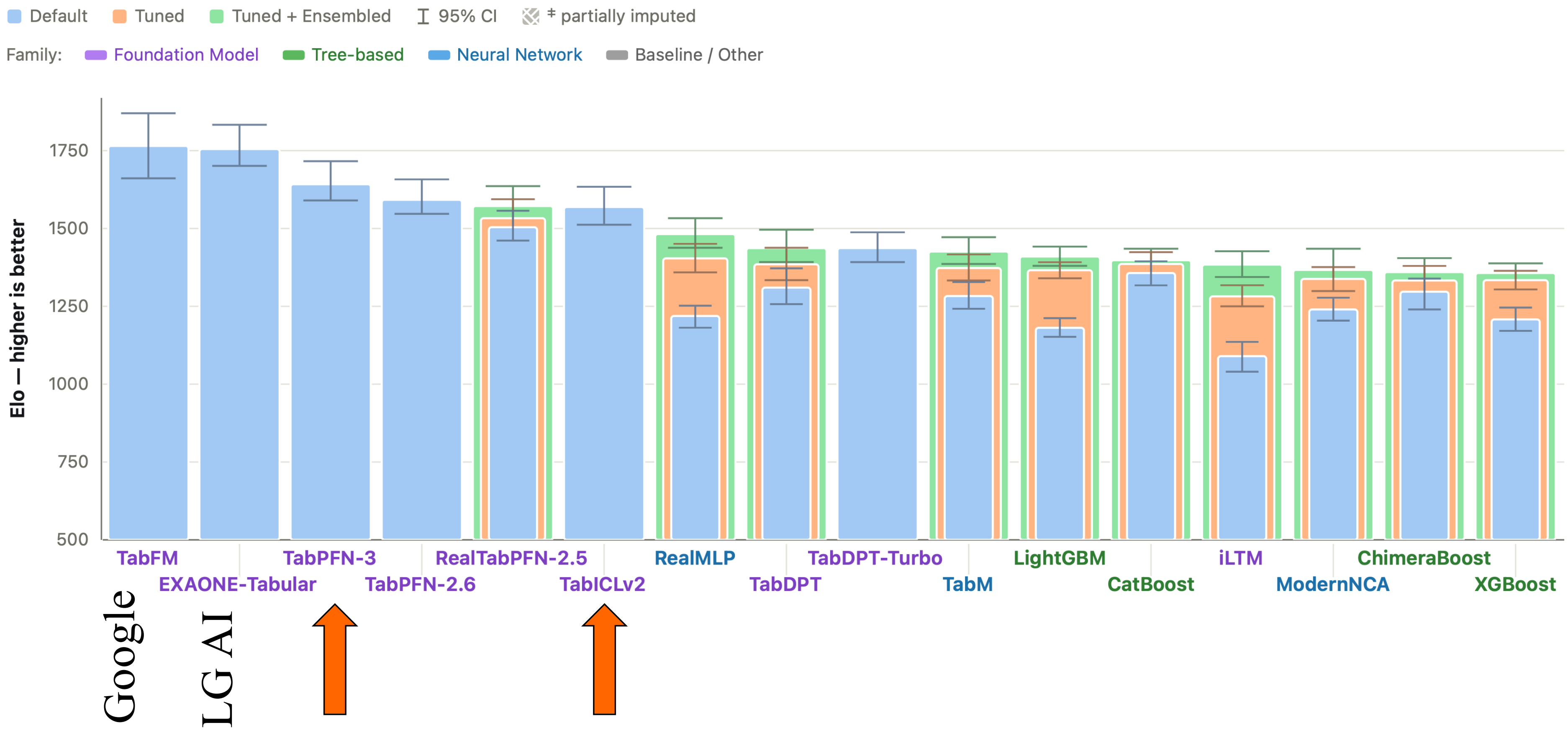
Elo ranking for model competition over many dataset

https://huggingface.co/spaces/TabArena/leaderboard



Benchmarking Tabular Foundation Models (16 Sep 2026)

Elo ranking for model competition over many datasets



Hands on Example

```
pip install xgboost
```

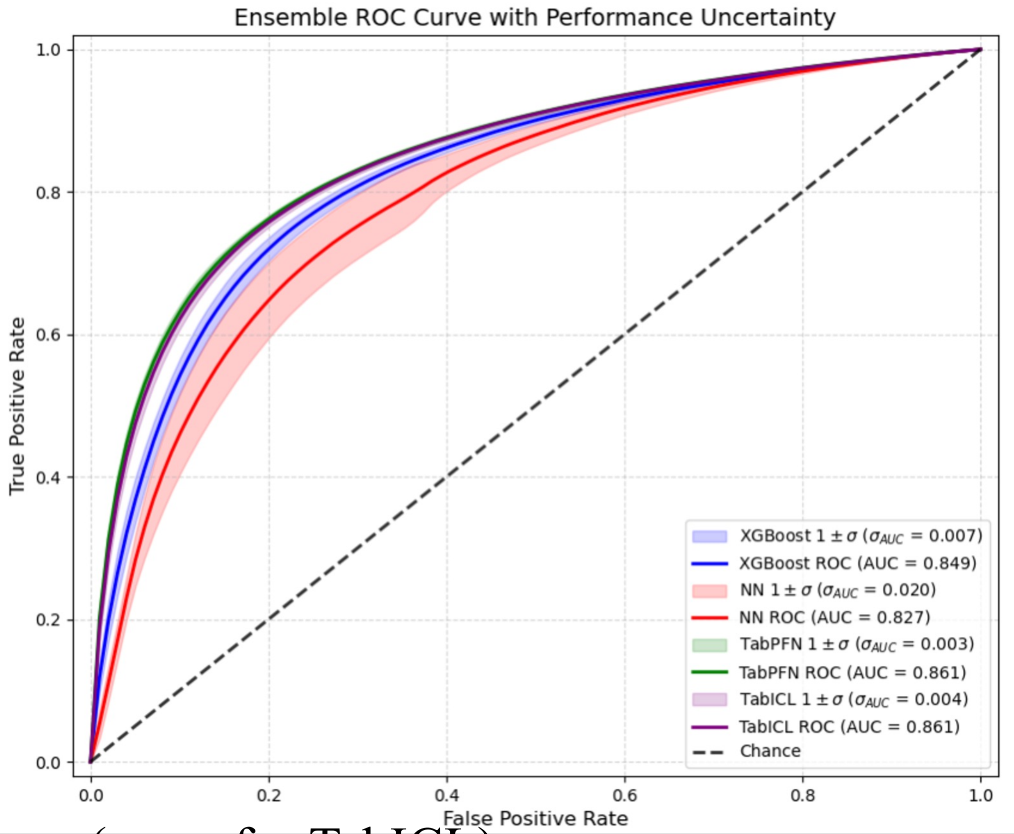
```
import xgboost as xgb
clf_xgb = xgb.XGBClassifier()
clf_xgb.fit(X_train, y_train)
y_pred_xgb = clf_xgb.predict_proba(X_test)
```

⚠️ No fit, no training, only provide a « context »

(on top of pytorch)

```
pip install tabpfn
```

```
from tabpfn import TabPFNClassifier
clf_pfn = TabPFNClassifier()
clf_pfn.fit(X_train, y_train)
y_pred_pfn = clf_pfn.predict_proba(X_test)
```

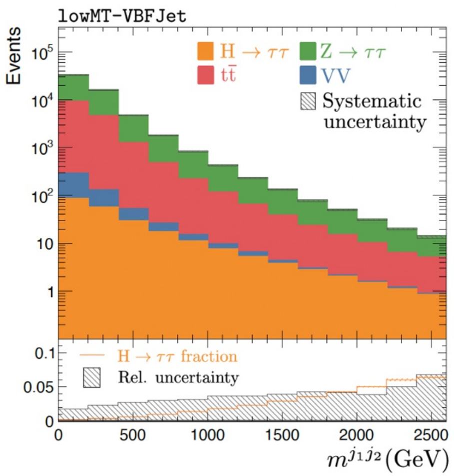


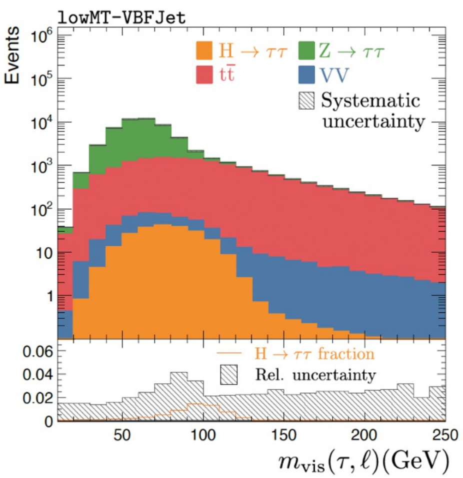
(same for TabICL)

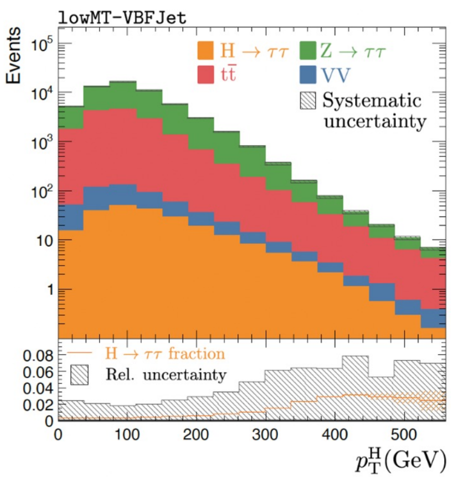
Tabular Foundational Model For HEP

Fair Universe HiggsML Challenge Dataset

- NeurIPS 2024 competition (tie-gold medal by Vienna MBI team !), presented ML4Jets 2025
- $pp \rightarrow \tau^+\tau^-$ Events split into 4 samples (H, Z, ttar and diboson)
- Fast simulation with Pythia LO and Delphes detector simulation
- Parameterised biases (not used here)
- ~**280 Million** Events after initial cuts equivalent to **220** X **10**fb-1 (1 million used here)
- Data Organised into tabular form with **28** features per event.







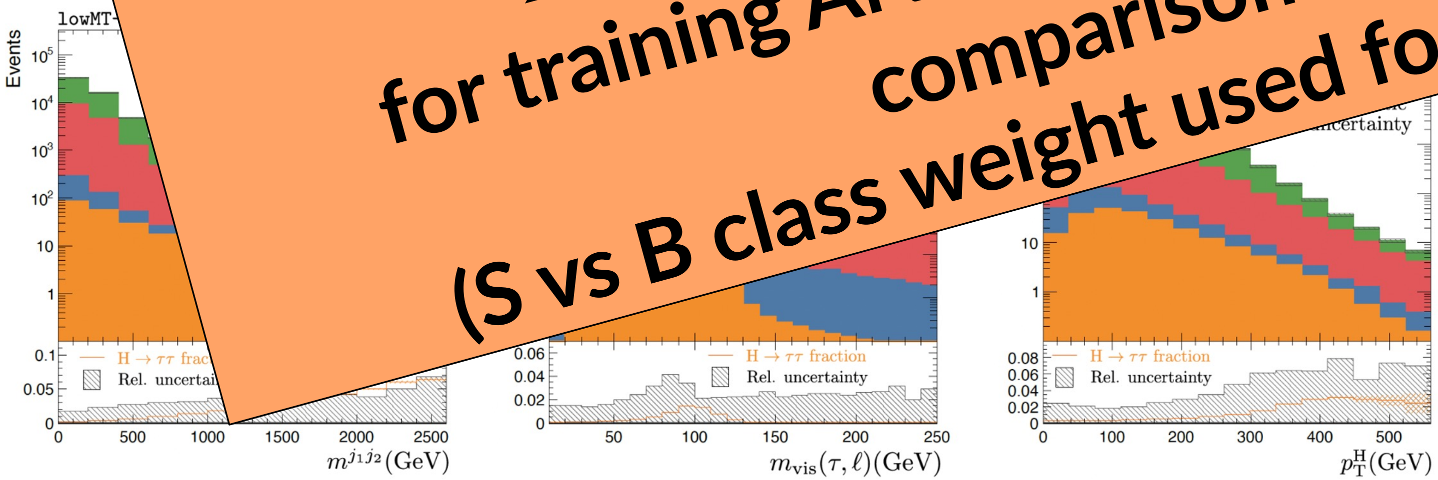
Dataset permanently released on Zenodo https://zenodo.org/records/15131565

Fair Universe HiggsML Challenge D aset

- NeurIPS 2024 competition (tie-gold medal by Vi
- pp→$\tau^+\tau^-$ Events split into 4 sampl
- Fast simulation wit
- Para
-
-

⚠️ CAUTION

TFM models can't handle event weights
➔Event weights are removed for training AND testing to have a fair comparison
(S vs B class weight used for testing)



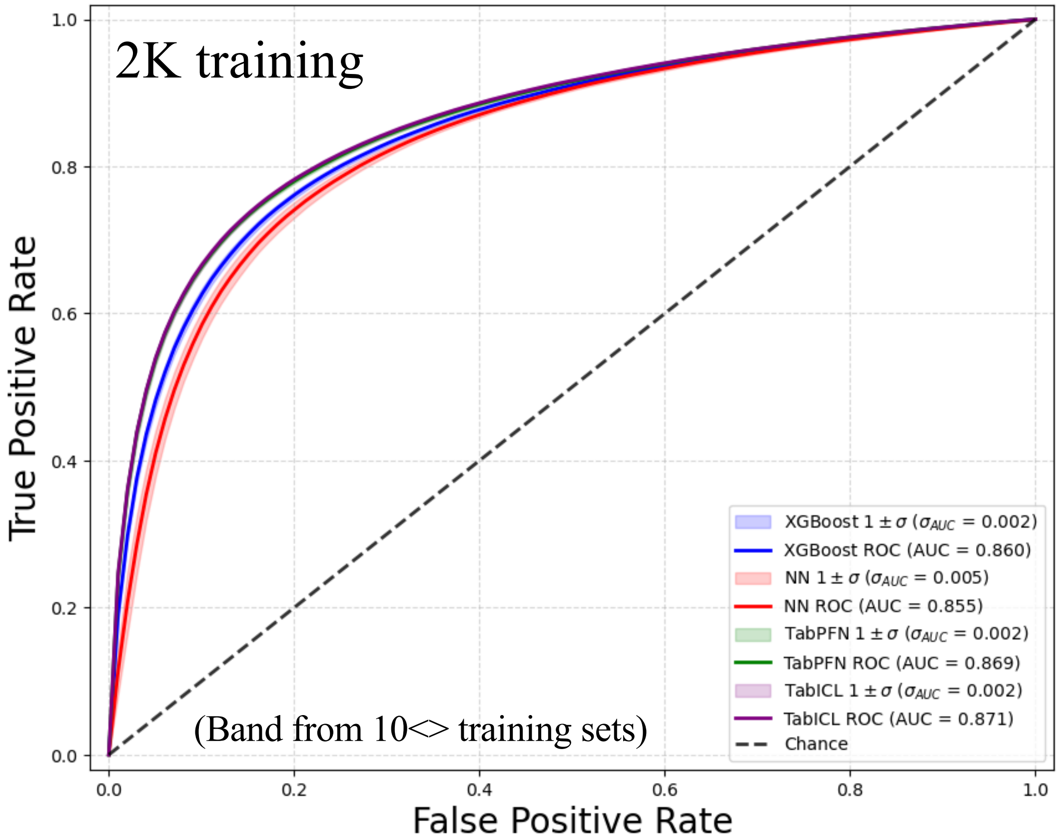
Dataset permanently released on Zenodo https://zenodo.org/records/15131565

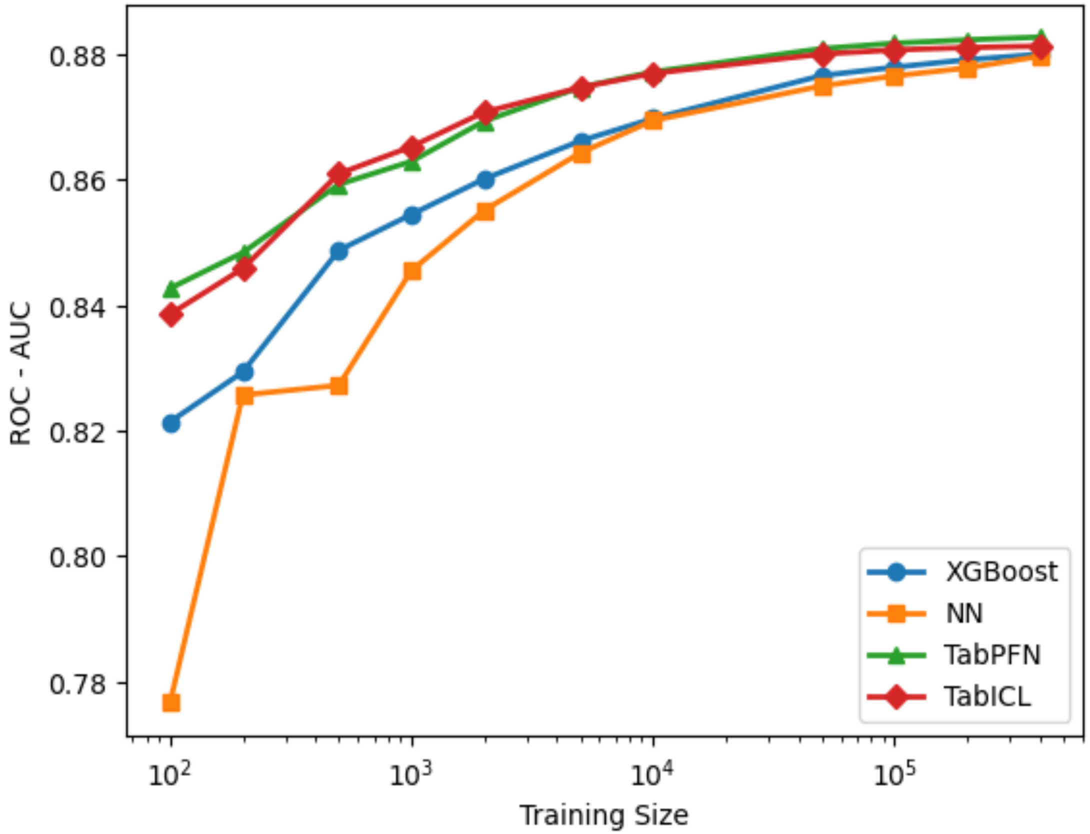
Objective

Thorough comparison of TabPFN and TabICL with XGBoost and dense NN without HPO, focussing on small training statistics (with nS=nB)

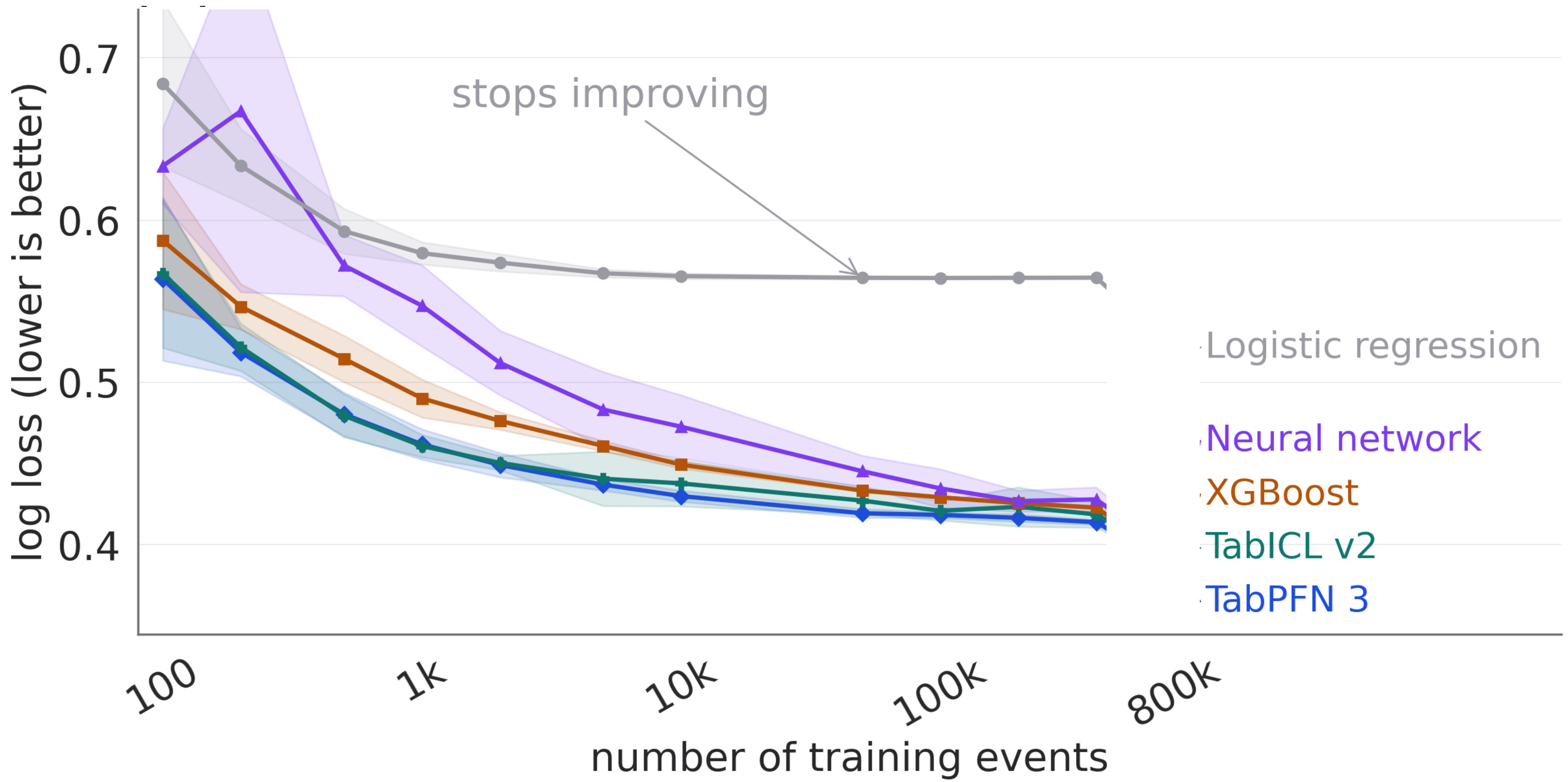
Testing done on fixed 200k test dataset

AUC


2K training
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
XGBoost 1 ± σ (σ_{AUC} = 0.002)
XGBoost ROC (AUC = 0.860)
NN 1 ± σ (σ_{AUC} = 0.005)
NN ROC (AUC = 0.855)
TabPFN 1 ± σ (σ_{AUC} = 0.002)
TabPFN ROC (AUC = 0.869)
TabICL 1 ± σ (σ_{AUC} = 0.002)
TabICL ROC (AUC = 0.871)
Chance
(Band from 10<> training sets)


ROC - AUC
0.78
0.80
0.82
0.84
0.86
0.88
Training Size
10^2
10^3
10^4
10^5
XGBoost
NN
TabPFN
TabICL

Cross-entropy



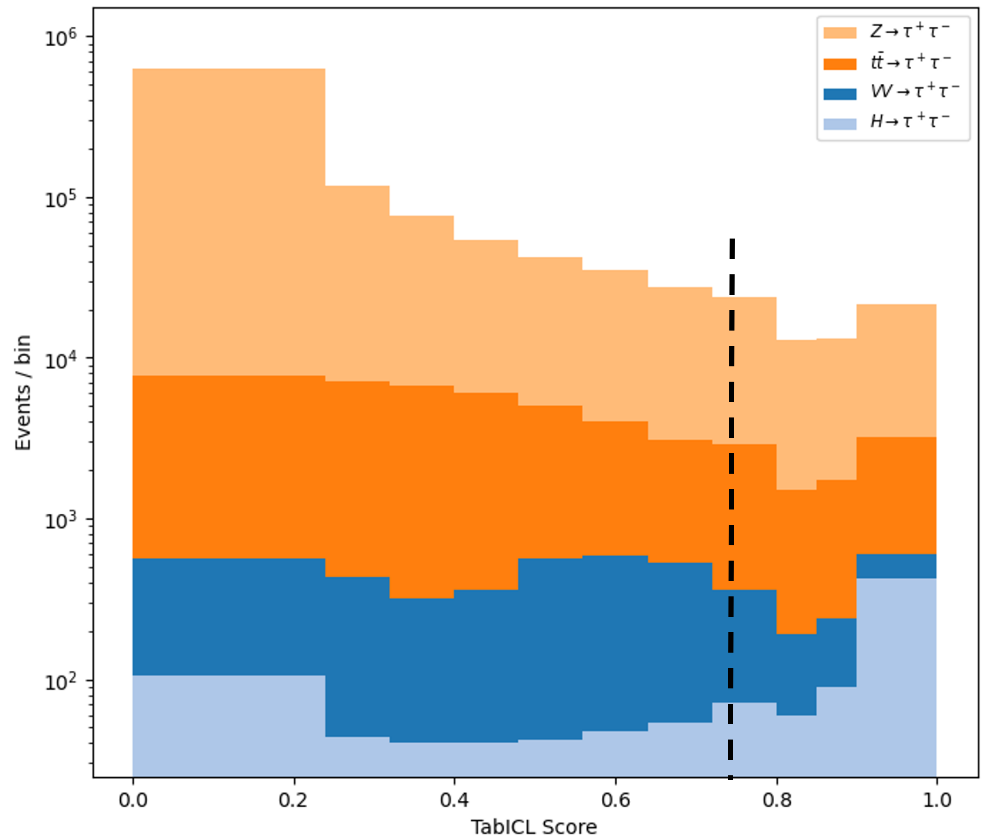
Discovery Significance Scan and Zmax

Simple cut and count over threshold

Asimov Discovery Significance is the statistical power to which we can confirm a discovery

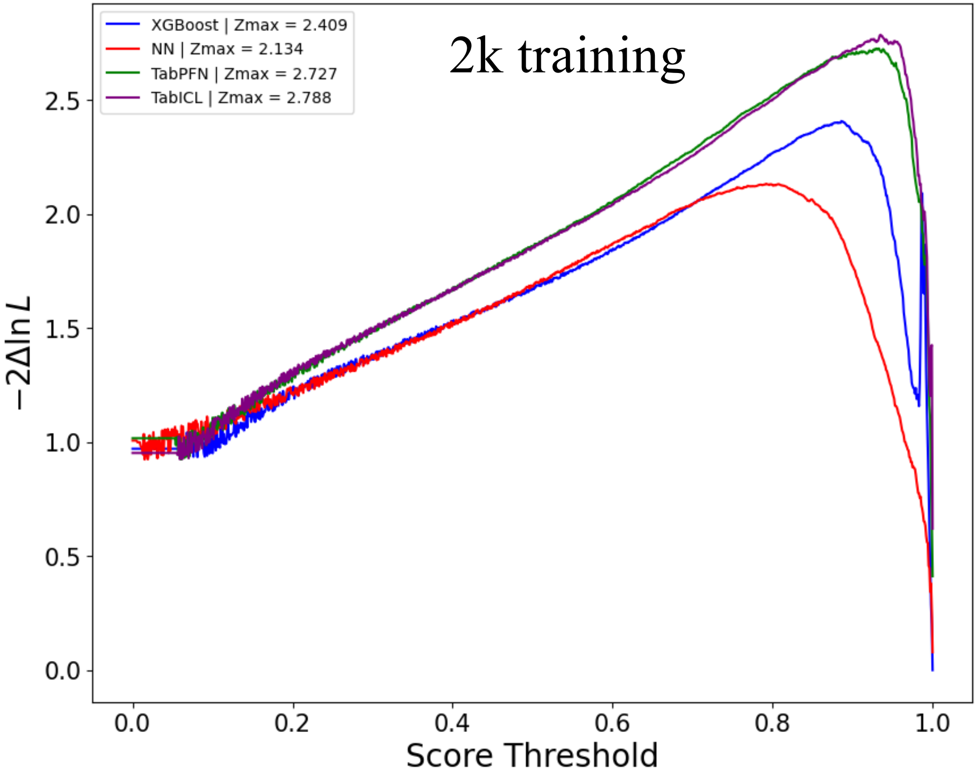
$$Z_A = \sqrt{2\left[(s+b)\ln\left(1+\frac{s}{b}\right)-s\right]}$$

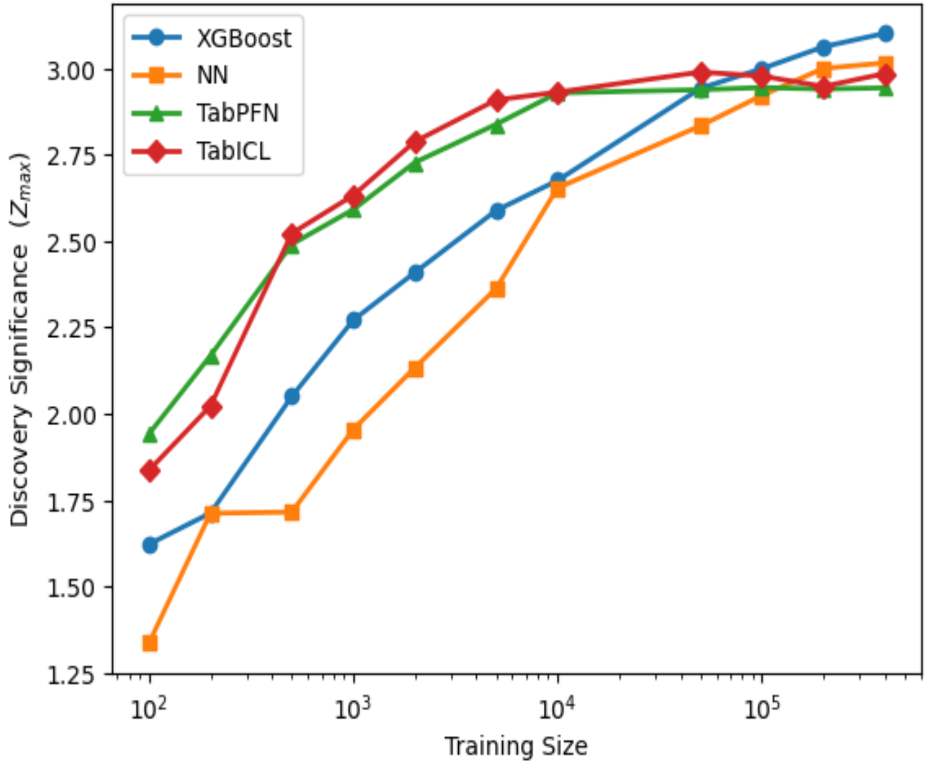
where s and b are the expected number of signal and background.



Discovery Significance Scan and Zmax

Discovery Significance is the statistical power to which we can confirm a discovery

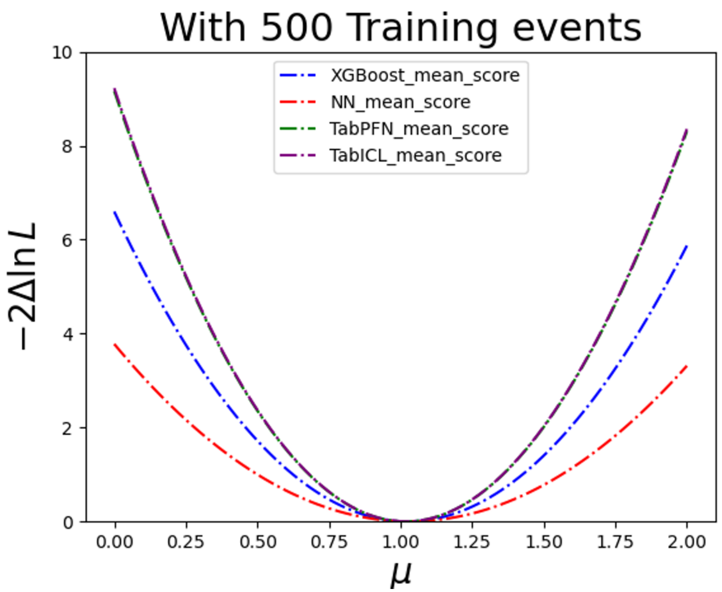


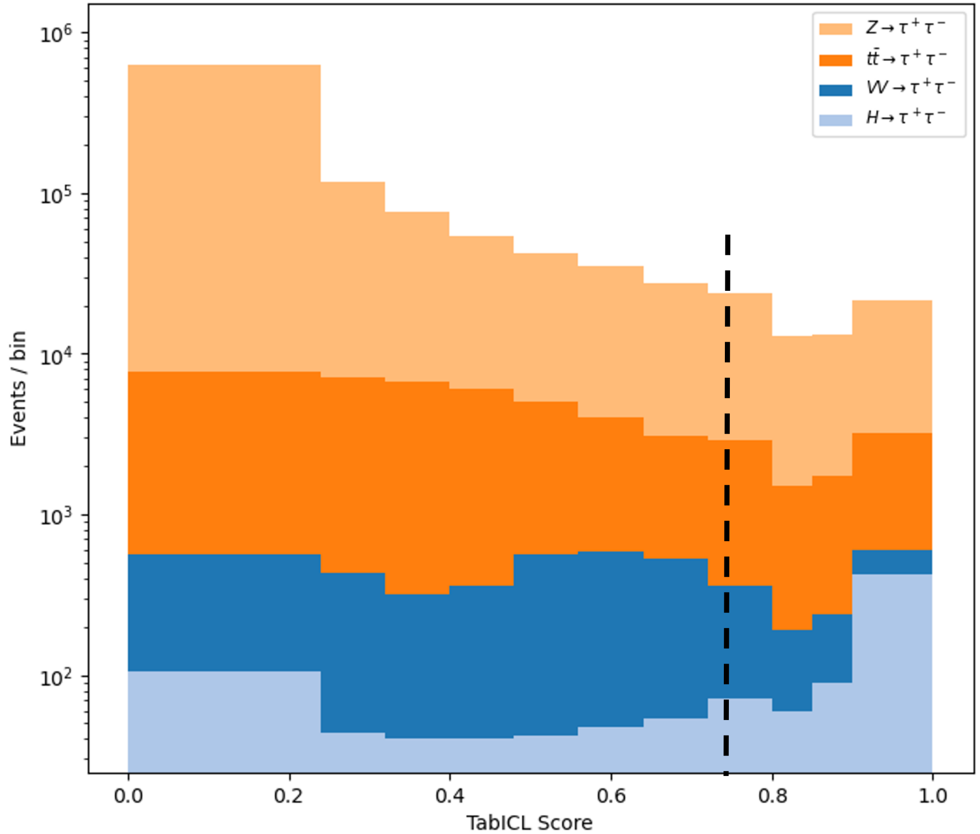


Z - Significance with Histogram.

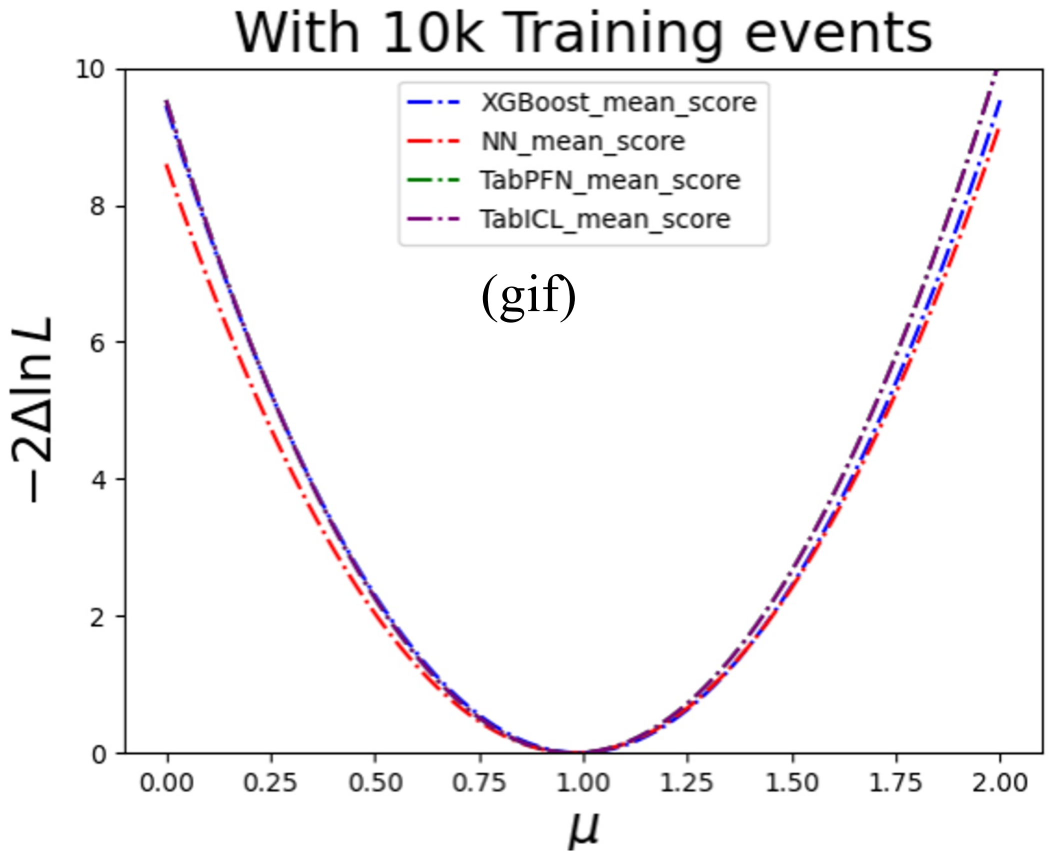
Discovery Significance is the statistical power to which we can confirm a discovery

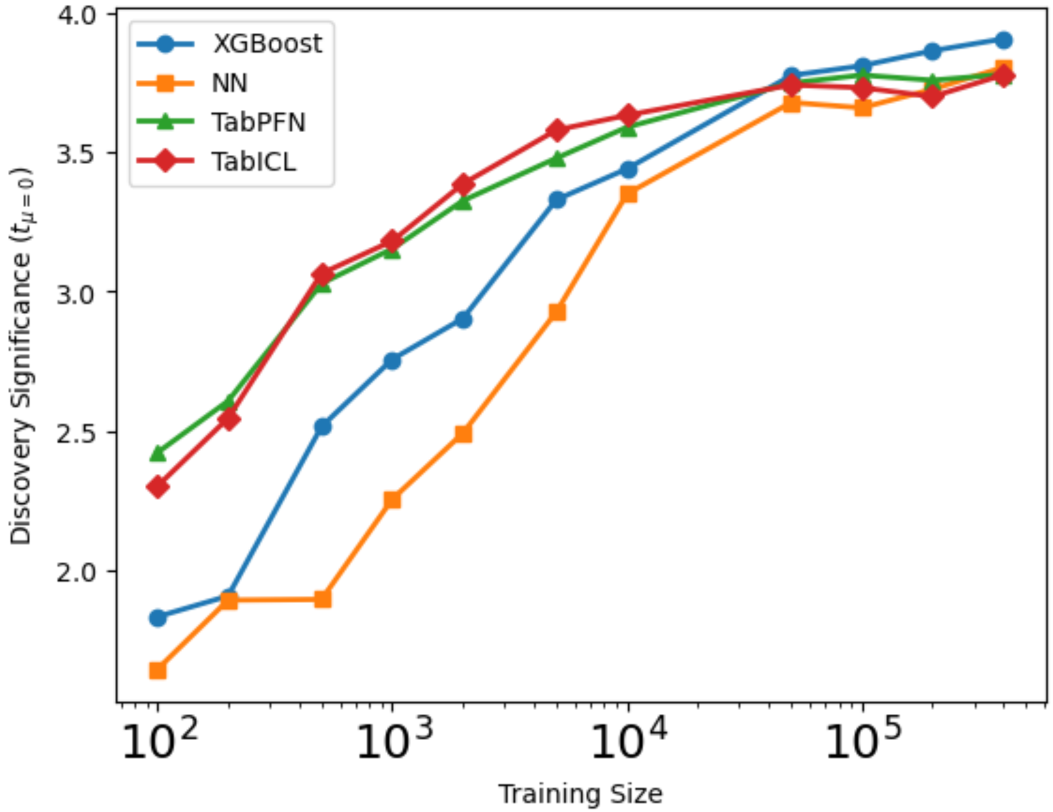
- Compute it in each bin for better statistical power
- Use Negative log likelihood to get Z





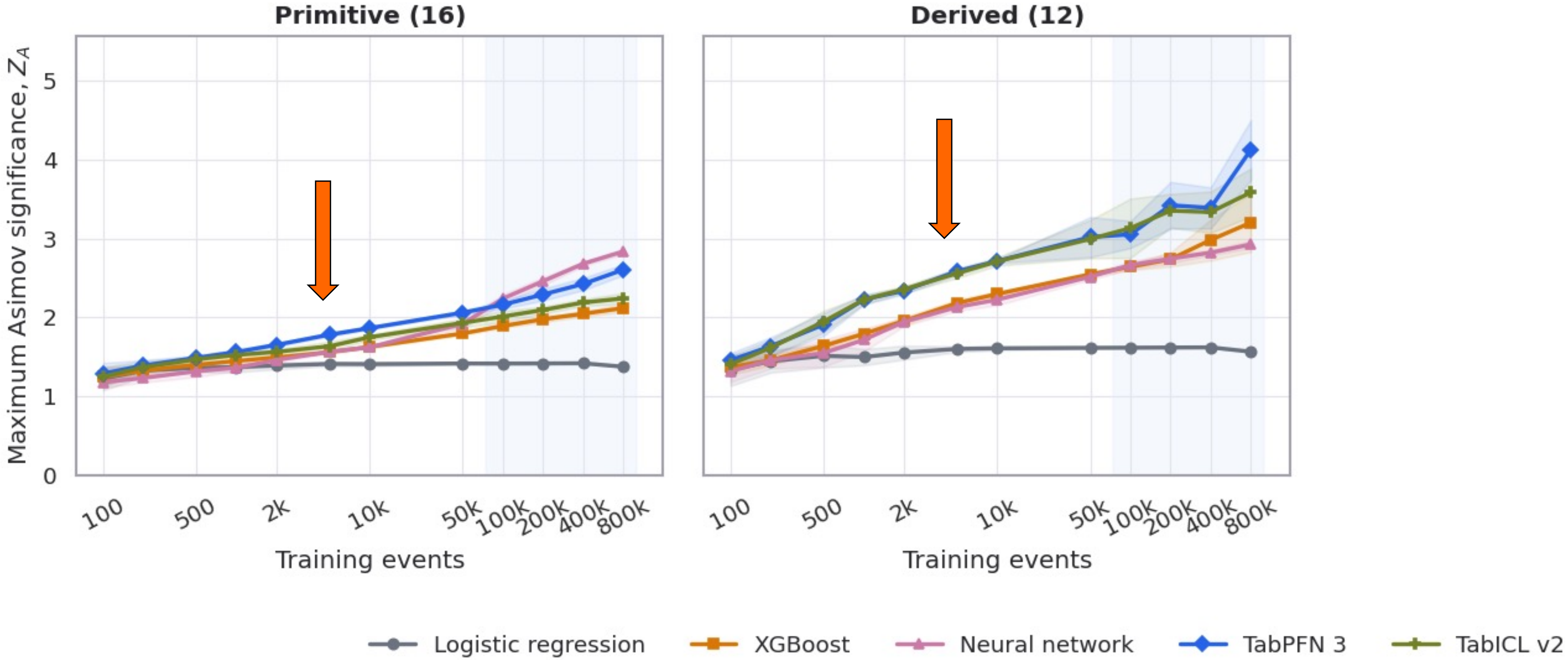
Z - Significance with Binned Histogram





Which features

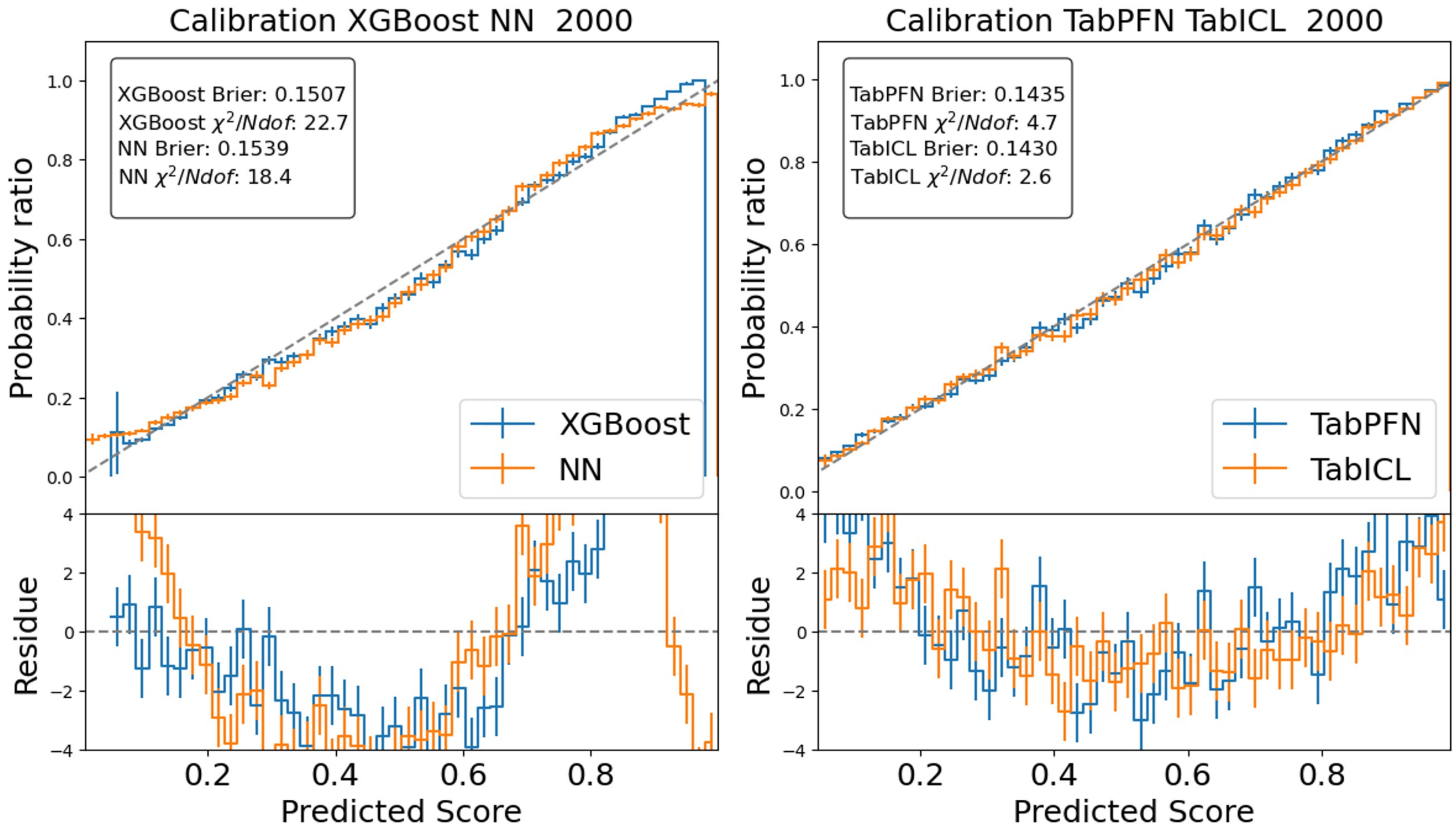
- Features: PRImitive (~4 vectors) vs DERived (effective mass etc…)
- ➔TabPFN/TabICL appear to learn better from DERived features



Calibration curve for different Models

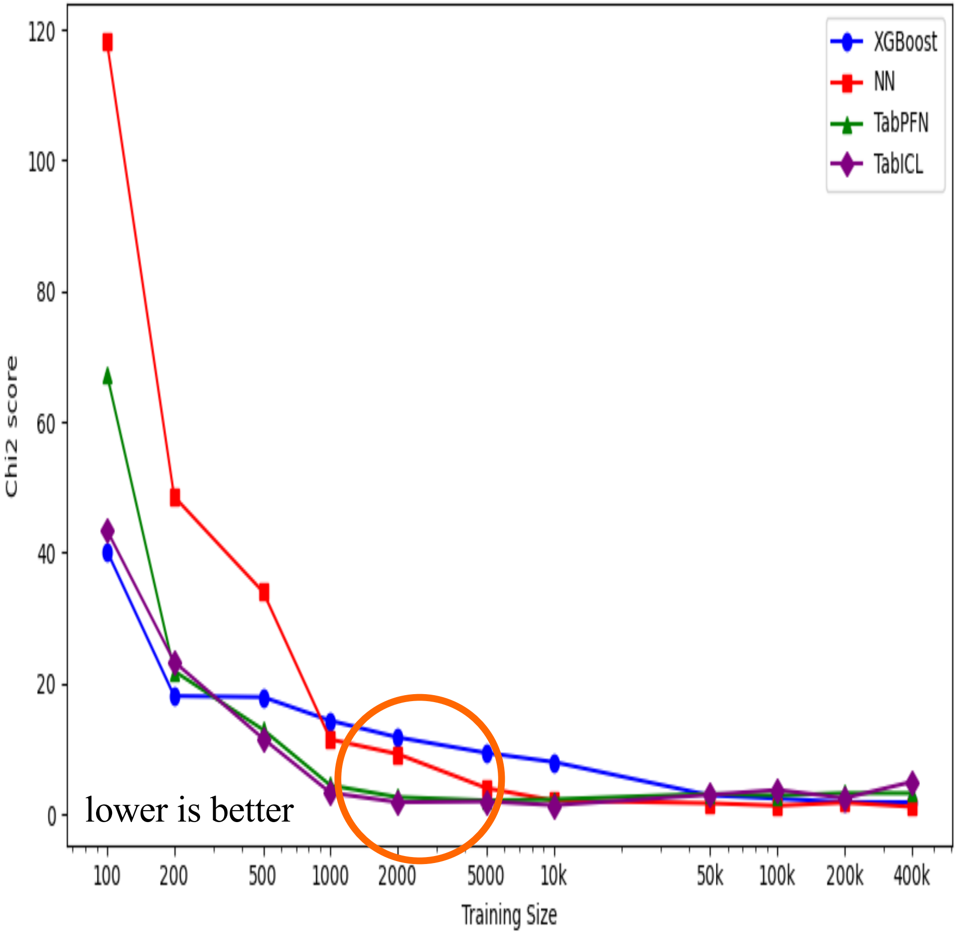
See Jay's talk on SBI

TabICL/TabPFN appear to be « naturally » well calibrated

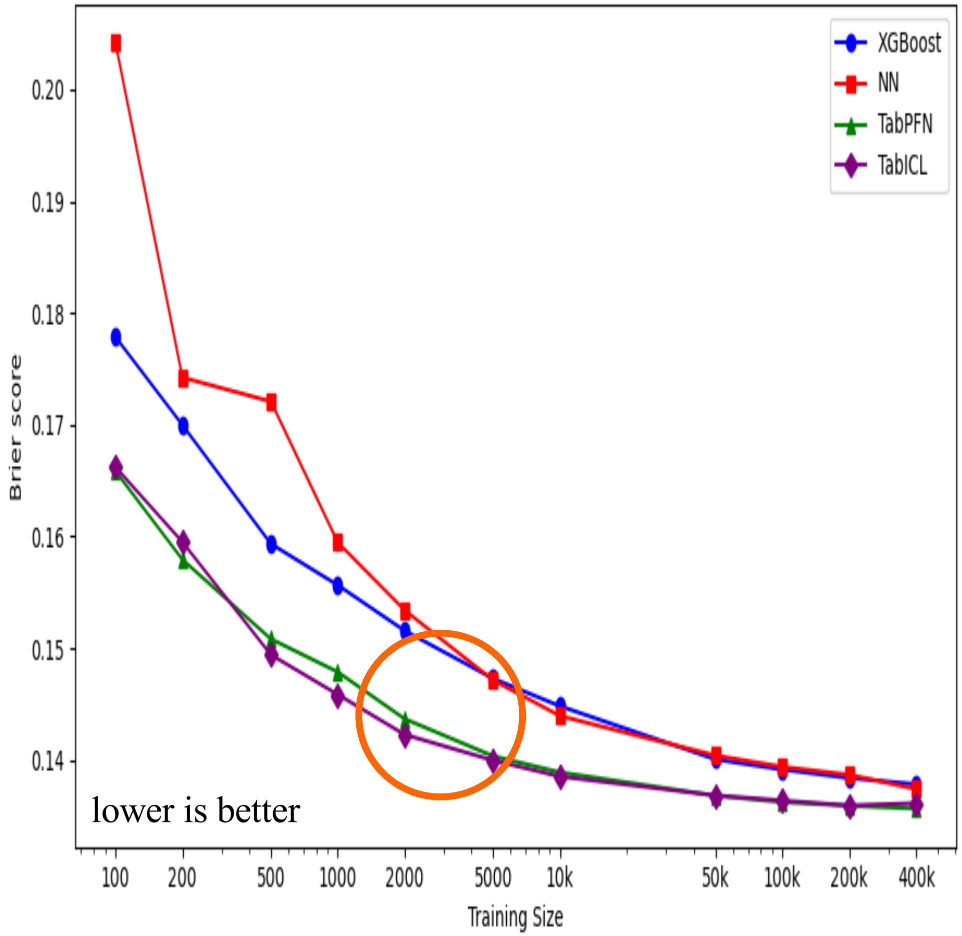


Calibration curve for different Models

Chi2 w.r.t diagonal



Brier score (combines accuracy and calibration)



“Training” and Inference Time

TabPFN/TabICL “Training” Time << XGBoost

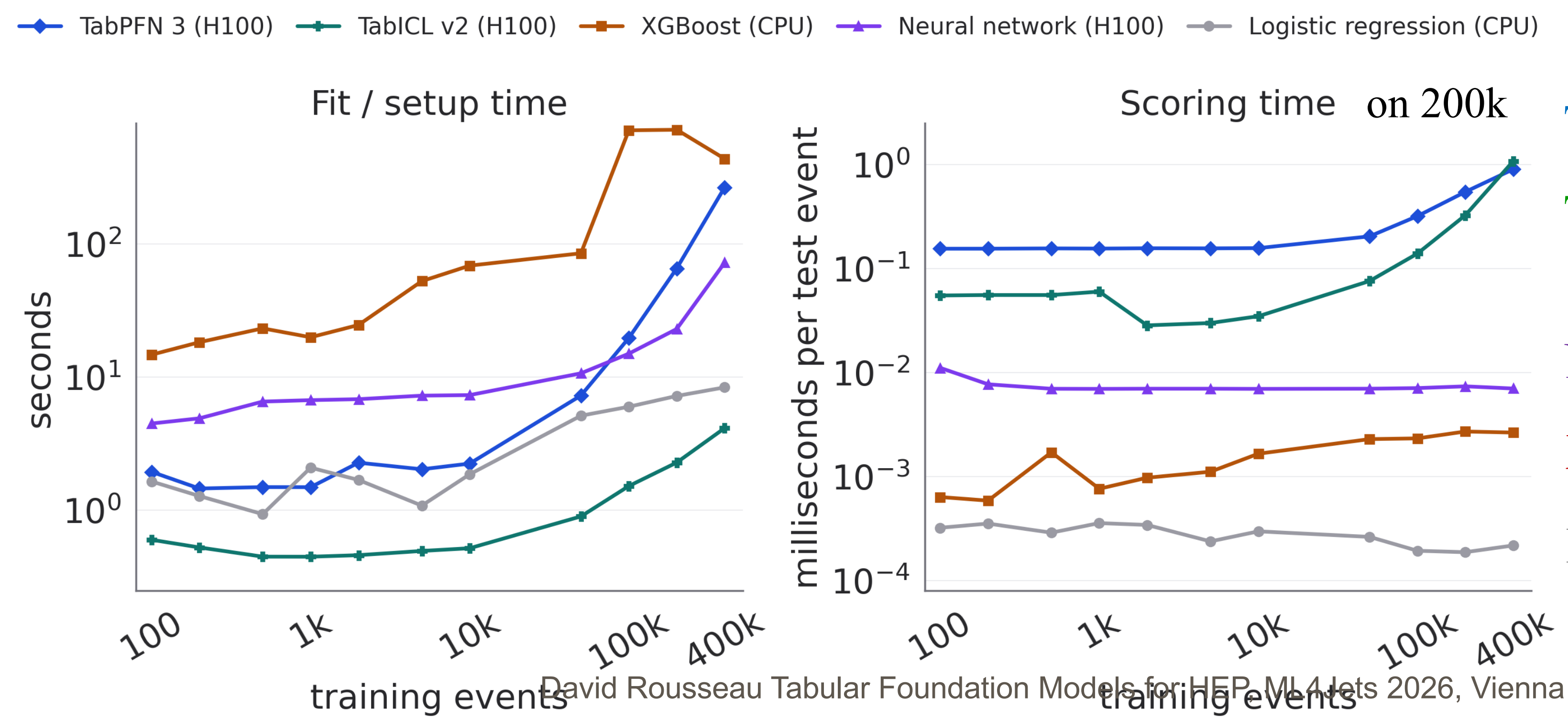
(not training really, but setting the context)

Inference Time Order of Magnitude slower

TabICL Order 3-5 Faster than TabPFN

Runtime up to 400k

Memory is an issue if $N_{training}$ large



TabPFN 3 H100

TabICL v2 H100

NN H100

XGBoost CPU

Logistic CPU

Conclusion

- Tabular Foundation Models show strong potential for use in HEP Analysis
- Significantly (~15%) better than XGBoost and probably all, non foundation models, if low training statistics <~4k (common real life experience)
- Run out of the box, no HPO (remember SAP)
- Better calibration curve (Simulation Based Inference!)
- Inference much slower
- Current how-stopper (analysis dependent): cannot handle event weights (at least so far, stay tuned)