SLING

Fast classifier-based GoF test for online data quality monitoring

Marco Letizia – MaLGa, University of Genoa and INFN

In collaboration with: G. Grosso (UniPd), N. Lai (UniPd), G. Losapio (MaLGa), J. Pazzini (UniPd), M. Pierini (CERN), M. Rando (MaLGa), L. Rosasco (MaLGa), A. Wulzer (IFAE), M. Zanetti (UniPd).

Based on: G. Grosso et al (2023), arXiv:2303.05413.

ML et al (2022), arXiv:2204.02317.

Outline

- Motivations
- GoF via Neyman-Pearson testing
- NPLM - Kernel methods and Falkon
- Applications:
  - HEP data
  - Data Quality Monitoring

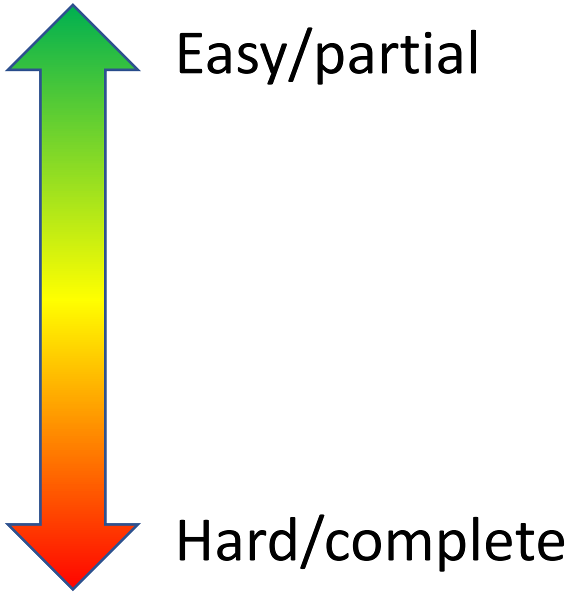
How good is the Standard Model

Analyse (LHC) data $\mathcal{D}$ to find departures from a reference model R (SM).

“Does R describe $\mathcal{D}$ correctly?”

Easier if we check specific alternatives to R (BSM).

More partial as well.



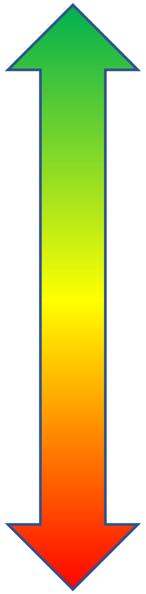
How good is the Standard Model

Analyse (LHC) data $\mathcal{D}$ to find departures from a reference model R (SM).

MODEL DEPENDENT strategy:

BSM models are tested individually against the SM.

However, observables that are sensitive to one BSM model are in general not sensitive to other BSM models.



How good is the Standard Model

Hard to find failures if correct BSM model has not been formulated.

→ Design a MODEL INDEPENDENT strategy.

Detect generic departures from the SM.

How good is the Standard Model

Hard to find failures if correct BSM model has not been formulated.

→ Design a MODEL INDEPENDENT strategy.

Detect generic departures from the SM.

Two flavours:

- Typical model independence: weaken hypothesis on alternative scenarios, e.g. simplified BSM models, bump hunts, effective field theories.
- Machine learning model independence.

GoF via Neyman-Pearson testing

Data: $\mathcal{D} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{D}}$, with $x_i \sim p_{\text{true}}(x)$

Reference hypothesis *R*: $n(x|R) = N(R)\, p(x|R)$ $\quad N = \int p(x)dx$

Goodness of fit: test the consistency between $\mathcal{D}$ and R.

GoF via Neyman-Pearson testing

Data: $\mathcal{D} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{D}}$, with $x_i \sim p_{\text{true}}(x)$

Reference hypothesis *R*: $n(x|R) = N(R)\, p(x|R)$ $\quad N = \int p(x)dx$

Goodness of fit: test the consistency between $\mathcal{D}$ and R.

Neyman-Pearson HT

→ alternative hypothesis: $n(x|A) = N(A)\, p(x|A)$

GoF via Neyman-Pearson testing

Likelihood ratio: $t(\mathcal{D}) = -2\log\frac{\mathcal{L}(\mathcal{D},R)}{\mathcal{L}(\mathcal{D},A)}$, $\mathcal{L}(\mathcal{D}) = \frac{e^{-N}}{\mathcal{N}_{\mathcal{D}}}\prod_{i=1}^{\mathcal{N}_{\mathcal{D}}} n(x)$

GoF via Neyman-Pearson testing

Likelihood ratio: $t(\mathcal{D}) = -2\log\frac{\mathcal{L}(\mathcal{D},R)}{\mathcal{L}(\mathcal{D},A)}$, $\mathcal{L}(\mathcal{D}) = \frac{e^{-N}}{\mathcal{N}_{\mathcal{D}}}\prod_{i=1}^{\mathcal{N}_{\mathcal{D}}} n(x)$

(model dependent)

GoF via Neyman-Pearson testing

Likelihood ratio: $t(\mathcal{D}) = -2\log\frac{\mathcal{L}(\mathcal{D},R)}{\mathcal{L}(\mathcal{D},A)}$, $\mathcal{L}(\mathcal{D}) = \frac{e^{-N}}{\mathcal{N}_\mathcal{D}}\prod_{i=1}^{\mathcal{N}_\mathcal{D}} n(x)$

(model dependent)

Parametrized alternative hypothesis: $n(x|A) \rightarrow n(x|w),$ fix $\widehat{w}$ from data

$$t_{\widehat{w}}(\mathcal{D}) = -2\log\frac{\mathcal{L}(\mathcal{D},R)}{\mathcal{L}(\mathcal{D},\widehat{w})} = 2\left[N(R) - N(\widehat{w}) + \sum_{x=1}^{\mathcal{N}_\mathcal{D}} f_{\widehat{w}}(x)\right], \qquad f_{\widehat{w}}(x) = \log\frac{n(x|\widehat{w})}{n(x|R)}$$

UniGe | MaLGa

GoF via Neyman-Pearson testing

Likelihood ratio: $$t(\mathcal{D}) = -2\log\frac{\mathcal{L}(\mathcal{D},R)}{\mathcal{L}(\mathcal{D},A)},$$ (model dependent)

$$\mathcal{L}(\mathcal{D}) = \frac{e^{-N}}{\mathcal{N}_{\mathcal{D}}}\prod_{i=1}^{\mathcal{N}_{\mathcal{D}}} n(x)$$

Parametrized alternative hypothesis: $$n(x|A) \to n(x|w),$$ fix $\widehat{w}$ from data

$$t_{\widehat{w}}(\mathcal{D}) = -2\log\frac{\mathcal{L}(\mathcal{D},R)}{\mathcal{L}(\mathcal{D},\widehat{w})} = 2\left[N(R) - N(\widehat{w}) + \sum_{x=1}^{\mathcal{N}_{\mathcal{D}}} f_{\widehat{w}}(x)\right],$$

$$f_{\widehat{w}}(x) = \log\frac{n(x|\widehat{w})}{n(x|R)}$$

$$\approx \log\frac{n(x|\text{true})}{n(x|R)}$$

(alternative not related to any particular physics model)

GoF via Neyman-Pearson testing

Data: $\mathcal{D} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{D}}$, with $x_i \sim p_{\text{true}}(x)$

Reference hypothesis *R*: $n(x|R) = N(R)\, p(x|R)$ $\quad N = \int p(x)dx$

$n(x|R)$ not known in closed form → $\mathcal{R} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{R}}$, with $x_i \sim p\,(x|R)$

Two-sample test: assess if $\mathcal{D}$ and $\mathcal{R}$ are sampled from the same distribution.

GoF via Neyman-Pearson testing

Data: $\mathcal{D} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{D}}$, with $x_i \sim p_{\text{true}}(x)$

Reference data: $\mathcal{R} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{R}}$, with $x_i \sim p\,(x|R)$

Goodness of fit via two-sample testing:

$\mathcal{N}_\mathcal{R} \gg \mathcal{N}_\mathcal{D}$: statistical fluctuations in the reference sample are subdominant

→ outcome of two-sample test nearly independent of the specific $\mathcal{R}$ and determined by the agreement of data $\mathcal{D}$ with model *R.*

Summary

Data: $\mathcal{D} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{D}}$, with $x_i \sim p_{\text{true}}(x)$

Reference data: $\mathcal{R} = \{x_i\}_{i=1}^{\mathcal{N}_\mathcal{R}}$, with $x_i \sim p\,(x|R)$

Test statistics:

$$t_w(\mathcal{D}) = 2\left[N(R) - N(w) + \sum_{x\in\mathcal{D}} f_w(x)\right],$$

$$f_w(x) = \log\frac{n(x|w)}{n(x|R)}$$

Summary

How can machine learning help?

- Rich family of alternative hypotheses $n(x|w)$

- Multivariate approach

NPLM

Two approaches:

- Maximum likelihood by minimum loss* (based on neural networks):

$$\max_{w} t_w(\mathcal{D}) = -2 \min_{w} L(f_w, y)$$

$$L(y, f_w) = \sum_{(x,y)} \left[(1-y) \frac{N(R)}{\mathcal{N}_{\mathcal{R}}} \left(e^{f_w(x)} - 1 \right) - y f_w(x) \right]$$

$$t_{\widehat{w}}(\mathcal{D}) = -2\, L(y, f_{\widehat{w}}), \qquad \text{at the end of training}$$

* D'Agnolo et al, arXiv:1806.02350; D'Agnolo et al, arXiv:1912.12155.

NPLM

- Fast kernel-based logistic regression*:

Data: $\{(x_i, y_i)\}_{i=1}^{\mathcal{N}}$, with $\begin{cases} y_i = 0 \text{ if } x_i \in \mathcal{R} \\ y_i = 1 \text{ if } x_i \in \mathcal{D} \end{cases}$

$$L(y, f_w) = \frac{1}{\mathcal{N}} \sum_{(x,y)} (1-y)\log\left(1 + e^{f_w(x)}\right) + y \log\left(1 + e^{-f_w(x)}\right), \qquad f_{\hat{w}} \approx f^* = \log \frac{p(x|1)}{p(x|0)}$$

* ML et al (2022), arXiv:2204.02317.

NPLM

- Fast kernel-based logistic regression*:

Data: $\{(x_i, y_i)\}_{i=1}^{\mathcal{N}}$, with $\begin{cases} y_i = 0 \text{ if } x_i \in \mathcal{R} \\ y_i = 1 \text{ if } x_i \in \mathcal{D} \end{cases}$

$$L(y, f_w) = \sum_{(x,y)} (1-y)\frac{N(R)}{\mathcal{N}_{\mathcal{R}}} \log\left(1 + e^{f_w(x)}\right) + y \log\left(1 + e^{-f_w(x)}\right),$$

$$f_{\hat{w}} \approx f^* = \log \frac{n_{\text{true}}(x)}{n(x|R)}$$

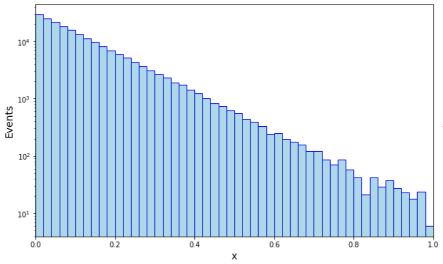
$$\rightarrow t_{\hat{w}}(\mathcal{D}) = 2 \sum_{(x,y)} \left[(1-y)\frac{N(R)}{\mathcal{N}_{\mathcal{R}}}\left(1 - e^{f_{\hat{w}}(x)}\right) + y f_{\hat{w}}(x)\right]$$

* ML et al (2022), arXiv:2204.02317.

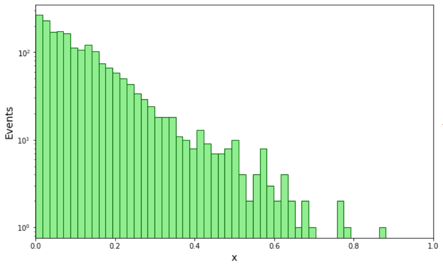
NPLM

INPUT

Reference sample $\mathcal{R}$



Data sample $\mathcal{D}$



Logistic regression classifier

$f_w \to f_{\hat{w}}$

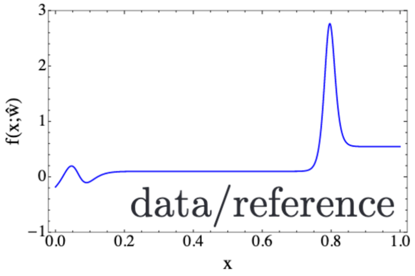
OUTPUT

Likelihood ratio test statistic

$t_{\hat{w}}(\mathcal{D})$

Learned density ratio

$$f_{\hat{w}}(x) \approx \log \frac{n_{\text{true}}(x)}{n(x|R)}$$



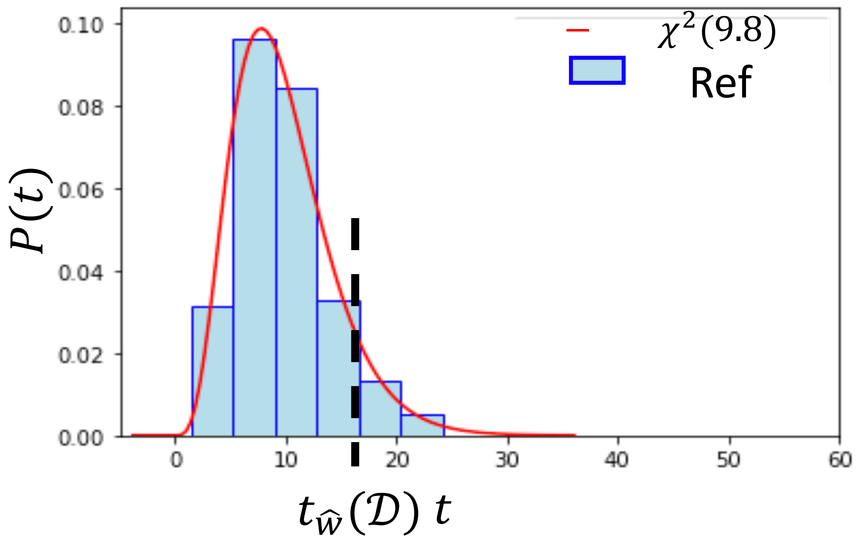
NPLM

Large $t_{\widehat{w}}(\mathcal{D})$ $\rightarrow$ data are anomalous with respect to the reference model.

How large? We need to <u>calibrate</u>:

We use *reference pseudo-experiments* (toys):

Train large reference sample $\mathcal{R}$ against batches of reference data with the same statistics of the actual data but sampled from the reference distribution.



Estimate the distribution of t under the hypothesis of validity of the reference model (null hypothesis) → p-value/Z-score

Kernel methods and Falkon

ERM $\hat{f} = \arg\min_{f \in \mathcal{H}} \hat{L}(f) + \lambda\, R$

Loss function $\ell\big(y, f(x)\big)$

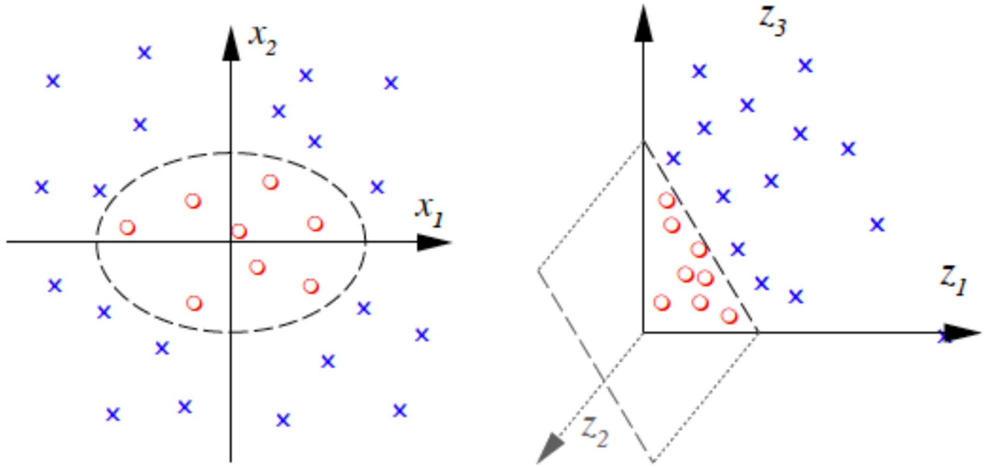
Select a space $\mathcal{H}$ of possible functions, e.g.

- Linear $f(x) = w^T x$
- Neural networks "$\sigma(\sigma\big(\ldots\sigma(w^T x)\big))$"

Kernel methods and Falkon

Kernel methods $f(x) = w^T \phi(x),$ $\phi: X \to F$ feature map

Input (x_1, x_2) $\phi(x) = (x_1^2, x_2^2, \sqrt{2}\, x_1\, x_2)$



Kernel methods and Falkon

One can take infinite dimensional feature maps if $k(x, x') = \phi^T(x)\phi(x')$ can be computed.

The solution to the ERM problem can be written as (*representer theorem*)

$$f_{\hat{w}}(x) = \sum_{i=1}^{n} c_i \, k(x, x_i)$$

Common kernels:

-Linear $k(x, x') = x^T x'$

-Polynomial $k_d(x, x') = (x^T x' + 1)^d$

-Gaussian $k_\sigma(x, x') = \exp - \frac{\|x - x'\|^2}{2\sigma^2}$

Kernel methods and Falkon

Kernel methods are very flexible, they can approximate any continuous function given enough data.*

They do not scale well: kernel matrix $K_{nn} \in \mathbb{R}^{n \times n}$ with entries $k(x_i, x_j)$.

- Computational complexity: $\mathcal{O}(n^2)$ in space and $\mathcal{O}(n^3)$ in time (direct KRR).

→ Some approximation is needed.

* A. Micchelli et al, Universal kernels (2006); A. Christmann and I. Steinwart, Support Vector Machines (2008)

Kernel methods and Falkon

Falkon:*

A modern algorithm to efficiently extend kernel methods to large scale problems, $n = \mathcal{O}(10^7)$.

- Nyström approximation (column subsampling)
- Approximate interative solver
- Efficient (multi-)GPU implementation

* G. Meanti et al, arXiv:2006.10350

Kernel methods and Falkon

- Nyström approximation

$$f(x) = \sum_{i=1}^{n} c_i \, k(x, x_i) \;\rightarrow \sum_{i=1}^{M} c_i \, k(x, x_i)\,,$$

$\{x_1, \dots, x_M\} \subset \{x_1, \dots, x_n\}$ sampled uniformly at random (*centers*).

Theorem (Rudi, Camoriano, R. '15)

Let $(\tilde{x}_i)_{i=1}^M \subseteq (x_i)_{i=1}^n$ ***picked uniformly at random, if*** $\lambda = 1/\sqrt{n}$ ***and*** $M \geqslant \sqrt{n}$ ***then***

$$\mathbb{E}L(\widehat{f}_{\lambda,M}) - \min_{f \in \mathcal{H}} L(f) \lesssim \frac{1}{\sqrt{n}}$$

Kernel methods and Falkon

- Approximate interative solver

Preconditioner: $\mathrm{P}^{\mathrm{T}}\mathrm{P} = (K_{nM} K_{nM} + \lambda n K_{MM})^{-1}$

Nyström → $P^T P \approx \left(\frac{n}{M} K_{MM}^2 + \lambda n K_{MM}\right)^{-1}$

→ Conjugate gradient with approximate preconditioning.

Theorem (Rudi, Carratino, Rosasco '17)

Let $(\tilde{x}_i)_{i=1}^M \subseteq (x_i)_{i=1}^n$ *uniformly at random*, ***then if*** $\lambda = 1/\sqrt{n}$, $M \geqslant \sqrt{n}$ ***and*** $t \geqslant \log(n)$

$$\mathbb{E}L(\widehat{f}_{\lambda,M,t}) - \min_{f \in \mathcal{H}} L(f) \lesssim \frac{1}{\sqrt{n}}$$

Kernel methods and Falkon

Overall space $\mathcal{O}(n^2)$, time $\mathcal{O}(n^3)$ $\rightarrow$ space $\mathcal{O}(n)$, time $\mathcal{O}(n\sqrt{n}\log n)$

Falkon2.0: millions/billions (!) of points in minutes

	TAXI $n \approx 10^9$		HIGGS $n \approx 10^7$		YELP $n \approx 10^6$, $d \approx 10^7$		TIMIT $n \approx 10^6$	
	RMSE	time(h)	AUC	time(m)	RMSE	time(m)	c-err	time(m)
FALKON	**311.7**	**1**	0.8196	**7.4**	**0.810**	**16.8**	32.27%	**4.8**
LogFALKON	–	–	**0.8213**	37.8	–	–	–	–
EigenPro2	FAIL		FAIL		FAIL		**31.91%**	29
GPyTorch	322.5	10.8	0.8005	52.9	FAIL		–	–
GPflow	313.2	8.5	0.8042	24.3	FAIL		33.78%	44.5

	AIRLINE-CLS $n \approx 10^6$		AIRLINE $n \approx 10^6$		MSD $n \approx 10^5$		SUSY $n \approx 10^6$	
	c-err	time(m)	MSE	time(m)	relative error	time(m)	c-err	time(m)
FALKON	31.5%	**3.1**	**0.758**	**4.1**	4.4834×10^{-3}	**1**	19.67%	**0.4**
LogFALKON	**31.3%**	21.5	–	–	–	–	**19.58%**	1.4
EigenPro2	32.5%	27.2	0.785	24.5	4.4778×10^{-3}	6.3	20.08%	1.5
GPyTorch	33.0%	24.2	0.803	31	4.5344×10^{-3}	15.5	19.71%	16.5
GPflow	32.6%	10.5	0.790	28.7	4.4986×10^{-3}	8.8	19.65%	9.3

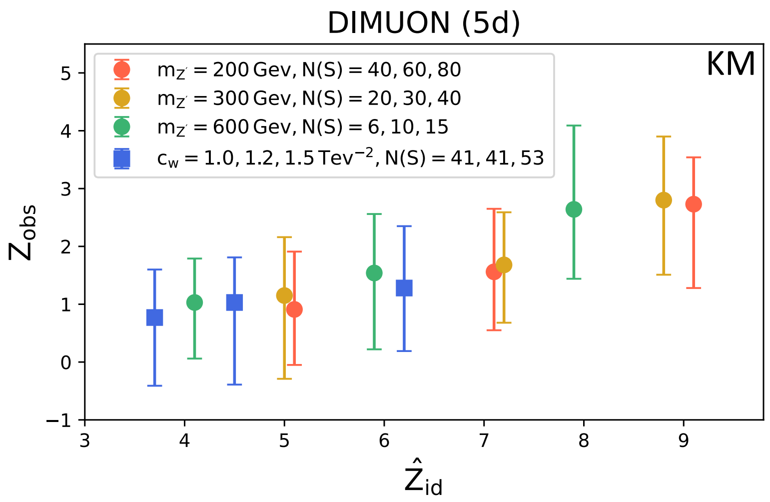
G. Meanti et al, arXiv:2006.10350

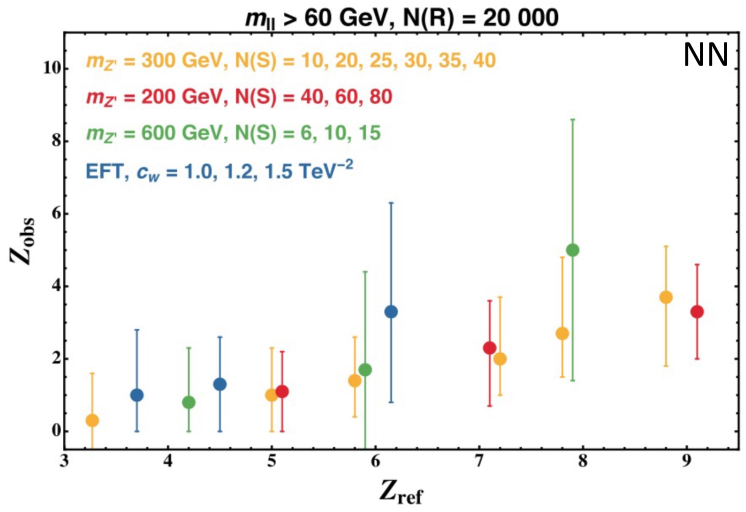
Hyperparameters

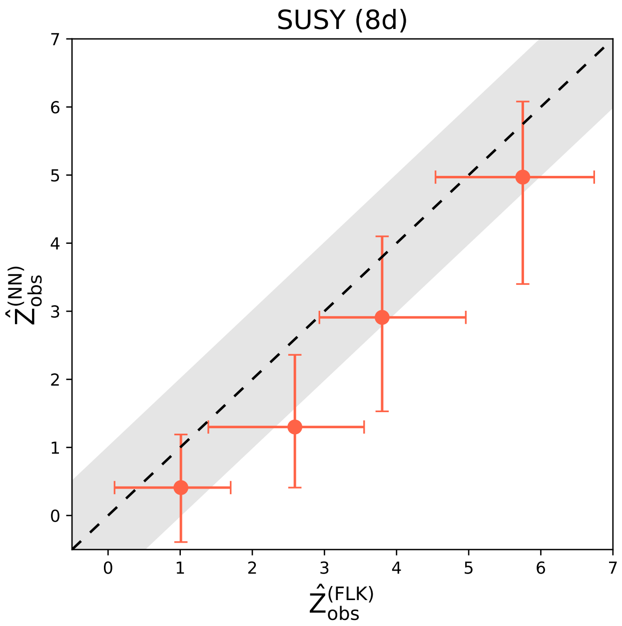
Three main hyperparameters:

- The bandwidth σ: 90th percentile of the pairwise distance between reference-distributed data point.
- The L2 parameter λ: as small as possible (stability and training time).
- The number of centers $M \geq \sqrt{N}$

HEP data







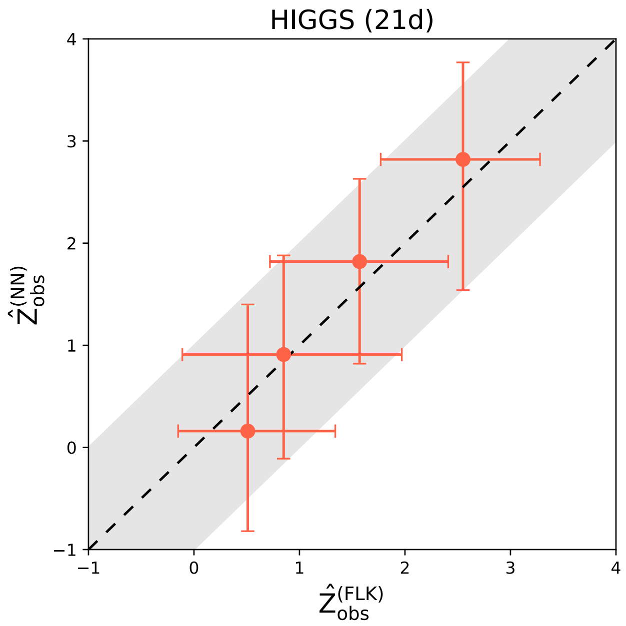


Table 1 Average training times per single run with standard deviations (low level features and reference toys). Note that time measured in hours (for NN) and seconds (for Falkon)

Model	DIMUON	SUSY	HIGGS
FLK	**(44.9 ± 3.4) s**	**(18.2 ±1.2) s**	**(22.7 ± 0.4) s**
NN	(4.23 ± 0.73) h	(73.1 ± 10) h	(112 ± 9) h

Bold values indicate the lowest for each column (lower is better)

DQM

Monitoring data to assess their quality and to promptly detect detector malfunctions.

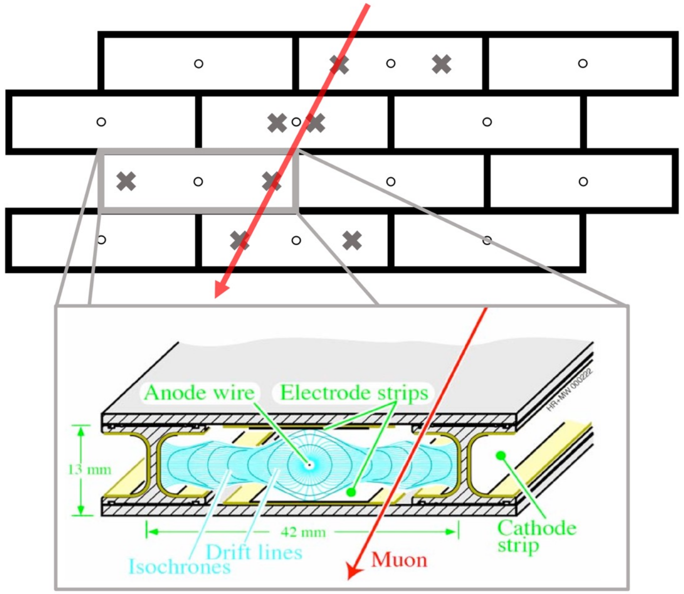
Modern high-energy physics experiments operating at colliders are extremely sophisticated devices consisting of millions of sensors sampled every few nanoseconds, producing an enormous throughput of data.

→ fast, automated, multivariate approach to DQM.

→ NPLM.

G. Grosso et al, arXiv:2303.05413.

DQM



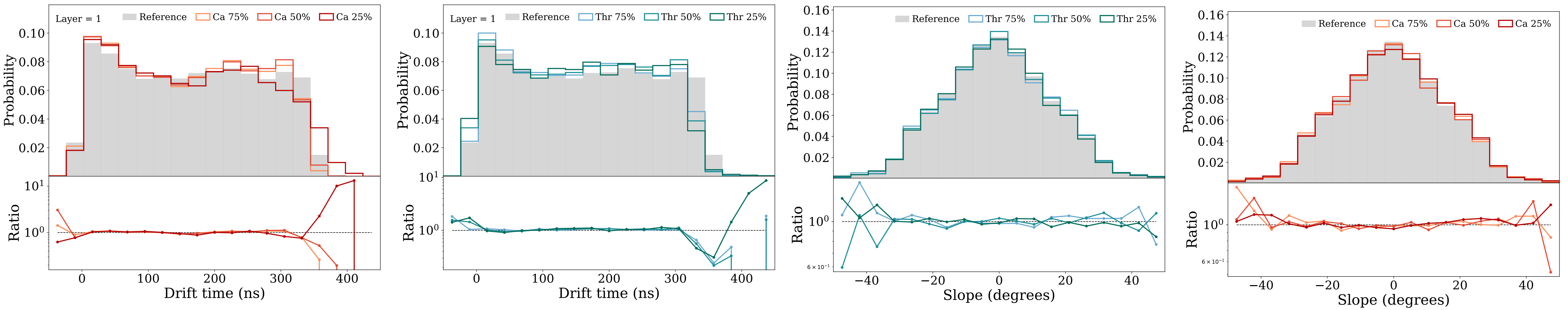
L: 70 cm
cross section: 4×2.1 cm^2
Anode: 3.6 kV
Cathode: −1.2 kV
Gas: Argon Carbon Dioxide (85+15)

DT chambers from Legnaro INFN National Laboratory.
Reduced-size version of the muon chambers installed in the CMS experiment.

A drift tube chamber consists of 64 tubes arranged in four layers of 16 tubes each.
Data acquisition at 40 MHz.

G. Grosso et al, arXiv:2303.05413.

DQM



DATASET:

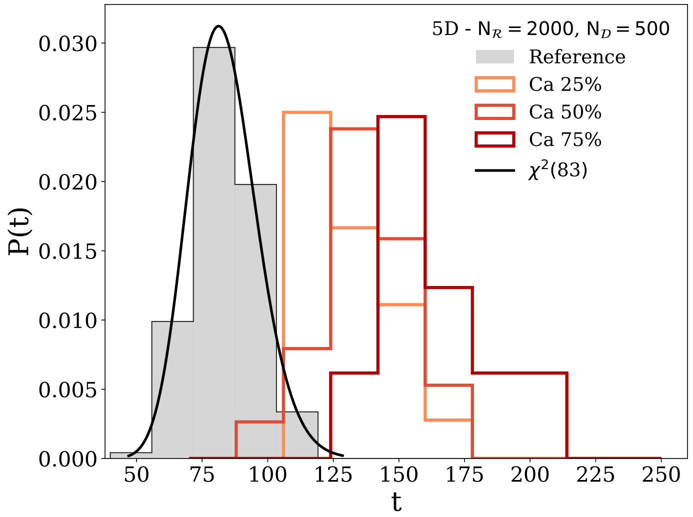
- Drift times (t_i): the four drift times of the muon track.
- Slope (ϕ): the angle with respect to the vertical axis.

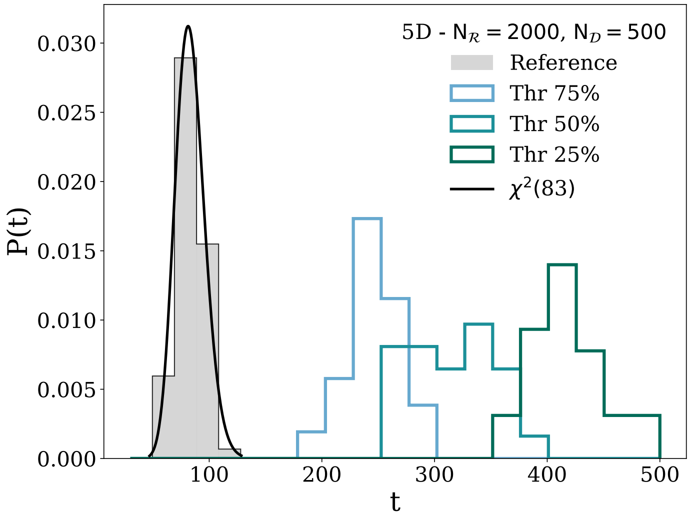
- Reference data is collected in a controlled regime.

- Anomalies:
  - reduced voltage of cathodic strips to 75%, 50%, and 25% of their nominal value (-1.2 kV)
  - lowered front-end thresholds to 75%, 50%, and 25% of nominal value (100 mV)

G. Grosso et al, arXiv:2303.05413.

DQM



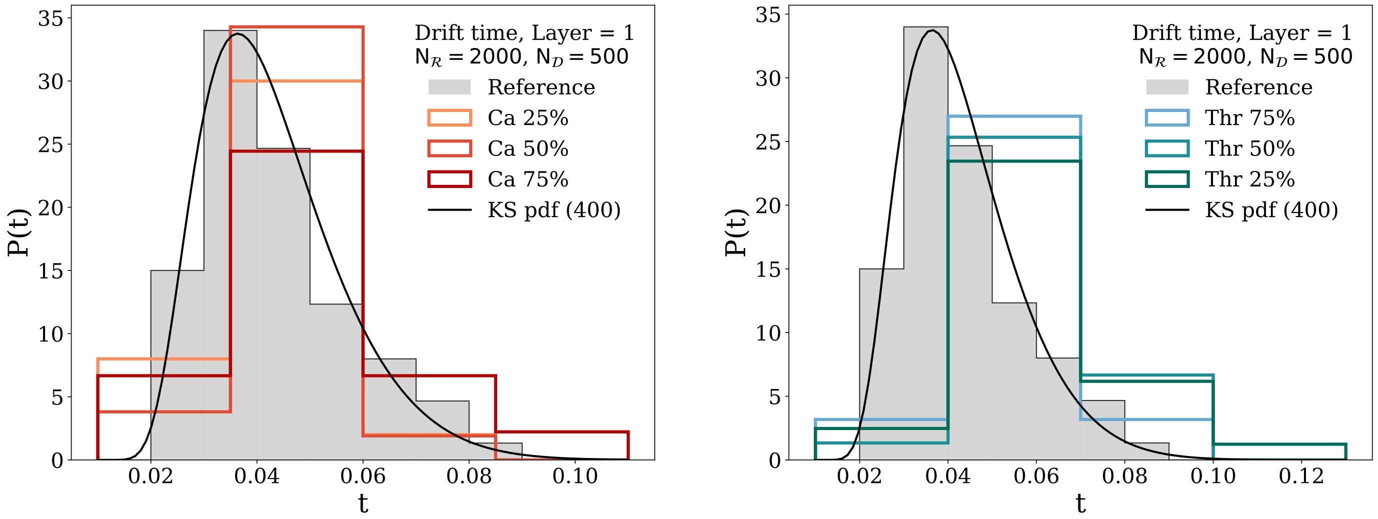


Anomaly	$N_{\mathcal{D}} = 250$	$N_{\mathcal{D}} = 500$	$N_{\mathcal{D}} = 1000$
Cathode 75%	0.0034	1.1×10^{-6}	$< 10^{-7}$
Cathode 50%	0.029	3.4×10^{-4}	$< 10^{-7}$
Cathode 25%	0.14	0.0019	$< 10^{-7}$
Threshold 75%	2.8×10^{-7}	$< 10^{-7}$	$< 10^{-7}$
Threshold 50%	$< 10^{-7}$	$< 10^{-7}$	$< 10^{-7}$
Threshold 25%	$< 10^{-7}$	$< 10^{-7}$	$< 10^{-7}$

GPU: NVIDIA Titan Xp GPU (12 GB VRAM)
$\bar{t}_{train} \approx 0.5$ sec

G. Grosso et al, arXiv:2303.05413.

DQM



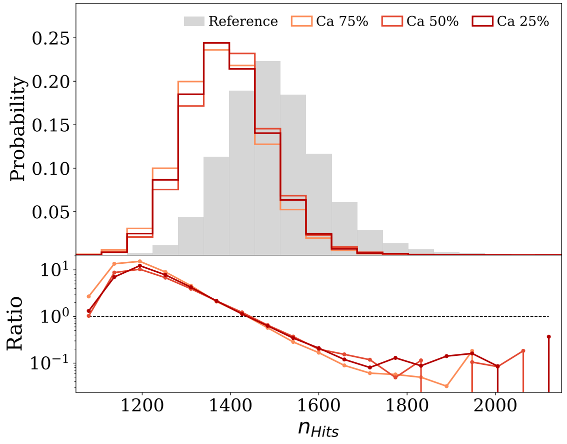
Anomaly	NPLM (5D)	KS (t_1)	KS (t_2)	KS (t_3)	KS (t_4)	KS (ϕ)
Cathode 75%	1.1×10^{-6}	0.50	0.41	0.43	0.40	0.42
Cathode 50%	3.4×10^{-4}	0.47	0.27	0.47	0.37	0.41
Cathode 25%	0.0019	0.45	0.44	0.13	0.45	0.50
Threshold 75%	$< 10^{-7}$	0.23	0.14	0.16	0.14	0.48
Threshold 50%	$< 10^{-7}$	0.09	0.10	0.06	0.17	0.42
Threshold 25%	$< 10^{-7}$	0.11	0.07	0.04	0.11	0.66

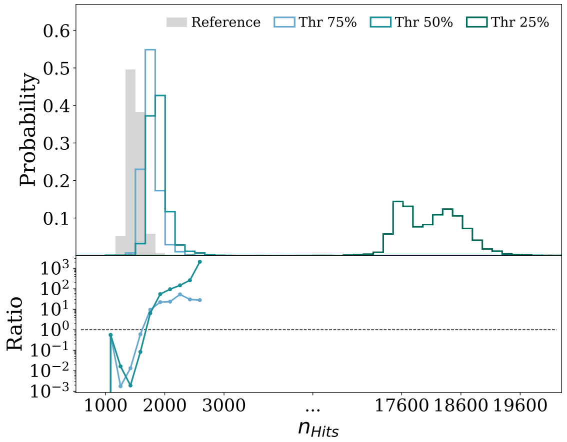
Table 3: Median p-values in the setup $N_{\mathcal{D}} = 500$.

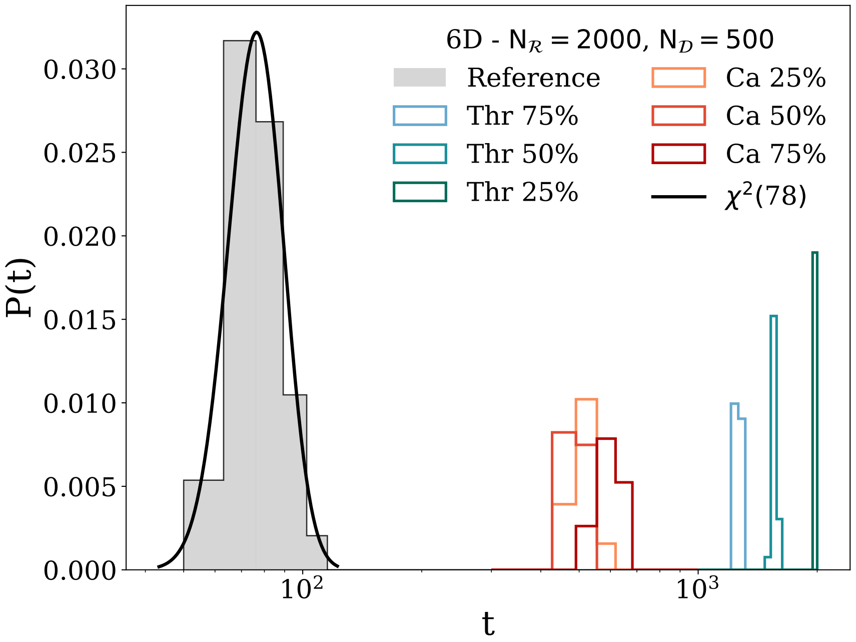
G. Grosso et al, arXiv:2303.05413.

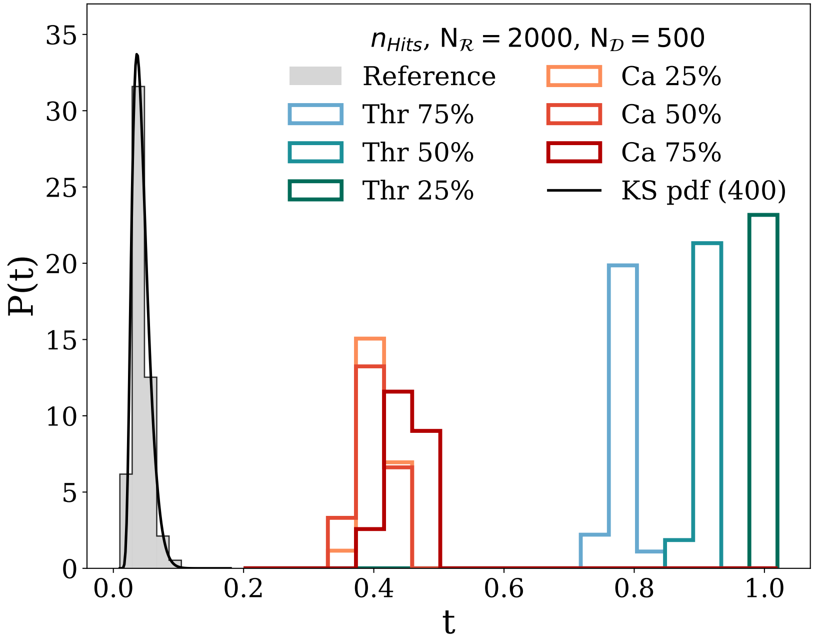
DQM

n_{Hits}: the number of hits in a time window of one second around the muon crossing time.









G. Grosso et al, arXiv:2303.05413.

Outlook

- Mismodeling of the reference distribution – systematic uncertainties
- Characterization of the null distribution
- Falkon algorithm (selection of centers, hyperparameter tuning,…)
- Small reference sample

References

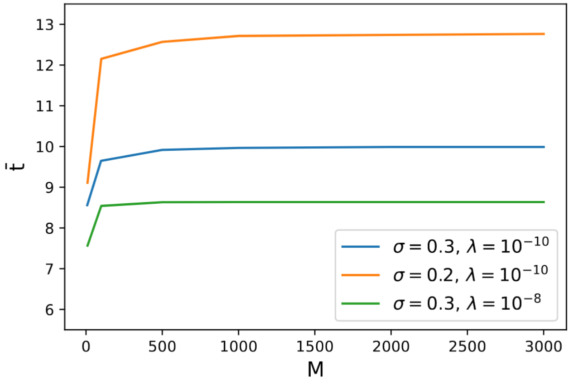
- M.L. et al, *Learning New Physics Efficiently with Nonparametric methods,* arXiv:2204.02317.
- G. Grosso et al, *Fast kernel methods for Data Quality Monitoring as a goodness-of-fit test*, arXiv:2303.05413.
- G. Grosso et al*, Goodness of Fit by Neyman-Pearson testing*, to appear soon.
- D’Agnolo et al, *Learning New Physics from a Machine,* arXiv:1806.02350.
- D’Agnolo et al, *Learning Multivariate New Physics,* arXiv:1912.12155.
- G. Meanti et al, *Kernel methods through the roof: handling billions of points efficiently,* arXiv:2006.10350.

DQM

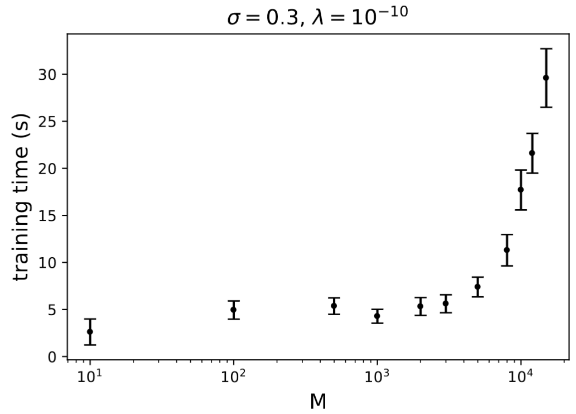
Training time

N_1	Reference	Cathode 75%	Cathode 50%	Cathode 25%	Threshold 75%	Threshold 50%	Threshold 25%
250	0.48±0.03	0.50±0.02	0.50±0.02	0.49±0.02	0.50±0.02	0.51±0.02	0.51±0.04
500	0.43±0.03	0.45±0.03	0.44±0.02	0.44±0.03	0.44±0.02	0.43±0.04	0.47±0.04
1000	0.57±0.05	0.52±0.06	0.52±0.05	0.58±0.05	0.53±0.05	0.55±0.05	0.59±0.05

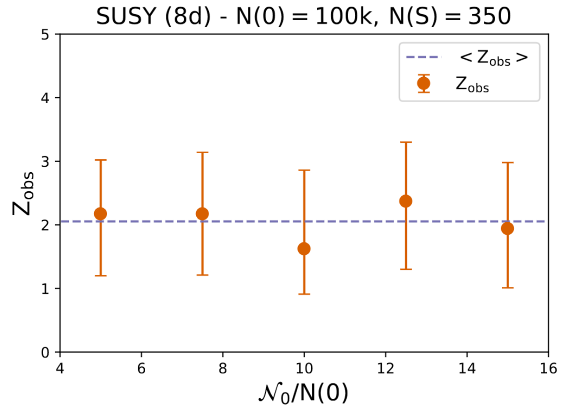
Hyperparameters



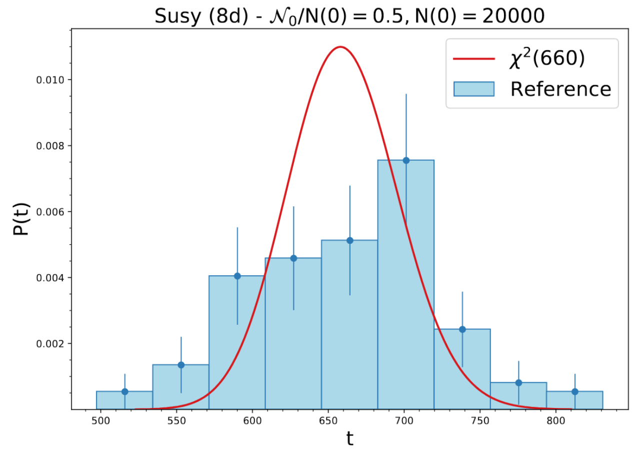
(a) Average test statistics as a function of the number of Nyström centers.



(b) Average training time as a function of the n ber of Nyström centers.



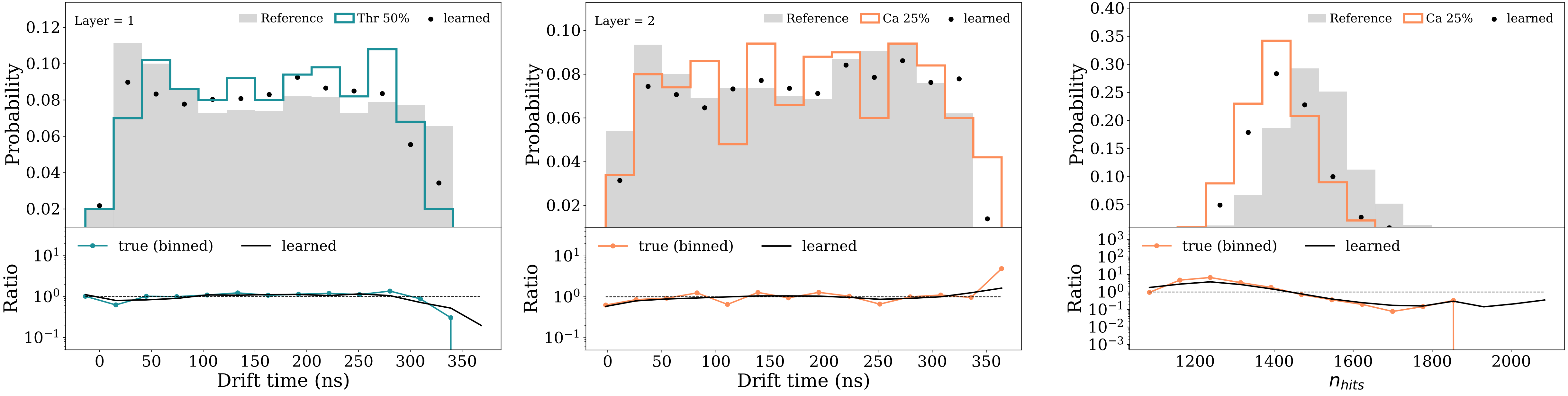
(a)



(b)

DQM

Reconstructed density ratios.



G. Grosso et al, arXiv:2303.05413.