Residual-ANODE (R-ANODE)

arXiv:2312.11629v1

Ranit Das[1],

Gregor Kasieczka[2] and David Shih[1]

[1] Rutgers University

[2] University of Hamburg

DPF-Pheno-2024

Date: 05/13/2024

Contents

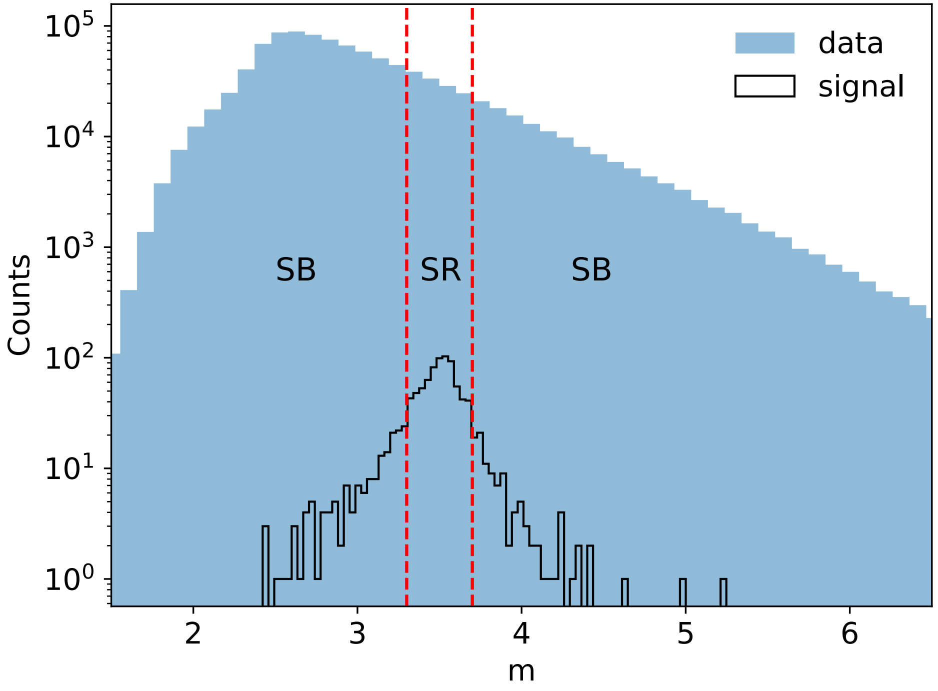
- Recap on ANODE
- R-ANODE method
- Dataset and Models
- Results

Resonant anomaly detection

- Assume we have a resonant variable m, and some other discriminating features x.

$$P_{data}(x,m) = w * P_S(x,m) + (1-w) * P_B(x,m)$$

- Signal Region(SR) and Side-Bands(SB) are defined with respect to the resonant variable m.



Data-driven anomaly detection techniques

Density Estimation Based approaches

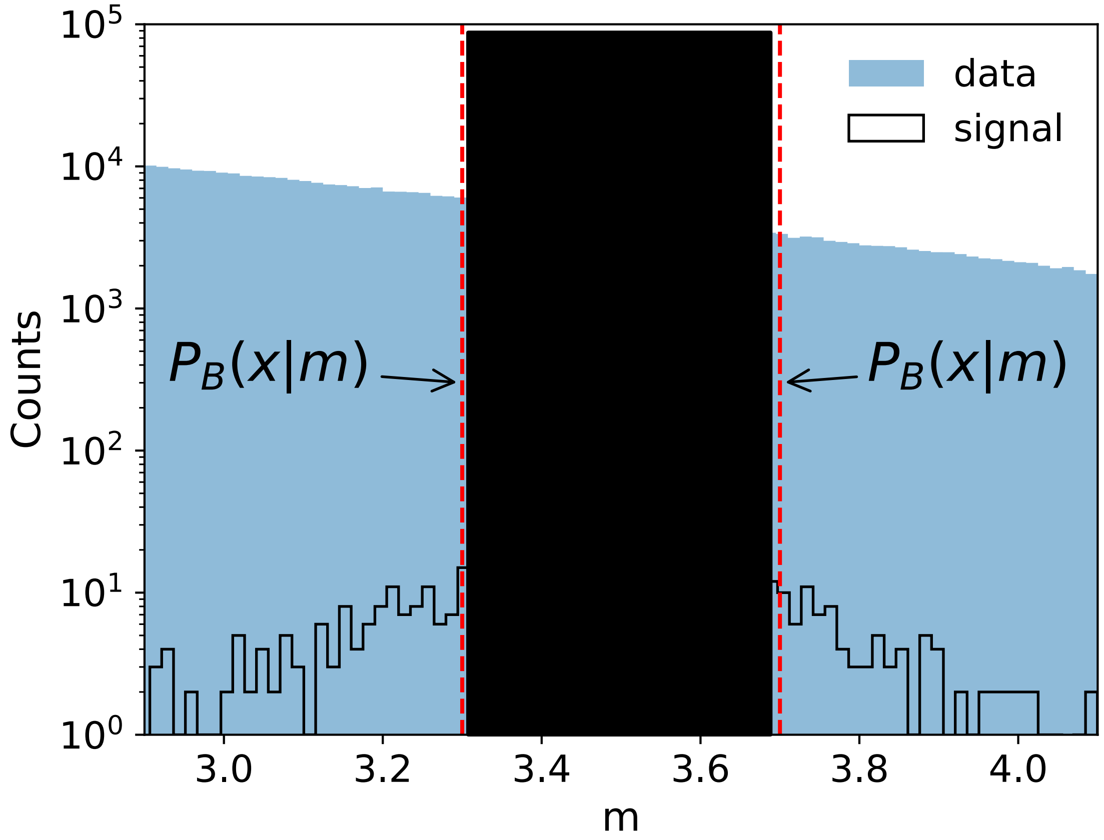
- ANODE(arXiv:2001.04990v2)
- **R-ANODE (this talk!)**

Classifier Based approaches

- CATHODE (arXiv:2109.00546v3)
- CURTAINS (arXiv:2203.09470v3)
- CWoLA (arXiv:1902.02634v2)
- Ideal AD (Ideal version of CATHODE, CURTAINS and CWoLA) (arXiv:2109.00546v3)

etc …

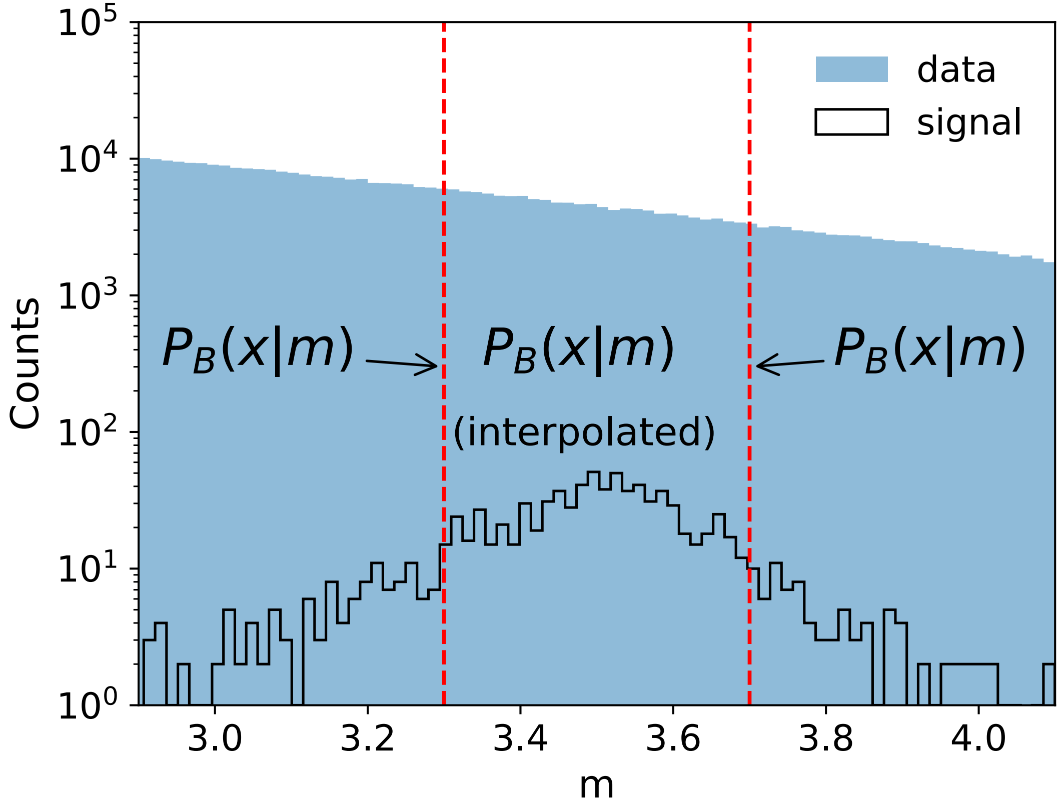
ANODE



- A conditional density estimator is trained to learn $P_B(x|m \in SB)$ in the side-bands(SB).

Anomaly Detection with Density Estimation (arXiv:2001.04990v2)
Anomaly Detection in the Presence of Irrelevant Features arXiv:2310.13057v1

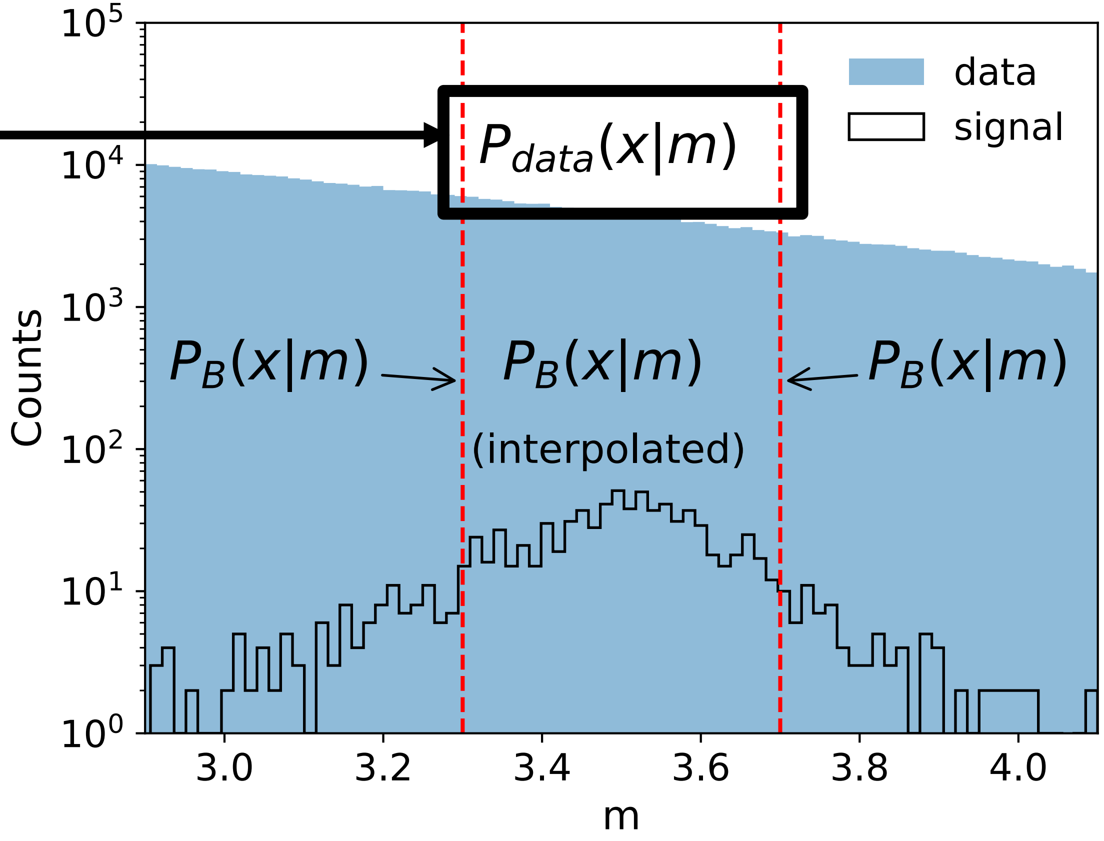
ANODE



- A conditional density estimator is trained to learn $P_B(x|m \in SB)$ in the side-bands(SB).
- The learned $P_B(x|m)$ is used to interpolate into the SR

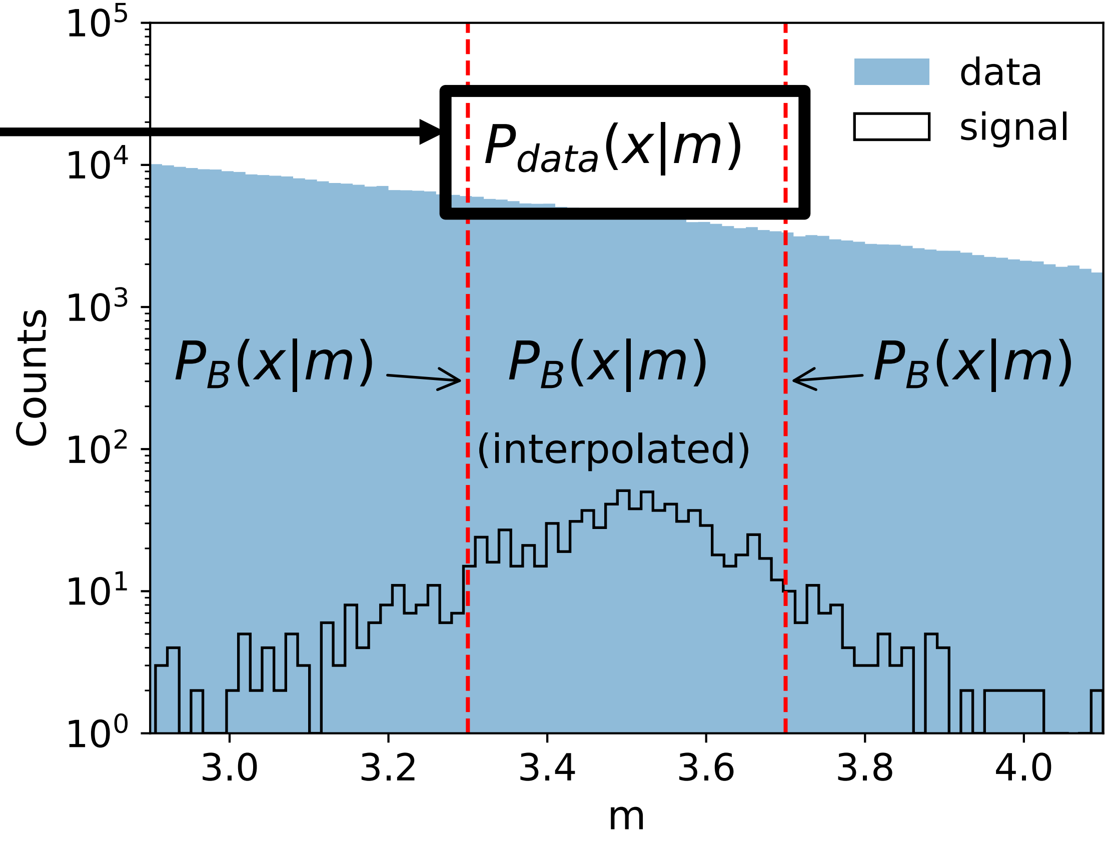
ANODE

In SR, directly learn



ANODE

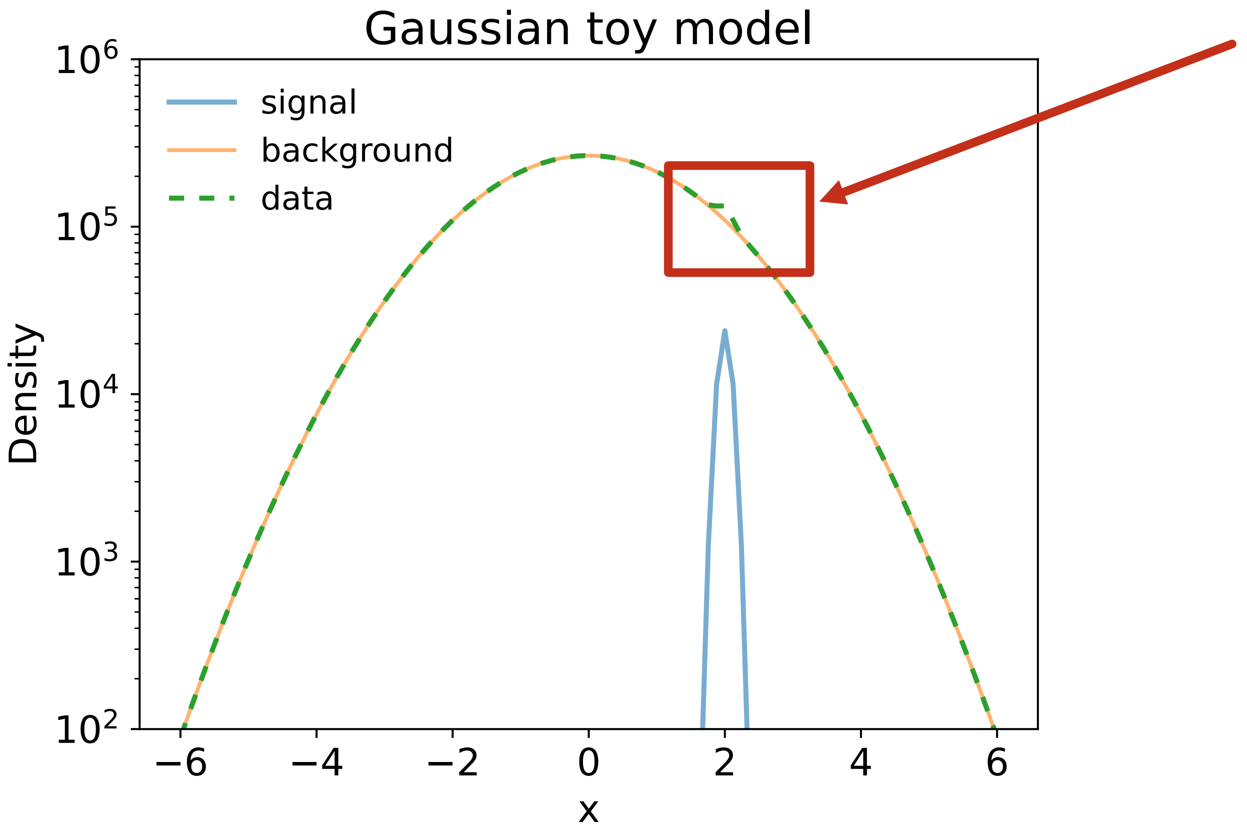
In SR, directly learn



Anomaly score: $$R(x|m) = \frac{\mathrm{P_{data}}(x|m \in SR)}{P_B(x|m \in SR)}$$

ANODE

In SR: Learn $P_{\mathrm{data}}(x|m)$

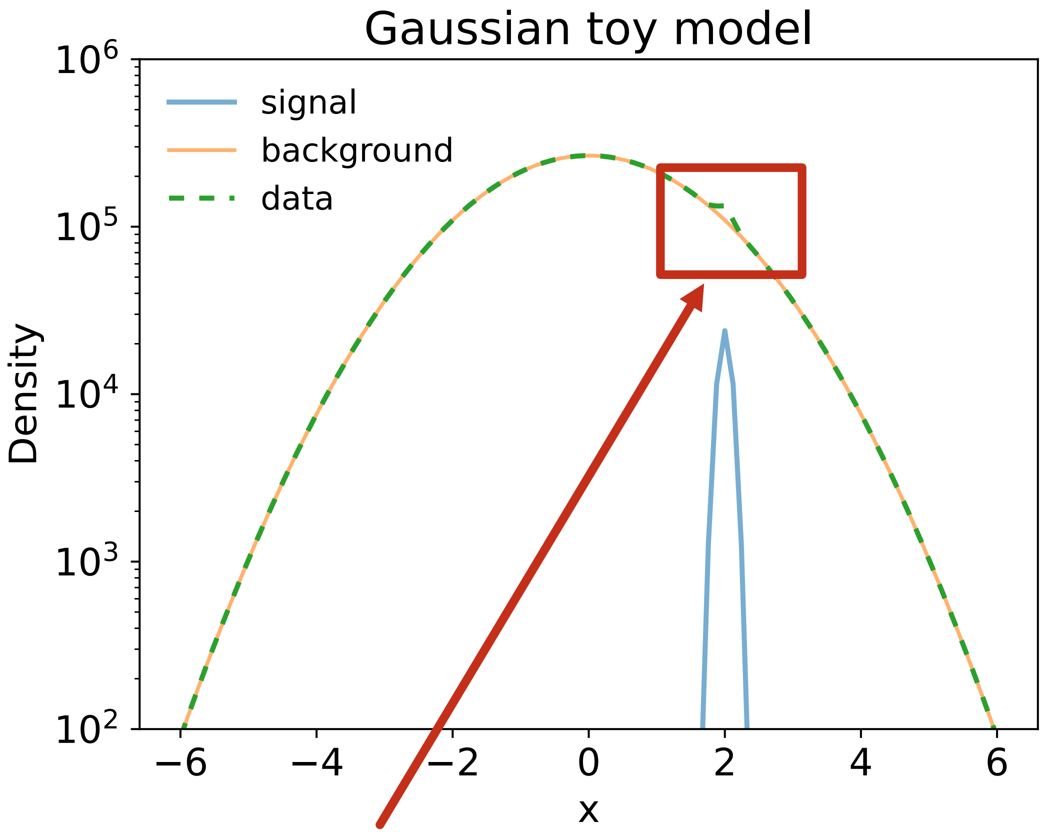


ANODE must learn the sharply peaked distributions in x where the signal is localized.

Given the small amount of signal events, this is a hard task for a generative model

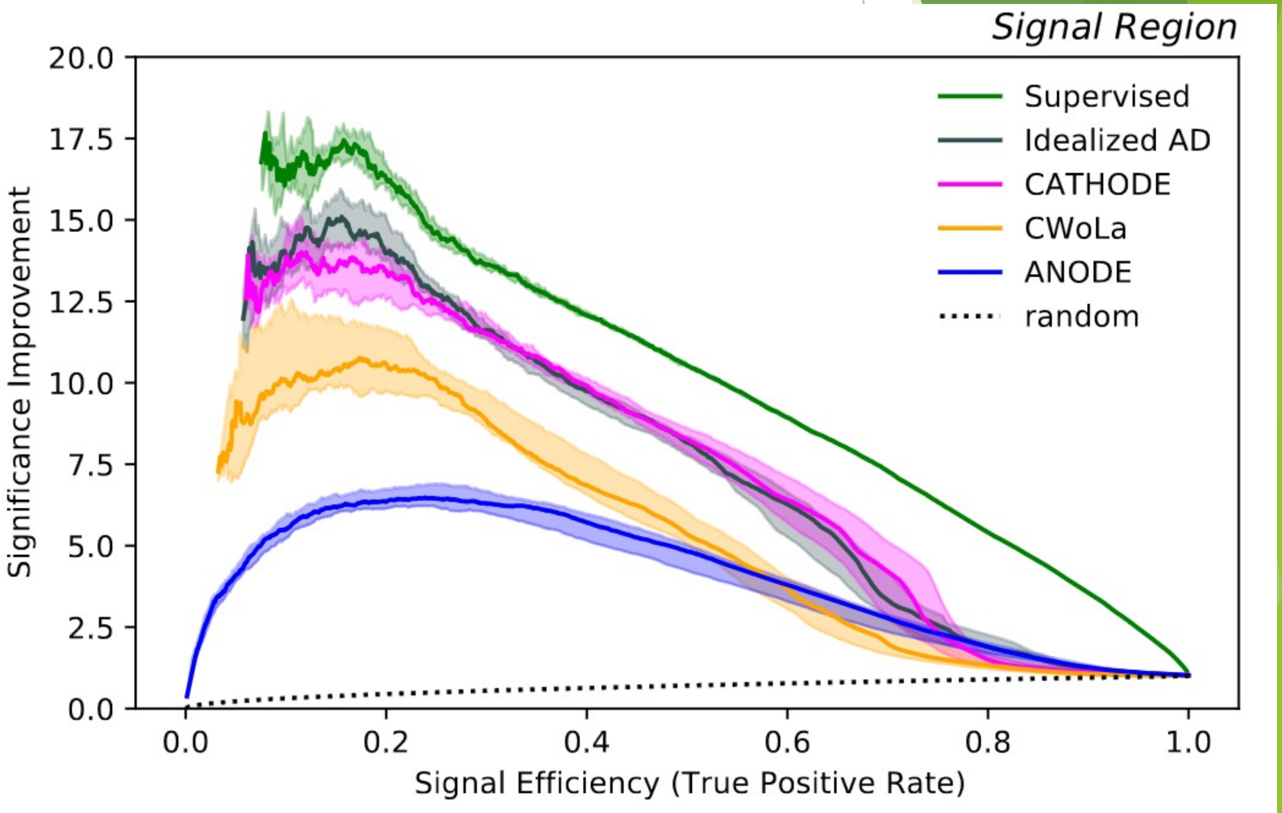
ANODE

In SR: Learn $P_{data}(x|m)$



ANODE must learn the sharply peaked distributions in x where the signal is localized.

Classifying Anomalies THrough Outer Density Estimation (CATHODE) arXiv:2109.00546v3

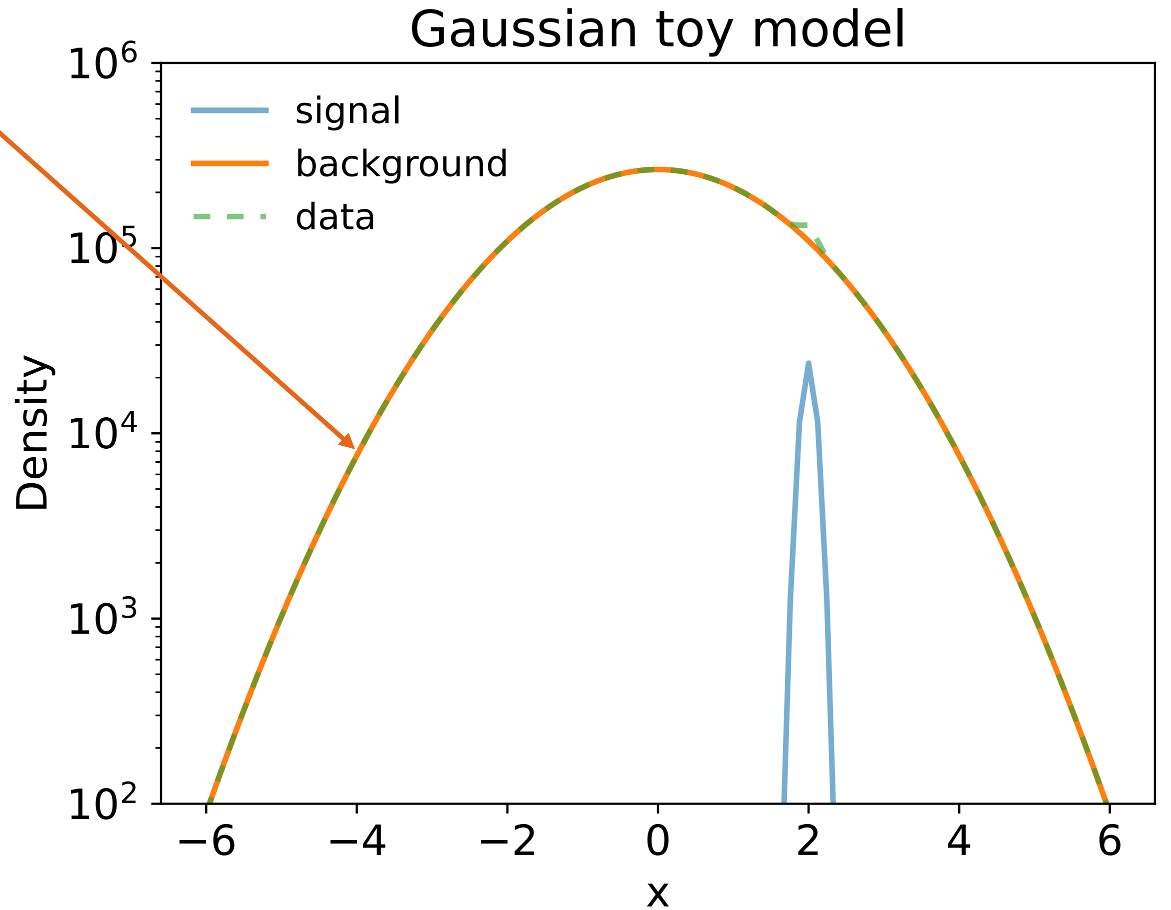


Worse performance than classifier-based approaches

R-ANODE (new method)

In the SR,

- Hold the interpolated $P_B(x, m)$ fixed



R-ANODE

In the SR,

- Hold the interpolated $\boldsymbol{P_B(x,m)}$ fixed.
- Directly model $\boldsymbol{P_S(x,m)}$ with a normalizing flow by fitting to data:

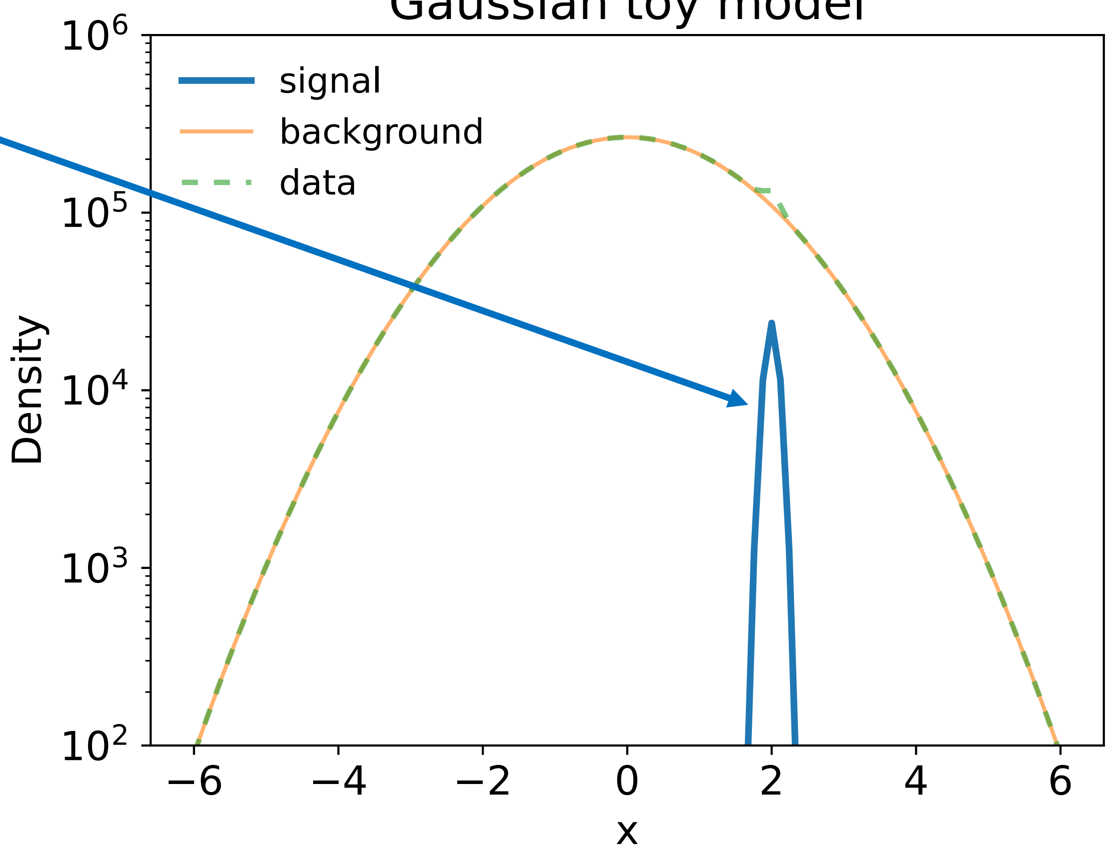
$$P_{data}(x,m) =$$

$$w * \boldsymbol{P_S(x,m)} + (1-w) * \boldsymbol{P_B(x,m)}$$

(Normalizing Flow)

(hold fixed)

Gaussian toy model



R-ANODE

$$P_{data}(x,m) = \boxed{\boldsymbol{w}} * \boldsymbol{P_S(x,m)} + (1 - \boldsymbol{w}) * \boldsymbol{P_B(x,m)}$$

(Normalizing Flow)

(hold fixed)

- Hold $\boldsymbol{w}$ fixed and scan over different $\boldsymbol{w}'s$ as working points
- Learn $\boldsymbol{w}$

R-ANODE (ideal): $\boldsymbol{w}$ fixed to the true $\boldsymbol{w}$-value

R-ANODE

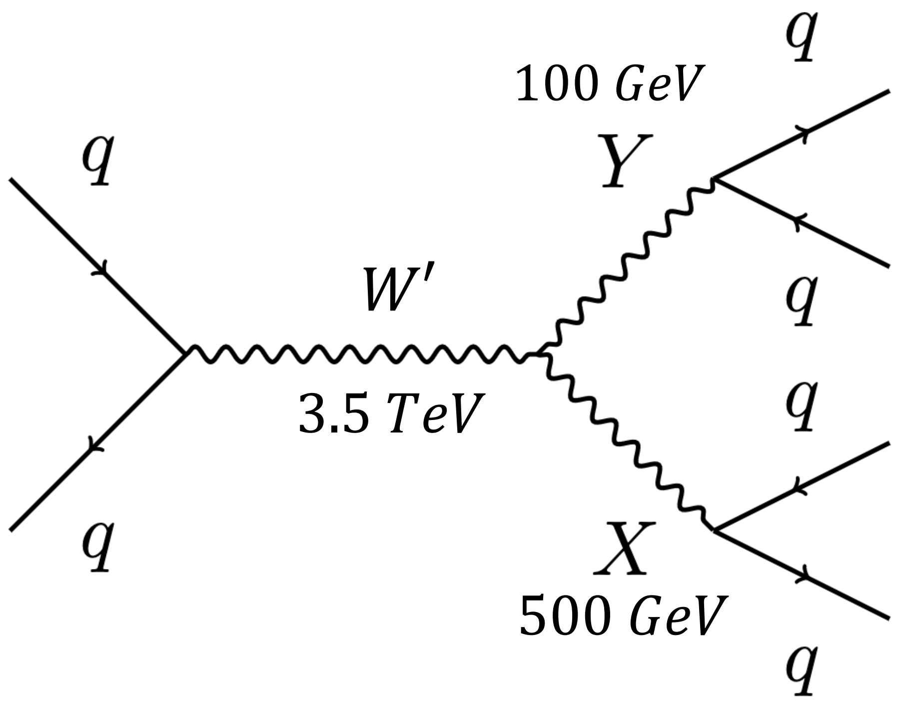
Loss:

$$\boxed{\text{Minimize: } -\log(P_{data}(x, m))}$$

- w.r.t parameters of $\boldsymbol{P_S(x,m)}$, holding $\boldsymbol{w}$ fixed
- w.r.t parameters of $\boldsymbol{P_S(x,m)}$ and $\boldsymbol{w}$

Dataset

- The LHC Olympics R&D dataset :
- Data: 1M QCD di-jet events as background and different amounts of signal events.



The LHC Olympics 2020: A Community Challenge for Anomaly Detection in High Energy Physics
: arXiv:2101.08320

Dataset

- The SR : $3.3\ TeV < m_{JJ} < 3.7\ TeV$
- The resonant variable is m_{JJ}, and the features x are $[m_{J1}, m_{J2} - m_{J1}, \tau_{21}^{J1}, \tau_{21}^{J2}]$
- Initial signal injection: N_{sig} = 1000(~770 in SR), $\mathrm{S/B} \sim 6 \times 10^{-3}$, $\mathrm{S}/\sqrt{B} \sim 2.2$

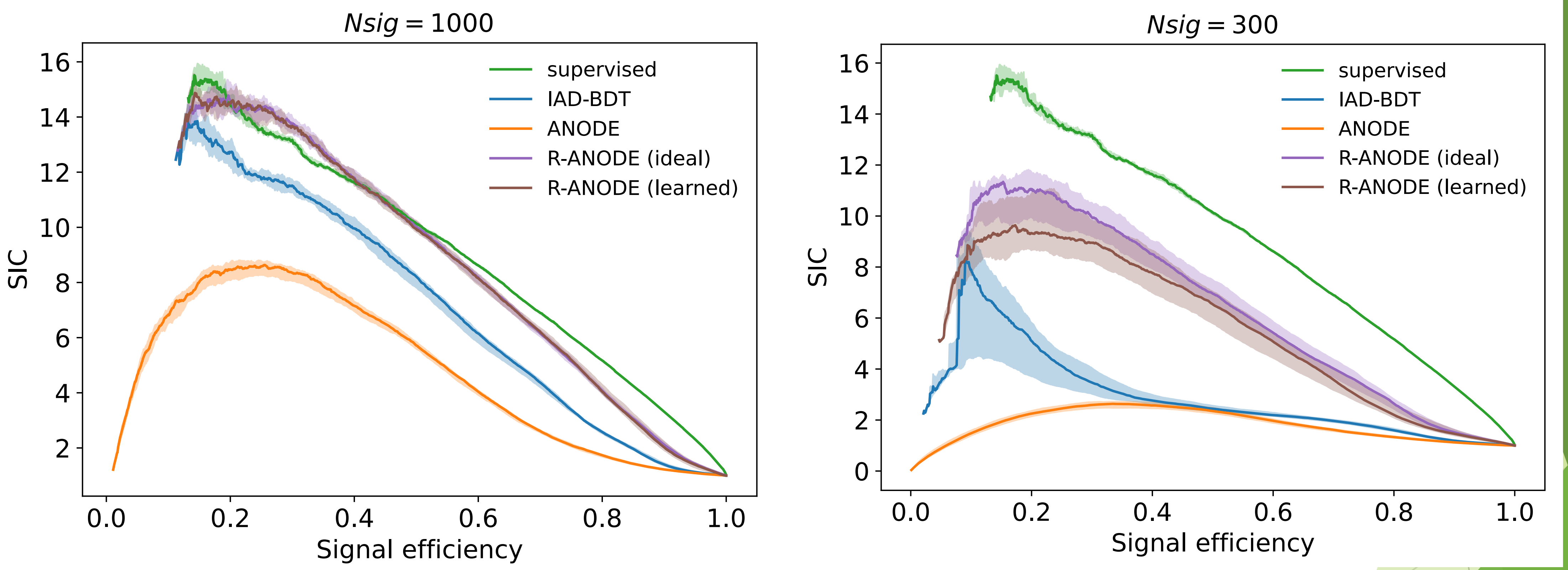
The LHC Olympics 2020: A Community Challenge for Anomaly Detection in High Energy Physics
: arXiv:2101.08320

Model architecture and hyperparameters

- The background model is the same as CATHODE/ANODE (arXiv:2001.04990v2, arXiv:2109.00546v3): Masked Autoregressive Flow (MAF) with affine transformations.
- For the signal model for $\boldsymbol{P_S(x,m)}$, we use RQS transformations with MADE blocks.
- For proof of concept, we use the true background density $\boldsymbol{P_B(m)}$ estimated from histograms of the background in SR.
- We also update the ANODE model to $\boldsymbol{P_{data}(x|m)}$, to the same RQS-based model, to compare R-ANODE vs ANODE

SIC Curves

$$SIC = TPR/\sqrt{FPR}$$



R-ANODE improves ANODE and also gives better SIC Curves than the idealized-AD

Back To The Roots: Tree-Based Algorithms for Weakly Supervised Anomaly Detection arXiv:2309.13111

Classifier based approaches

In SR:

Ideal-Anomaly Detector (IAD)

Perfectly Simulated background

vs

Data (mixture of signal and background)

Classification

Ideal AD is an ideal version of classifier-based approaches

Classifying Anomalies THrough Outer Density Estimation (CATHODE) arXiv:2109.00546v3
Full Phase Space Resonant Anomaly Detection arXiv:2310.06897v2
The Interplay of Machine Learning--based Resonant Anomaly Detection Methods arXiv:2307.11157v1
Back To The Roots: Tree-Based Algorithms for Weakly Supervised Anomaly Detection arXiv:2309.13111v1
Combining Resonant and Tail-based Anomaly Detection arxiv:2309.12918
Extending the Bump Hunt with Machine Learning arXiv:1902.02634
Anomaly Detection in the Presence of Irrelevant Features arXiv:2310.13057v1

Classifier based approaches

In SR:

Ideal-Anomaly Detector (IAD)

Perfectly Simulated background

vs

Data (mixture of signal and background)

Classification

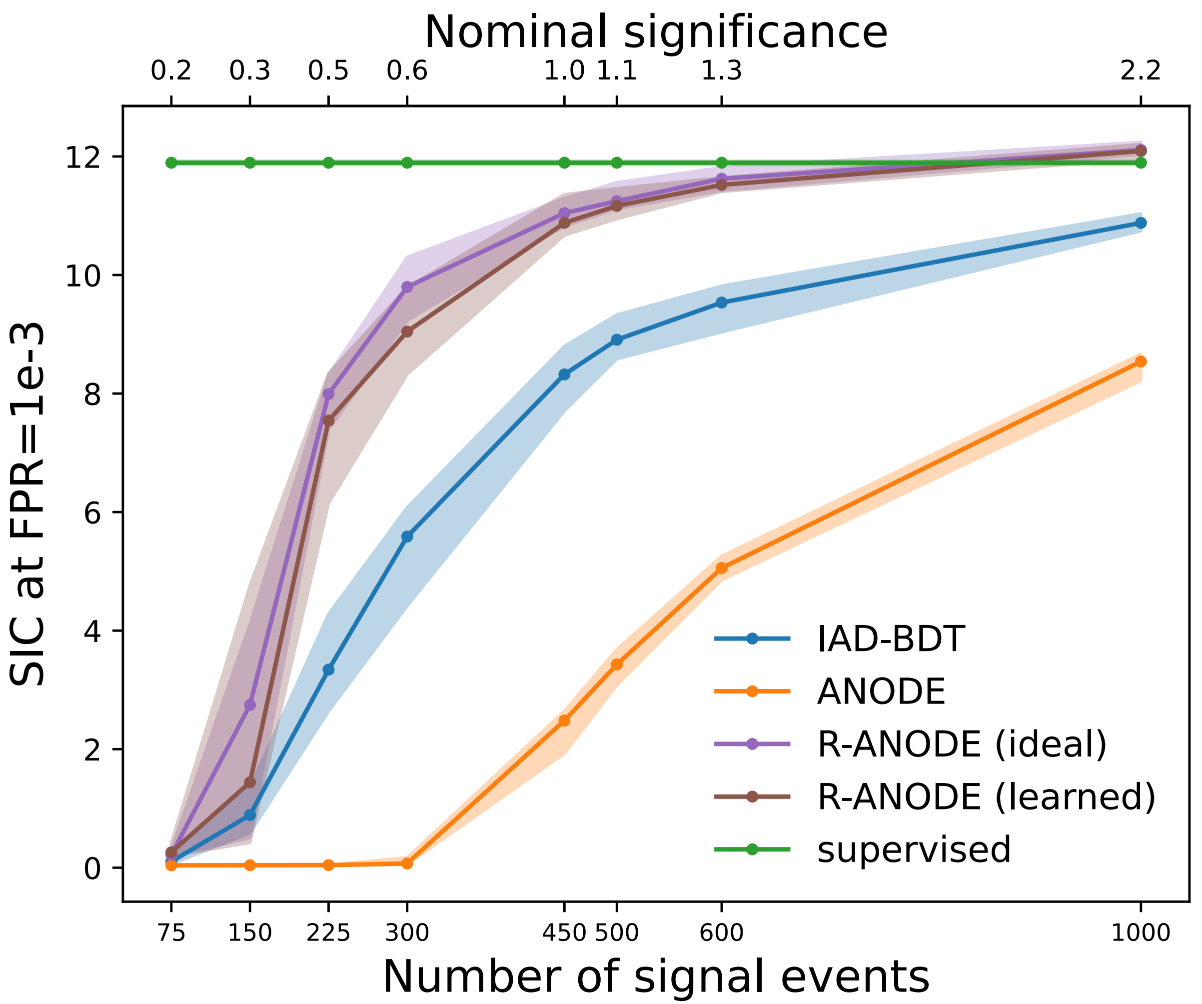
Ideal AD is an ideal version of CATHODE

It’s possible to exceed the IAD performance, if not using a classifier-based approach.

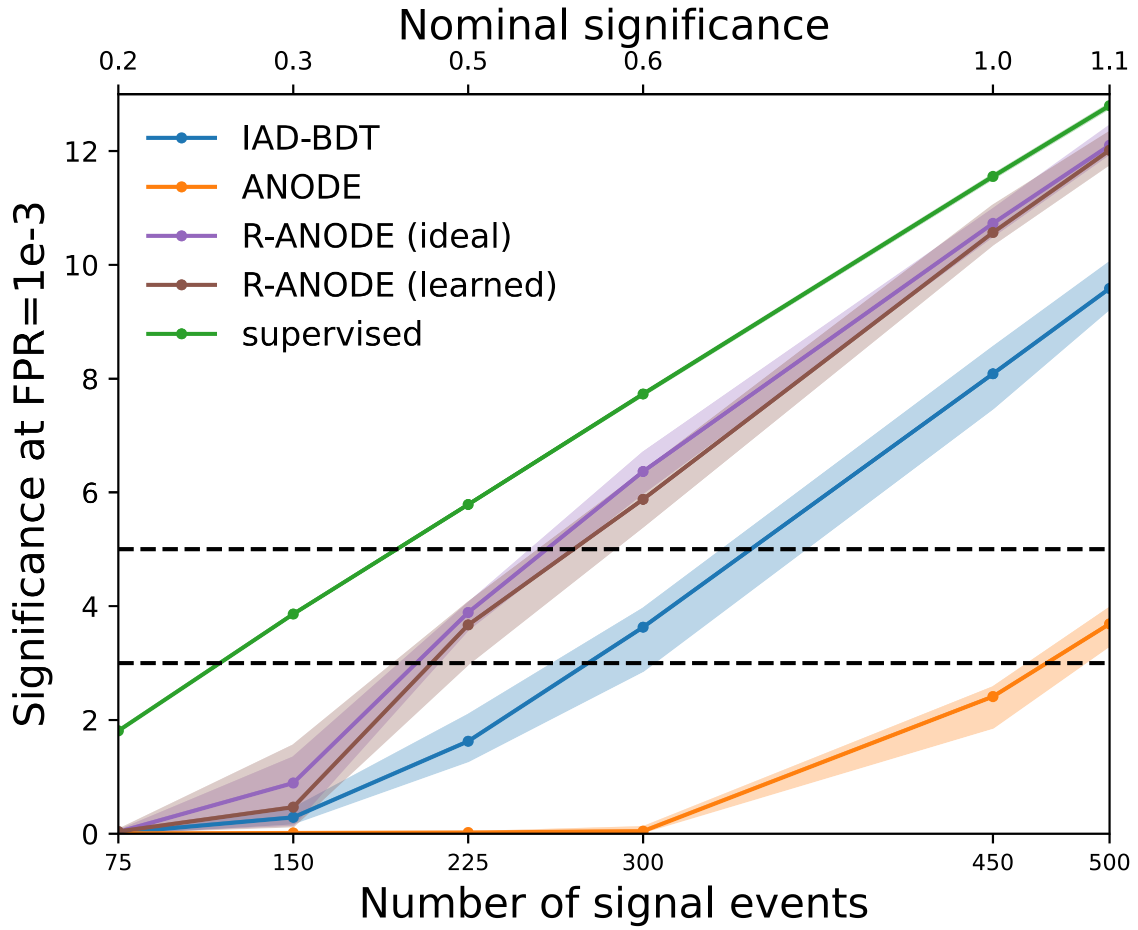
Supervised is the true upper limit for performance

Classifying Anomalies THrough Outer Density Estimation (CATHODE) arXiv:2109.00546v3
Full Phase Space Resonant Anomaly Detection arXiv:2310.06897v2
The Interplay of Machine Learning--based Resonant Anomaly Detection Methods arXiv:2307.11157v1
Back To The Roots: Tree-Based Algorithms for Weakly Supervised Anomaly Detection arXiv:2309.13111v1
Combining Resonant and Tail-based Anomaly Detection arxiv:2309.12918
Extending the Bump Hunt with Machine Learning arXiv:1902.02634
Anomaly Detection in the Presence of Irrelevant Features arXiv:2310.13057v1

Nsig vs SIC @ FPR=0.001

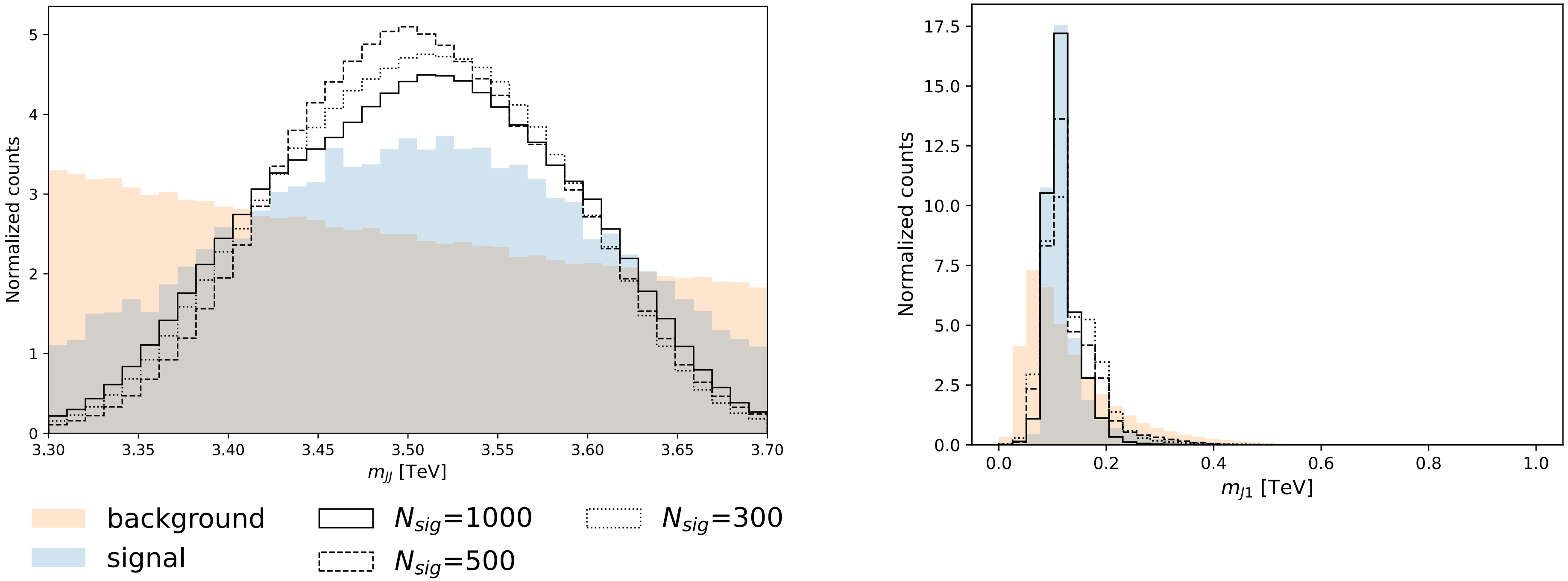


Nsig vs Significance



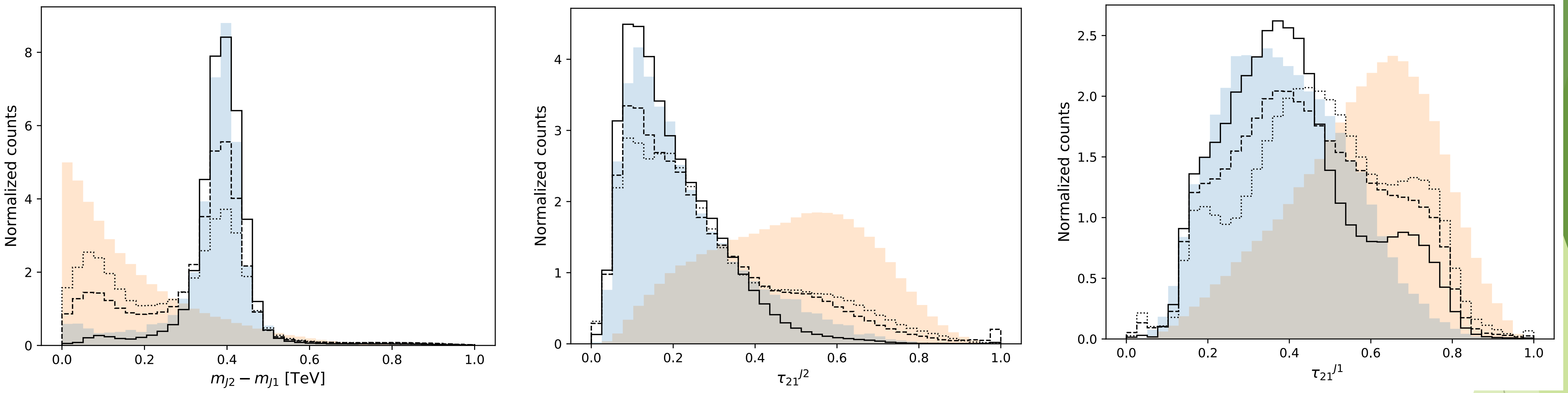
$$Significance = SIC * \frac{S}{\sqrt{B}}$$

Samples from $P_S(x, m)$



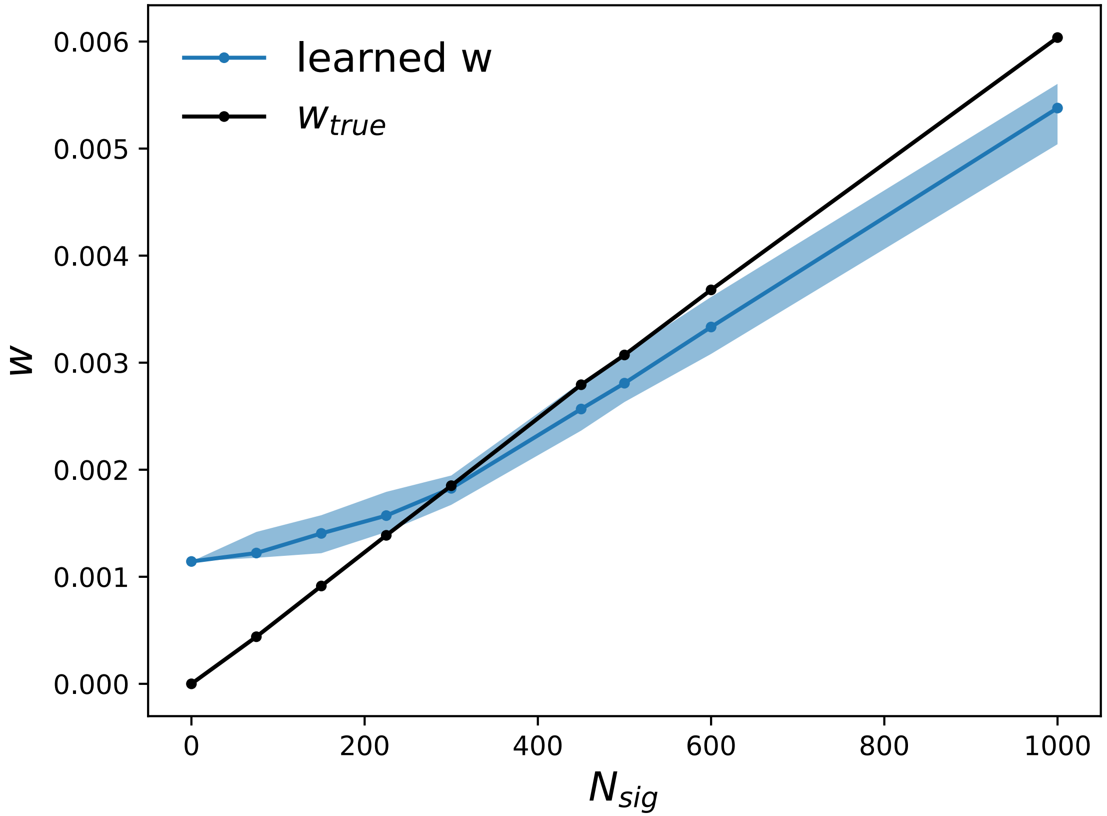
- Directly learning the signal distributions $P_S(x, m)$ leads to a more interpretable method.
- This could give us information about the signal: eg: mass of subjet, Pronginess of subjet.

Samples from $P_S(x, m)$



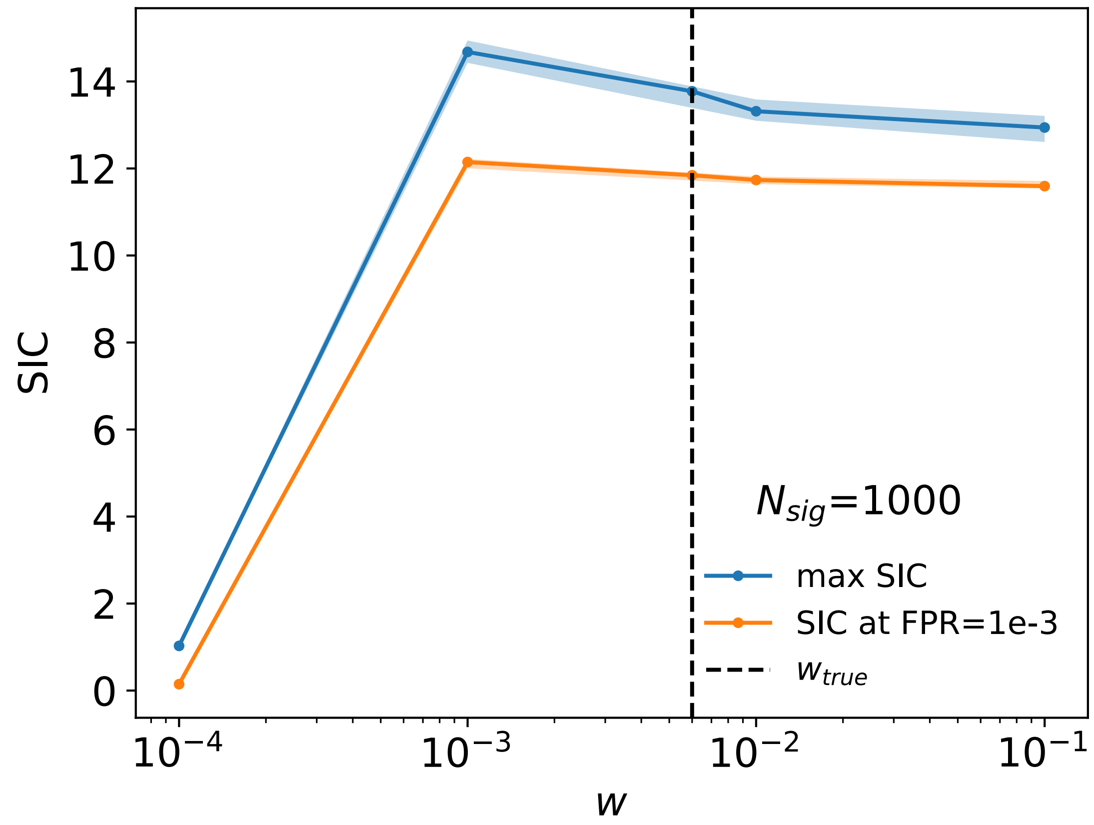
- Directly learning the signal distributions $P_S(x, m)$ leads to a more interpretable method.
- This could give us information about the signal: eg: mass of subjet, Pronginess of subjet.

Learned w



Learned w is very close to the true w values

Scanning over w



SIC is robust to incorrect choice of w, and could be used to put a lower bound on w

Conclusions

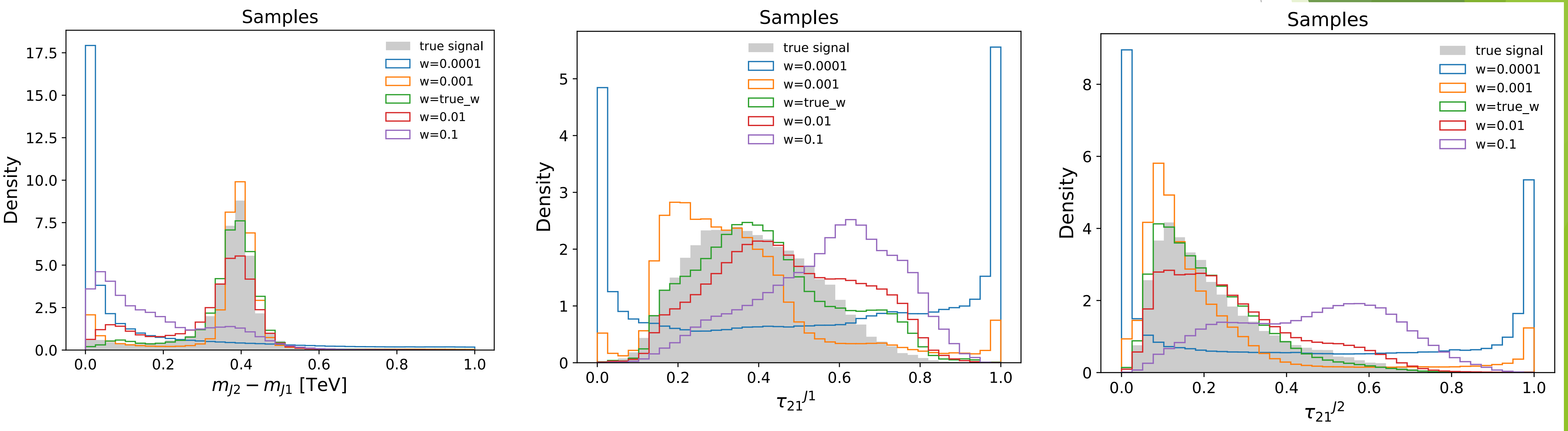
- R-ANODE improves ANODE and exceeds the performance of CATHODE and IAD.
- R-ANODE can learn w- values very close to the true w.
- Performance of R-ANODE is robust to the incorrect choice of w.
- R-ANODE directly learns the signal distribution, which allows us to draw samples directly from the signal distribution.

Future directions

- Study how irrelevant features affect the performance
- Apply this method with bump-hunt
- Study the effects of sculpting

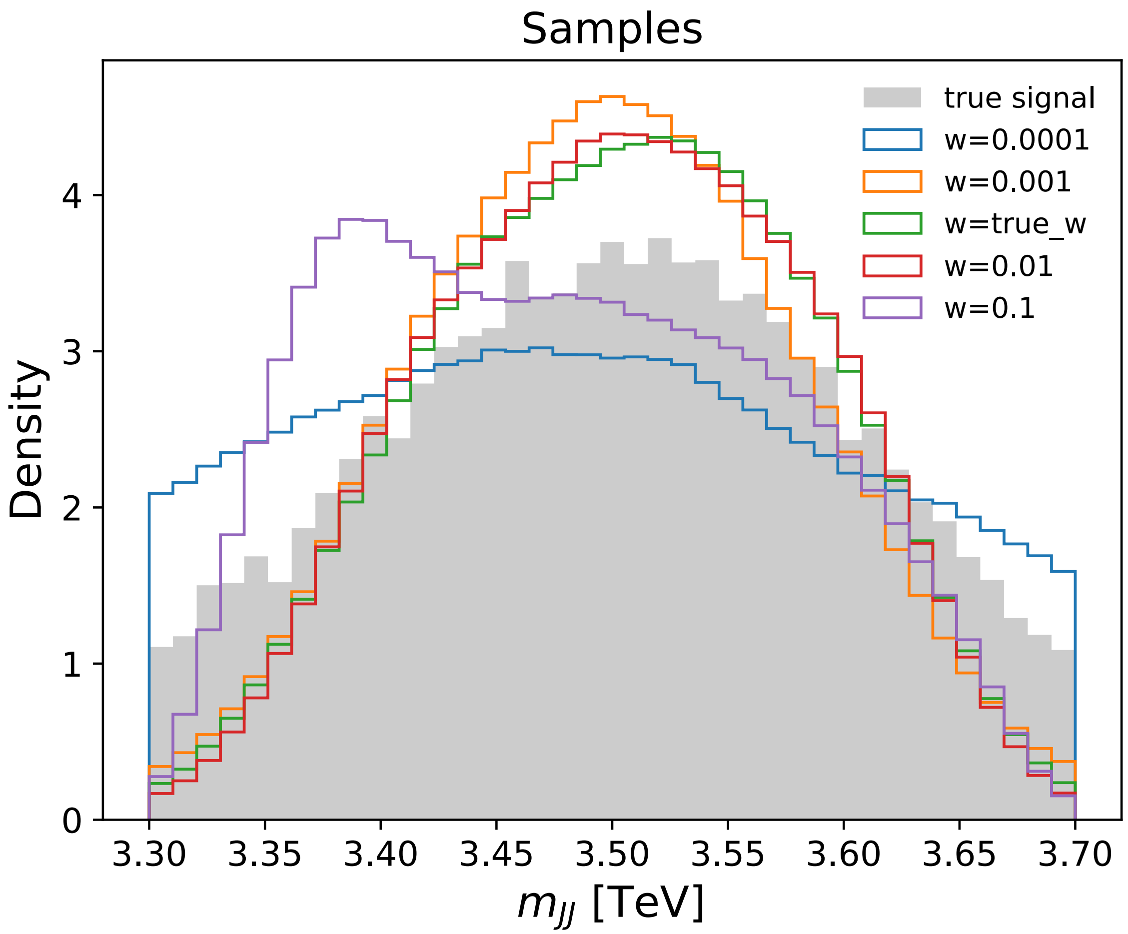
THANK YOU

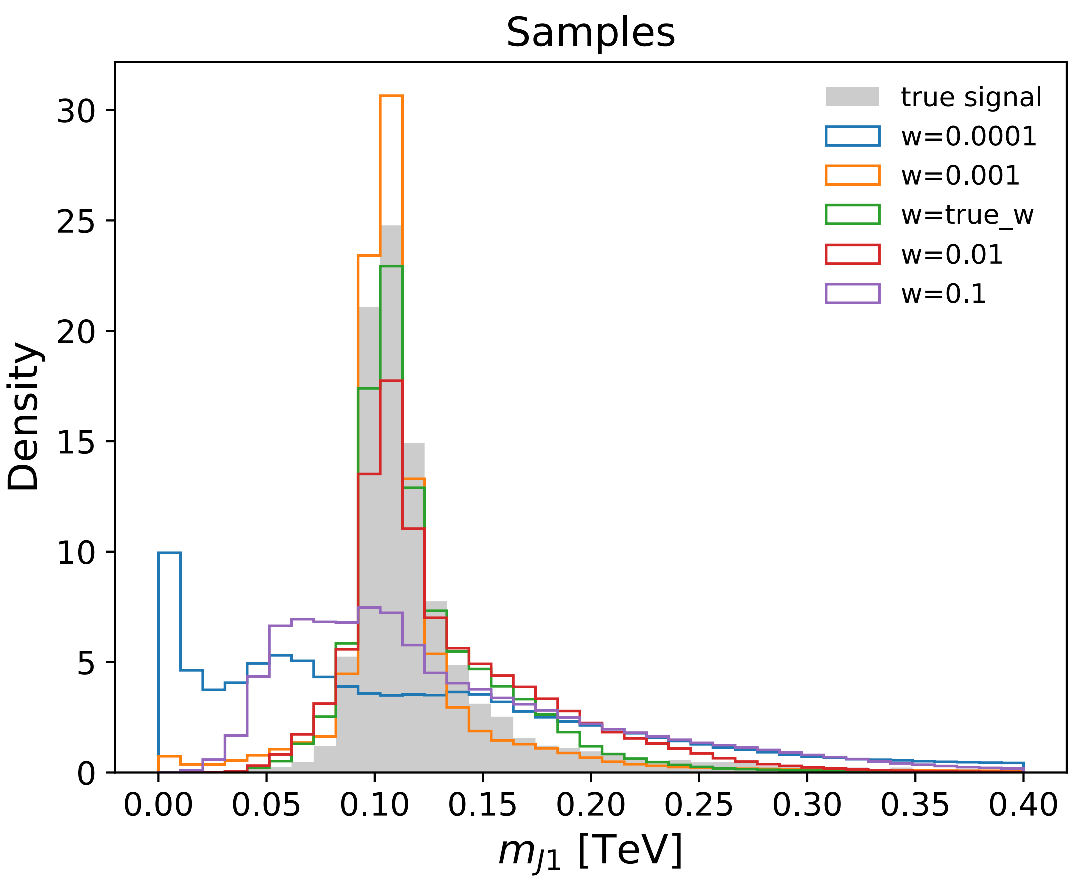
Samples for different w



With learned $\boldsymbol{P_S(x, m)}$

Samples for different w





With learned $\boldsymbol{P_S(x, m)}$

Ensembling

- For each signal injection, we resample the the signal 10 times. For each resample, we shuffle and split the data 20 times into training-validation splits (80-20) and train the model.
- For each resample, ensembling is done with 10 lowest validation loss models from each training, and 20 re-trainings (200 models).
- Similarly, the IAD-BDT we train HistGradientBoosting classifer, with default hyperparameters for 200 epochs, but shuffle-and split the data and retrained it 50 times (50-50), for ensembling.

Model architecture and hyperparameters

- For the signal model for $\boldsymbol{P_S(x, m)}$ we use RQS transformations with 6 MADE blocks, with block consisting of 2 hidden layers with 64 nodes each, dropout=0.2, and batch-normalization is applied in between layers.
- The RQS-model for all cases is trained with a learning rate = 0.0003, with the AdamW optimizer, with a batch size of 256, for 300 epochs.