Classifying Anomalies Through Outer Density Estimation (CATHODE)

Introducing a new model-agnostic search strategy for resonant new physics at the LHC and beyond

Mitchell Conference on Collider, Dark Matter, and Neutrino Physics 2022

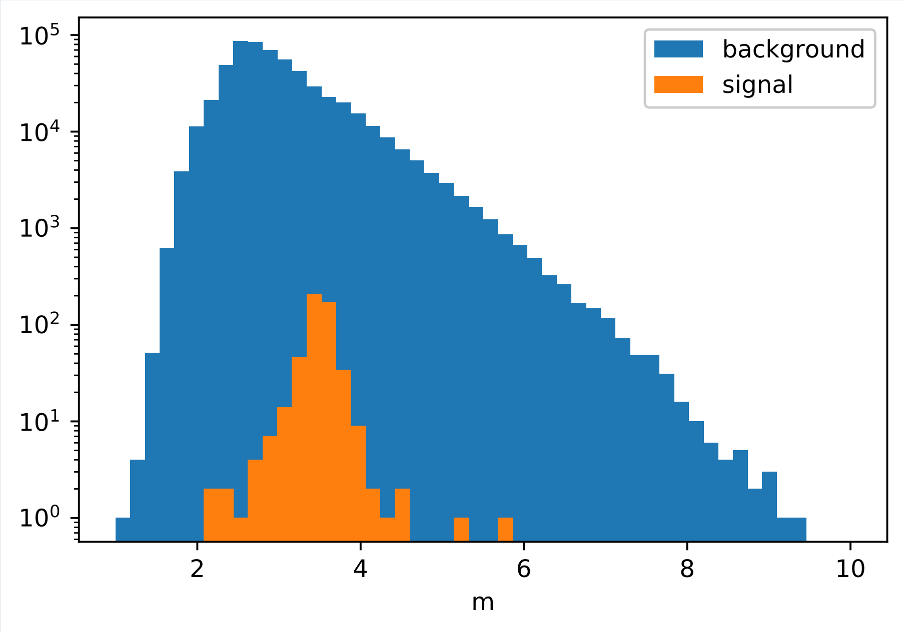
arXiv 2109.00546

Anna Hallin[1], Joshua Isaacson[2], Gregor Kasieczka[3], Claudius Krause[1], Benjamin Nachman[4], Tobias Quadfasel[3], Matthias Schlaffer[5,6], David Shih[1], Manuel Sommerhalder[3]

[1]Rutgers University
[2]Fermi National Accelerator Laboratory
[3]University of Hamburg
[4]Lawrence Berkeley National Laboratory
[5]University of Chicago
[6]University of Geneva

Introduction

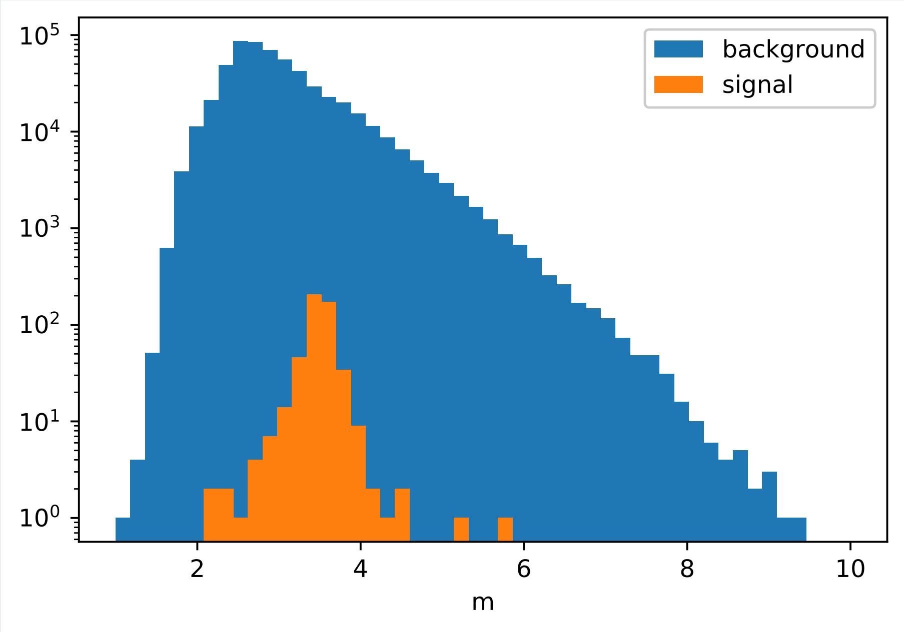
CATHODE is a new method for **model agnostic anomaly searches**.

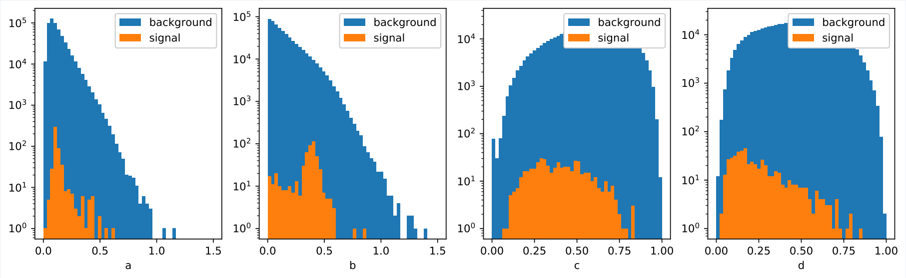


Introduction

CATHODE is a new method for **model agnostic anomaly searches**.

Assume we have a resonant variable m, and some other discriminating features $\mathbf{x}$.



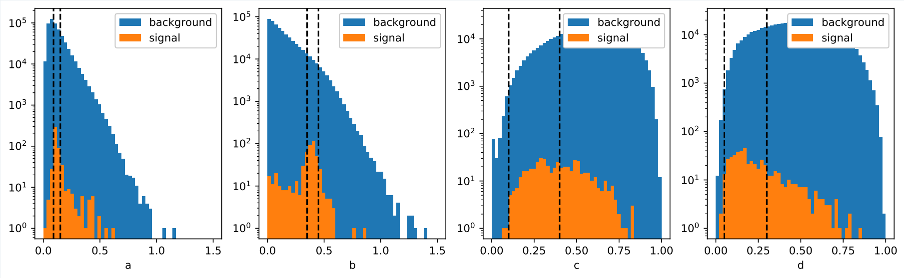


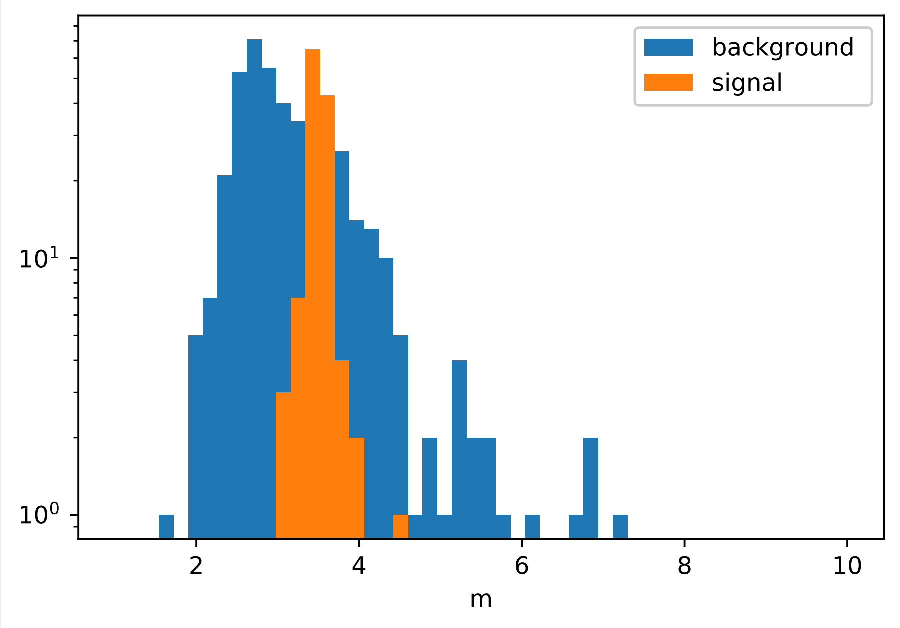
Introduction

CATHODE is a new method for **model agnostic anomaly searches**.

Assume we have a resonant variable m, and some other discriminating features $\mathbf{x}$.

If we knew where the signal was, we could place cuts on these features to reject background while retaining signal.





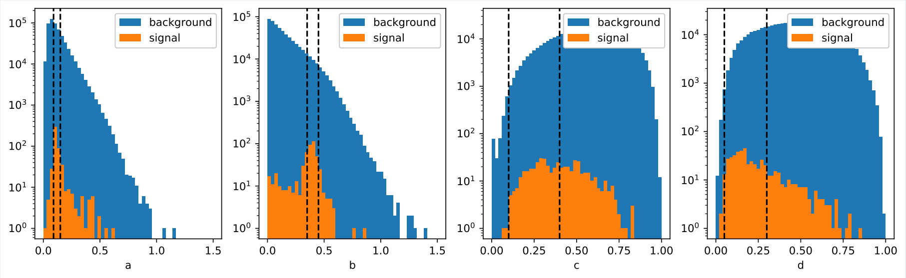
Introduction

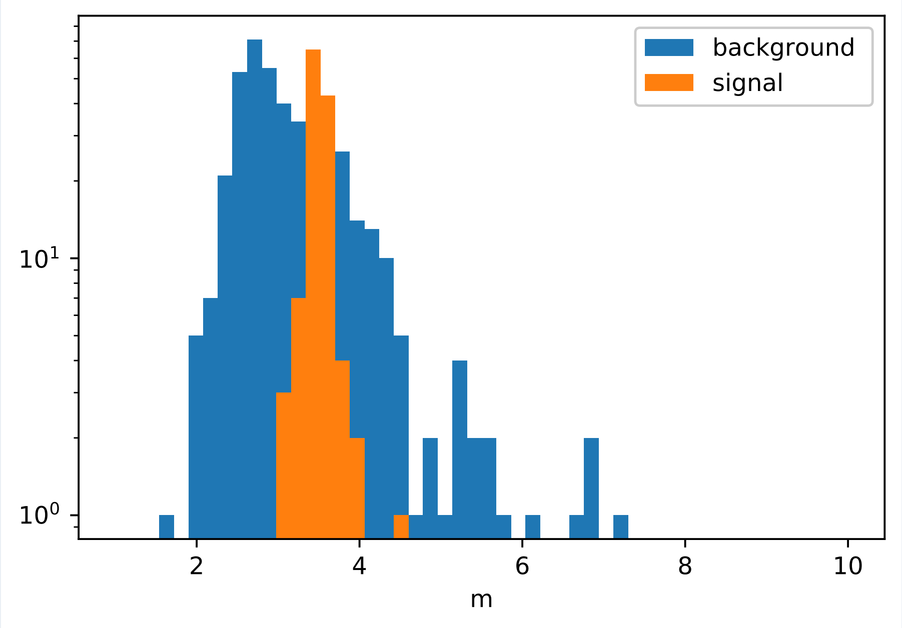
CATHODE is a new method for **model agnostic anomaly searches**.

Assume we have a resonant variable m, and some other discriminating features $\mathbf{x}$.

If we knew where the signal was, we could place cuts on these features to reject background while retaining signal.

This, however, requires us to model both signal and background. Furthermore, it is impossible to cover all possible models in all possible configurations.





Introduction

CATHODE is a new method for **model agnostic anomaly searches**.

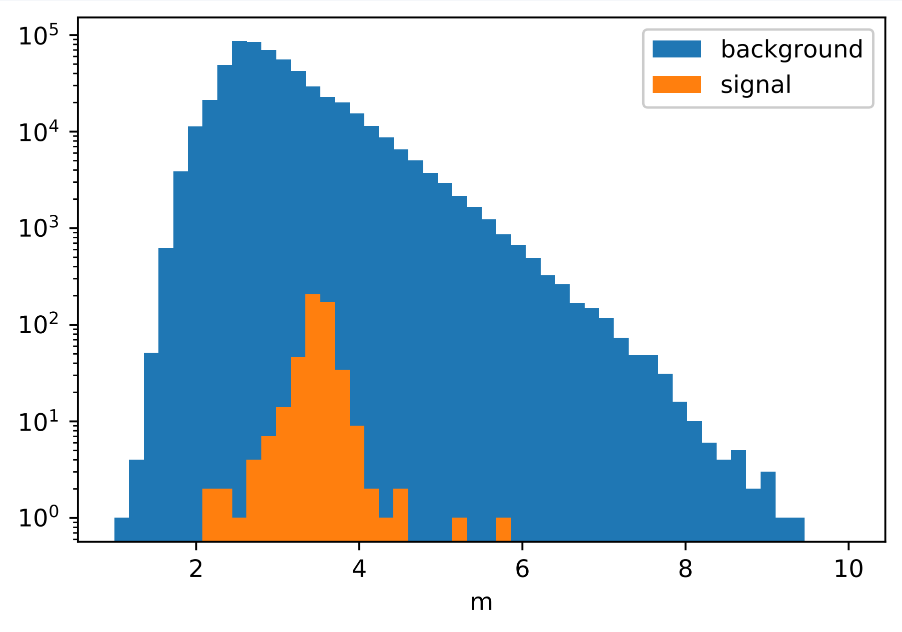
Assume we have a resonant variable m, and some other discriminating features $\mathbf{x}$.

If we knew where the signal was, we could place cuts on these features to reject background while retaining signal.

This, however, requires us to model both signal and background. Furthermore, it is impossible to cover all possible models in all possible configurations.

→ There is a need for **model-agnostic methods**.

But if we don't know the individual distributions of the signal and background, how can we find the (presumably) tiny signal in the giant haystack of background?



Probability densities and background

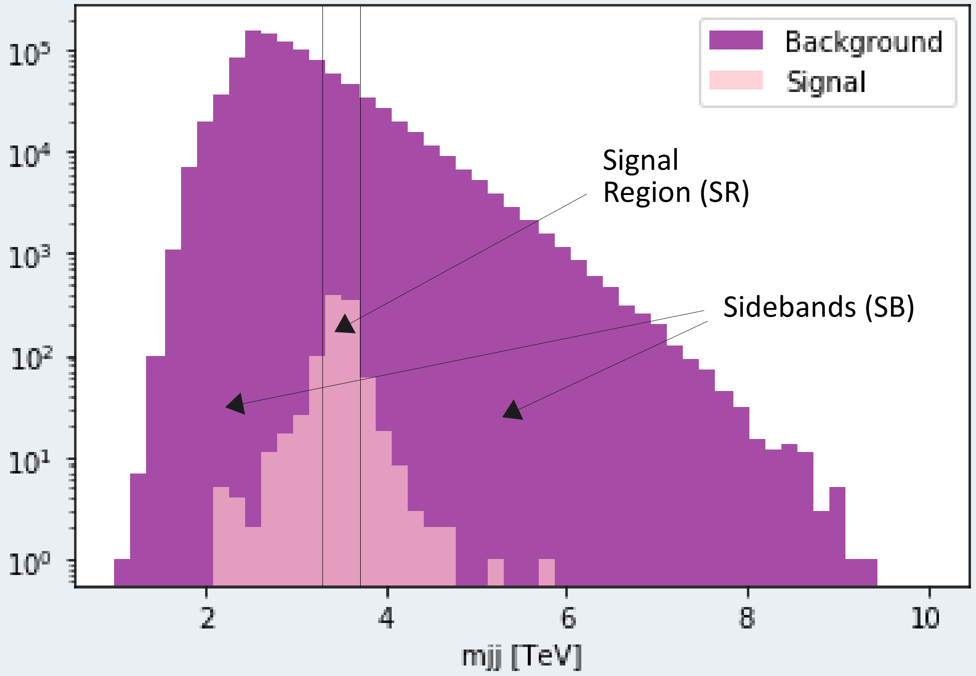
Assume we have a resonant variable m_{JJ}, and some other features $\mathbf{x}$.

$$p_{data}(m_{JJ}, \mathbf{x}) = \varepsilon\, p_{signal}(m_{JJ}, \mathbf{x}) + (1 - \varepsilon) p_{background}(m_{JJ}, \mathbf{x})$$

How to find $p_{bg}(m_{JJ}, \mathbf{x})$ for a localized signal?

3 different approaches:

- Find $p_{bg}(m_{JJ}, \mathbf{x})$ via **simulation** – but does this accurately represent the background in the data?



Probability densities and background

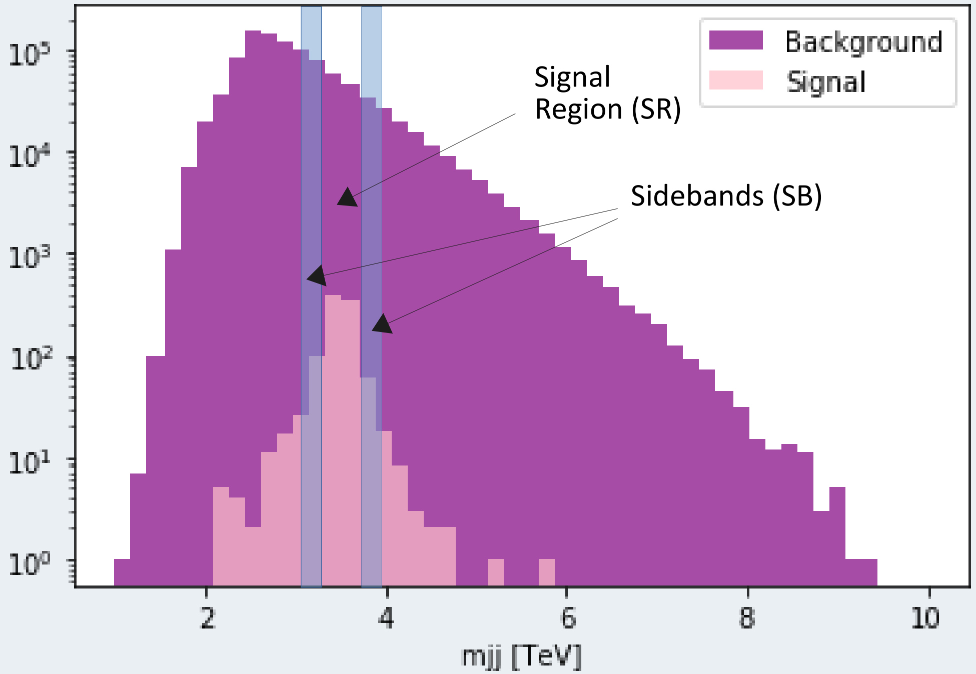
Assume we have a resonant variable m_{JJ}, and some other features $\mathbf{x}$.

$p_{data}(m_{JJ}, \mathbf{x}) = \varepsilon\, p_{signal}(m_{JJ}, \mathbf{x}) + (1 - \varepsilon) p_{background}(m_{JJ}, \mathbf{x})$

How to find $p_{bg}(m_{JJ}, \mathbf{x})$ for a localized signal?

3 different approaches:

- Find $p_{bg}(m_{JJ}, \mathbf{x})$ via simulation – accurate enough?
- Assume $p_{bg,SR}(m_{JJ}, \mathbf{x}) = p_{data,SB}(m_{JJ}, \mathbf{x})$ and train a **classifier** to distinguish between **data in the (narrow) sidebands** and the signal region (CWoLa) – not robust against correlations between m_{JJ} and $\mathbf{x}$.



CWoLA: E. M. Metodiev, B. Nachman, J. Thaler, 1708.02949; J.H. Collins, K. Howe, B. Nachman, 1805.02664 and 1902.02634

Probability densities and background

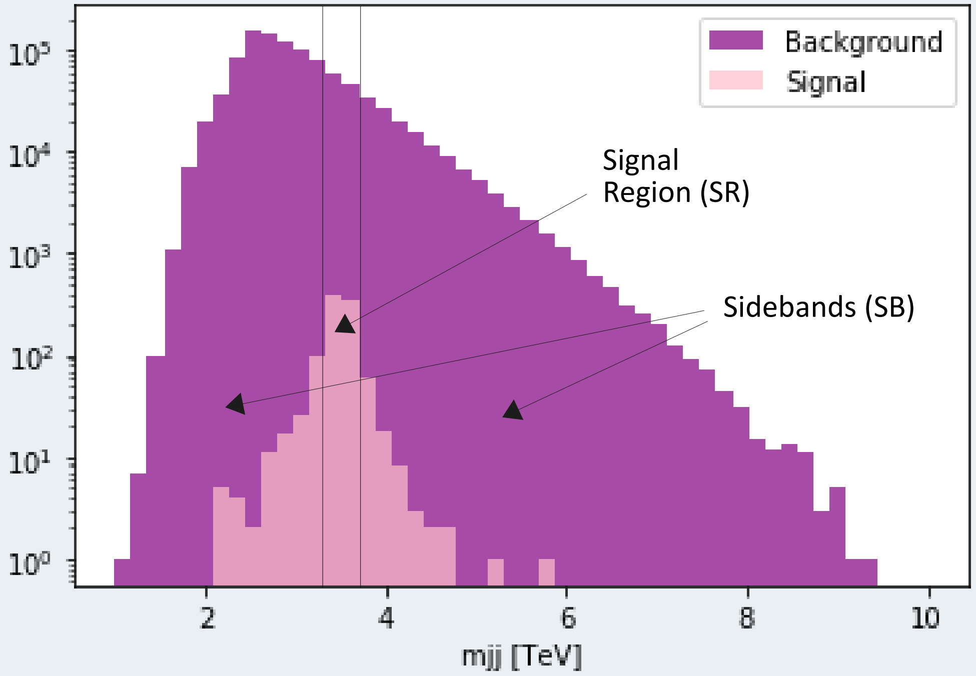
Assume we have a resonant variable m_{JJ}, and some other features $\mathbf{x}$.

$$p_{data}(m_{JJ}, \mathbf{x}) = \varepsilon\, p_{signal}(m_{JJ}, \mathbf{x}) + (1-\varepsilon) p_{background}(m_{JJ}, \mathbf{x})$$

How to find $p_{bg}(m_{JJ}, \mathbf{x})$ for a localized signal?

3 different approaches:

- Find $p_{bg}(m_{JJ}, \mathbf{x})$ via simulation – accurate enough?
- Assume $p_{bg,SR}(m_{JJ}, \mathbf{x}) = p_{data,SB}(m_{JJ}, \mathbf{x})$ and train a classifier to distinguish between data in the (narrow) sidebands and the signal region (CWoLa) – correlations?
- Train a **conditional density estimator** on $p_{data,SB}(\mathbf{x}|m_{JJ})$ and interpolate into the signal region. Separately train another density estimator on $p_{data,SR}(\mathbf{x}|m_{JJ})$ and **calculate the likelihood ratio*** (ANODE) – a much more difficult task than training a classifier, but more robust to correlations.



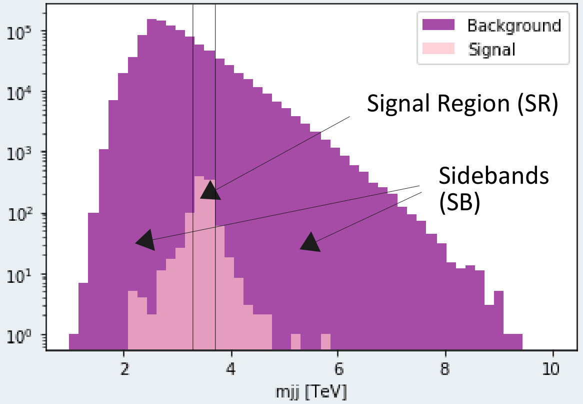
ANODE: B. Nachman, D. Shih 2001.04990 | *According to the Neyman-Pearson lemma, the likelihood ratio is the optimal binary classifier

The idea behind CATHODE

Combine the advantages of CWoLa and ANODE:

- Train a **conditional density estimator** on data in the **sidebands** and **interpolate into the signal region**. This protects against collapse due to correlations.

$$p_{data,SB}(\mathbf{x}|m_{JJ})$$

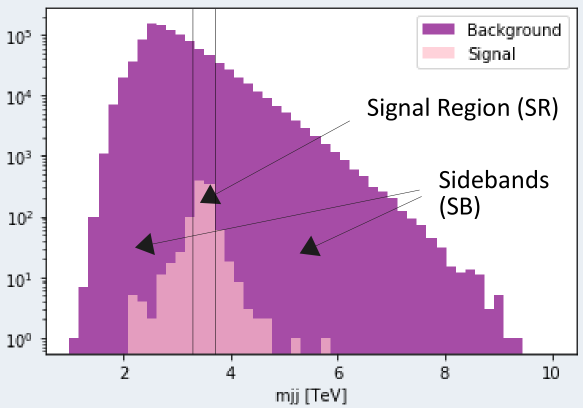


The idea behind CATHODE

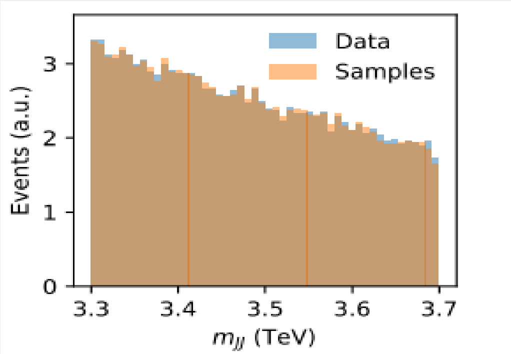
Combine the advantages of CWoLa and ANODE:

- Train a **conditional density estimator** on data in the **sidebands** and **interpolate into the signal region**. This protects against collapse due to correlations.
- **Generate samples** from the learned probability density, in the signal region. This is the background model.

$$p_{data,SB}(\mathbf{x}|m_{JJ})$$



$$p_{bg,SR}(m_{JJ},\mathbf{x})$$

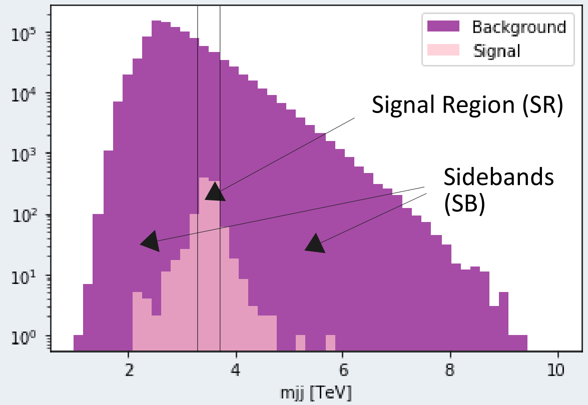


The idea behind CATHODE

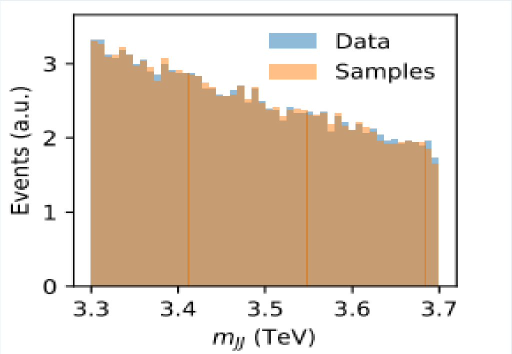
Combine the advantages of CWoLa and ANODE:

- Train a **conditional density estimator** on data in the **sidebands** and **interpolate into the signal region**. This protects against collapse due to correlations.
- **Generate samples** from the learned probability density, in the signal region. This is the background model.
- Train a **classifier** to **distinguish between data and samples** in the signal region. The combination of one density estimator and one classifier is an easier task than having to do two density estimations.

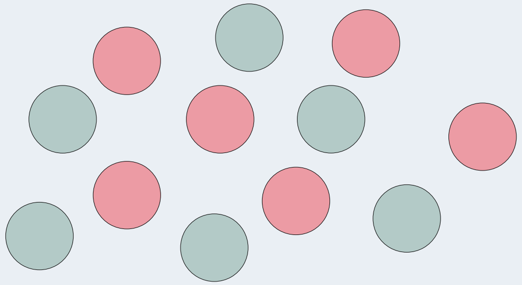
$p_{data,SB}(\mathbf{x}|m_{JJ})$



$p_{bg,SR}(m_{JJ},\mathbf{x})$



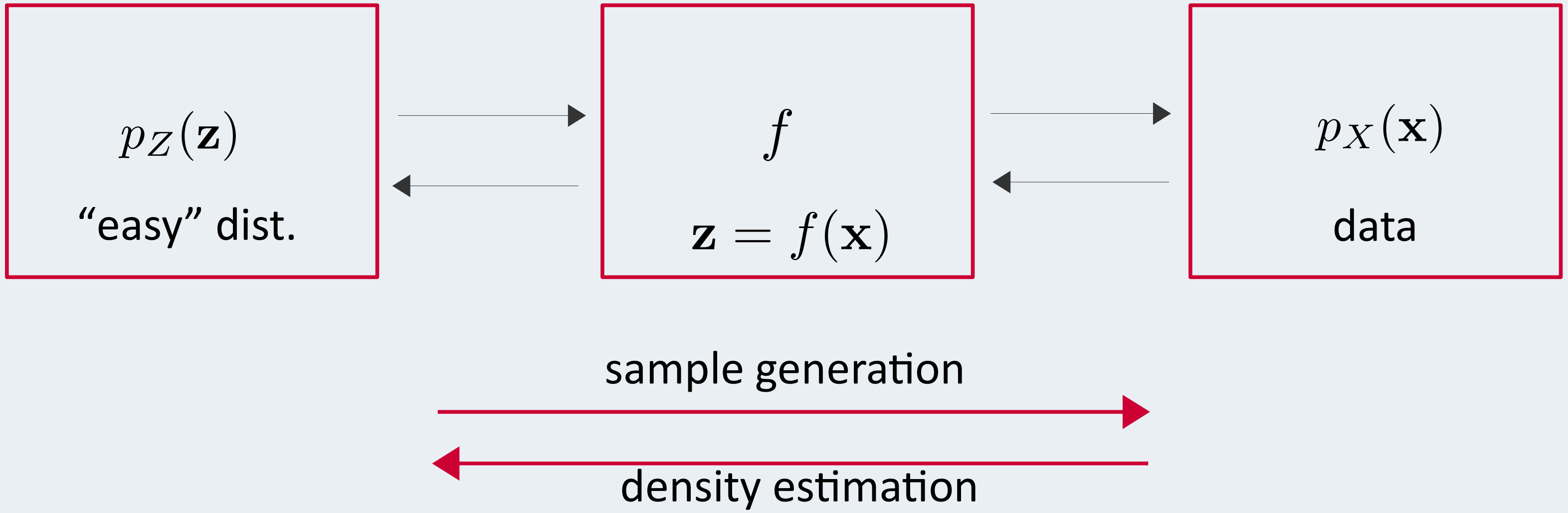
Classification



Quick intro to normalizing flows

The density estimation is performed using a Masked Autoregressive Flow (MAF), which is a type of normalizing flow.

Let f be a **bijective** map from a latent space with distribution $p_Z(\mathbf{z})$ to the feature space with distribution $p_X(\mathbf{x})$, such that $\mathbf{z} = f(\mathbf{x})$.



By the **change of variables** formula for random variables, $p_X(\mathbf{x}) = p_Z(\mathbf{z}) \left| \det \frac{f^{-1}(\mathbf{z})}{\partial \mathbf{z}} \right|^{-1}$

L. Dihn *et al* 1410.8516 | D. Jimenez Rezende *et al* 1505.05770 | M. Germain *et al* 1502.03509 | G. Papamakarios *et al* 1705.07057

Density estimation: Expressivity

A **chain of bijective maps** is also bijective: $f = f_1 \circ f_2 \circ ... \circ f_n$

In this way, we can use functions f_i that are easily invertible, while still obtaining **expressivity**.

f is not a **neural network**, since that wouldn't be invertible, but its **parameters** (eg. $\mu, \alpha, ...$) are. The parameters will be functions of z, such that $f(z) = f(\mu(z), \alpha(z), ...)$.

Density estimation: Jacobian

In general, a number of $\mathcal{O}(d^3)$ operations are needed to evaluate a d-dimensional Jacobian.

This is made tractable by turning it into a **triangular matrix**, which only requires $\mathcal{O}(d)$ operations for evaluation.

Use binary masks on the weights to ensure that **each output is conditioned only on the previous** outputs:
$\mu_i(z_1, ..., z_{i-1})$

This ensures the **autoregressive property** (→triangular Jacobian), and goes by the name **Masked Autoregressive Flow** (MAF).

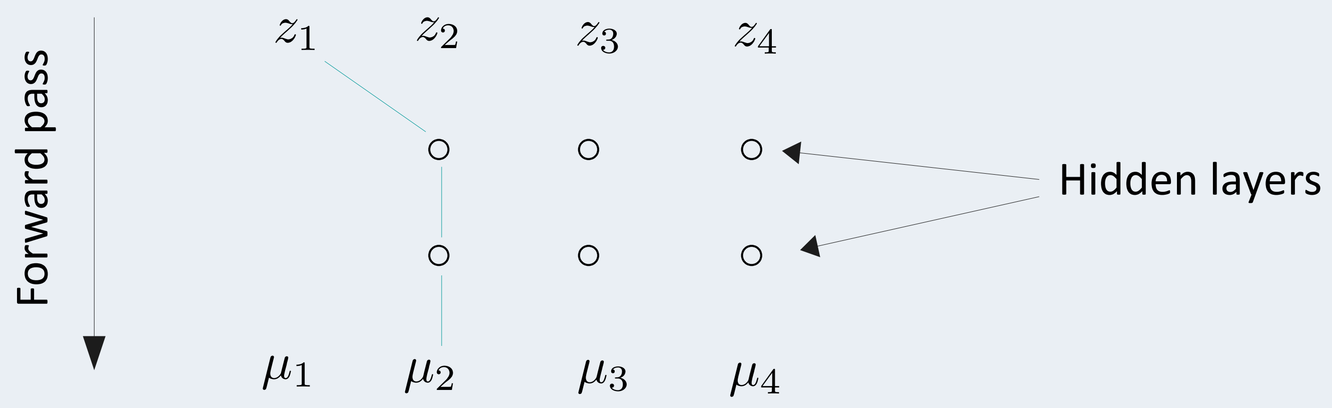
Density estimation: Masked Autoregressive Flow

In general, a number of $\mathcal{O}(d^3)$ operations are needed to evaluate a d-dimensional Jacobian.

This is made tractable by turning it into a **triangular matrix**, which only requires $\mathcal{O}(d)$ operations for evaluation.

Use binary masks on the weights to ensure that **each output is conditioned only on the previous** outputs: $\mu_i(z_1, ..., z_{i-1})$

This ensures the **autoregressive property** (→triangular Jacobian), and goes by the name **Masked Autoregressive Flow** (MAF).



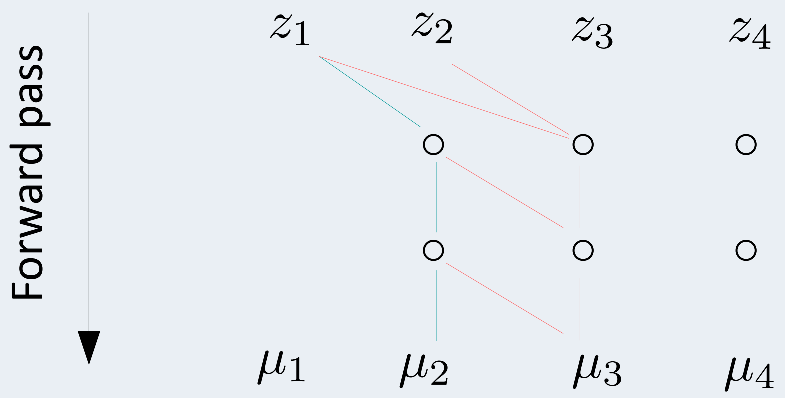
Density estimation: Masked Autoregressive Flow

In general, a number of $\mathcal{O}(d^3)$ operations are needed to evaluate a d-dimensional Jacobian.

This is made tractable by turning it into a **triangular matrix**, which only requires $\mathcal{O}(d)$ operations for evaluation.

Use binary masks on the weights to ensure that **each output is conditioned only on the previous** outputs: $\mu_i(z_1, ..., z_{i-1})$

This ensures the **autoregressive property** (→triangular Jacobian), and goes by the name **Masked Autoregressive Flow** (MAF).



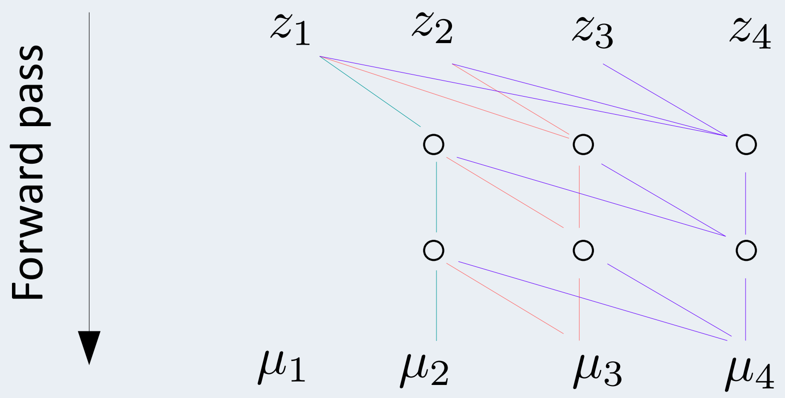
Density estimation: Masked Autoregressive Flow

In general, a number of $\mathcal{O}(d^3)$ operations are needed to evaluate a d-dimensional Jacobian.

This is made tractable by turning it into a **triangular matrix**, which only requires $\mathcal{O}(d)$ operations for evaluation.

Use binary masks on the weights to ensure that **each output is conditioned only on the previous** outputs:
$\mu_i(z_1, ..., z_{i-1})$

This ensures the **autoregressive property** (→triangular Jacobian), and goes by the name **Masked Autoregressive Flow** (MAF).



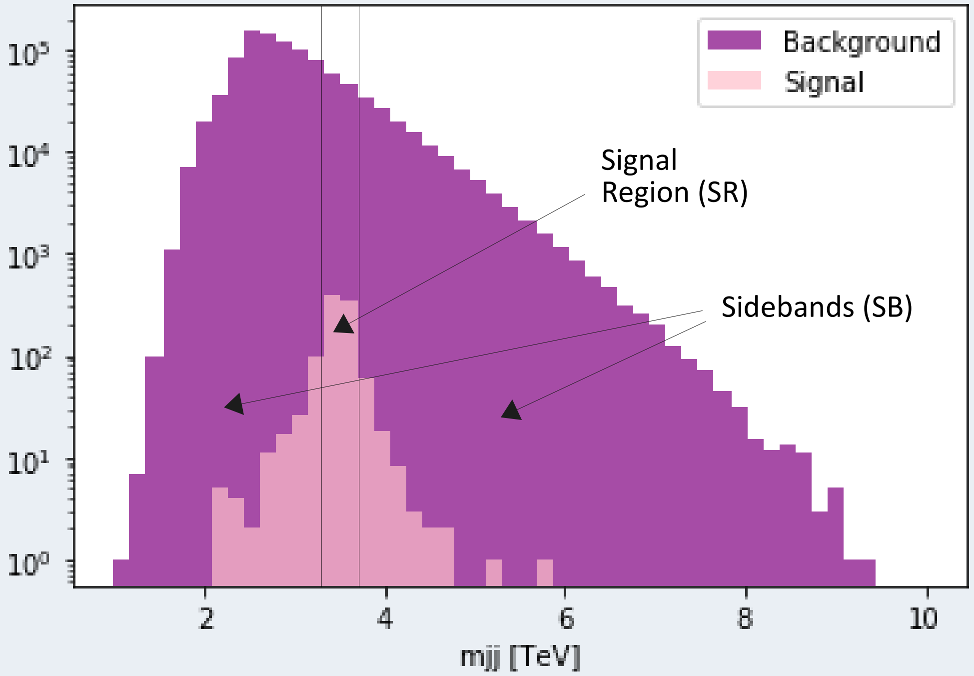
Note that we get all of this in a **single forward pass** through the network. The MAF is very fast at density estimation.

* For a conditional density estimator, add a feature z_5 which is not masked – all μ_i can depend on it

Dataset

From the LHC Olympics R&D dataset, we use:

- 1,000,000 QCD dijet events
- 1,000 $W' \to X(\to qq)Y(\to qq)$ events
- $m_{W'} = 3.5 \text{ TeV}, m_X = 500 \text{ GeV}, m_Y = 100 \text{ GeV}$
- In signal region, $3.3 \text{ TeV} < m_{JJ} < 3.7 \text{ TeV}$:
  - 121,352 background events
  - 772 signal events
- Initial $S/B = 6 \times 10^{-3}$, $S/\sqrt{B} = 2.2$



LHCO: G. Kasieczka *et al*, 2101.08320

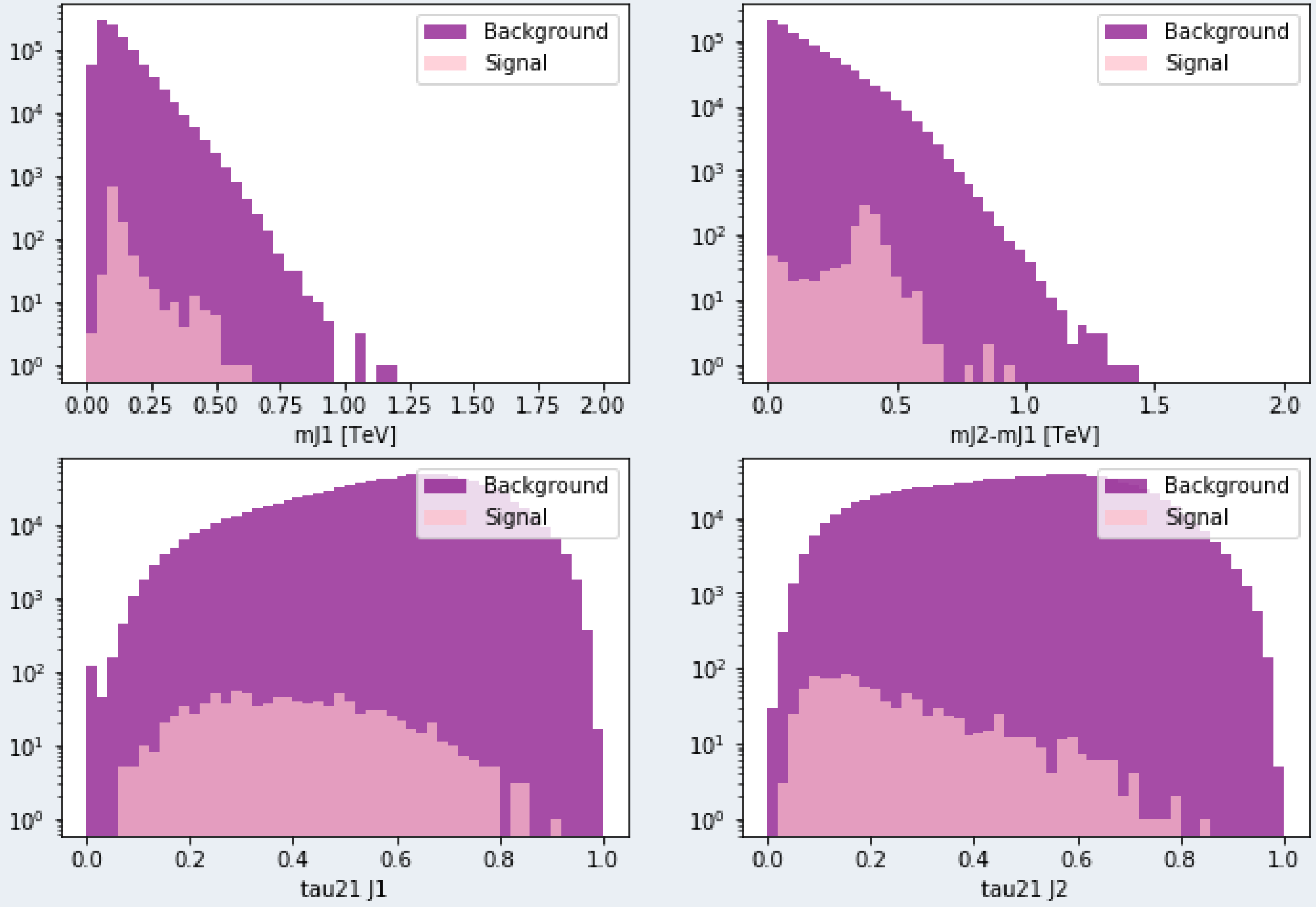
Dataset: features

Conditional feature:

- m_{JJ} – the total invariant mass of the two jets

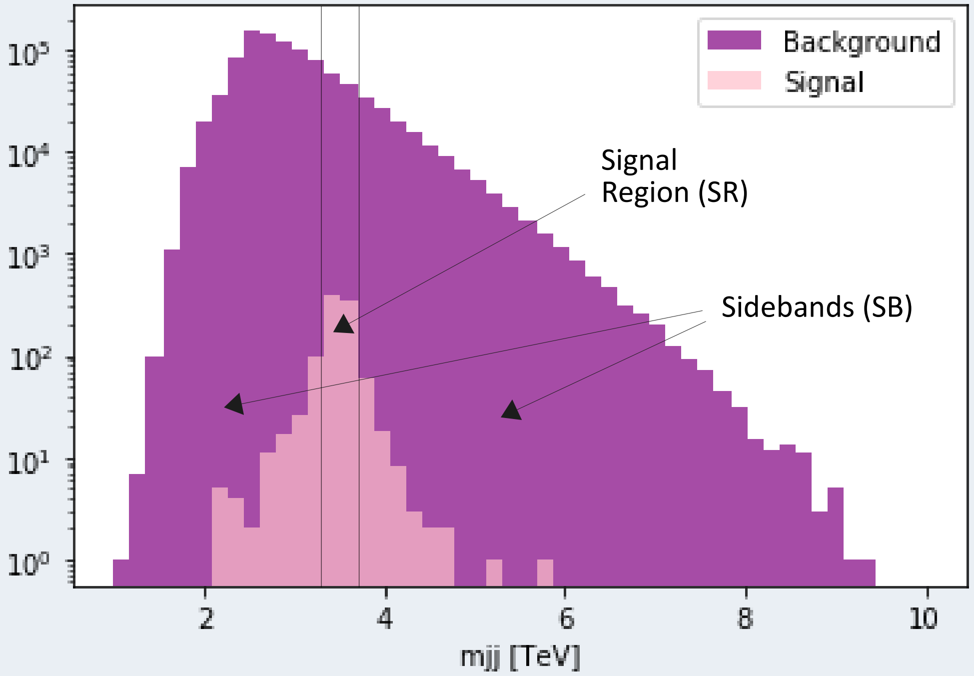
Auxiliary features:

- $m_{J1,J2}$ – the invariant masses of the individual jets
- $\tau^{21}_{J1,J2}$ – the n-subjettiness of the two jets



MAF training

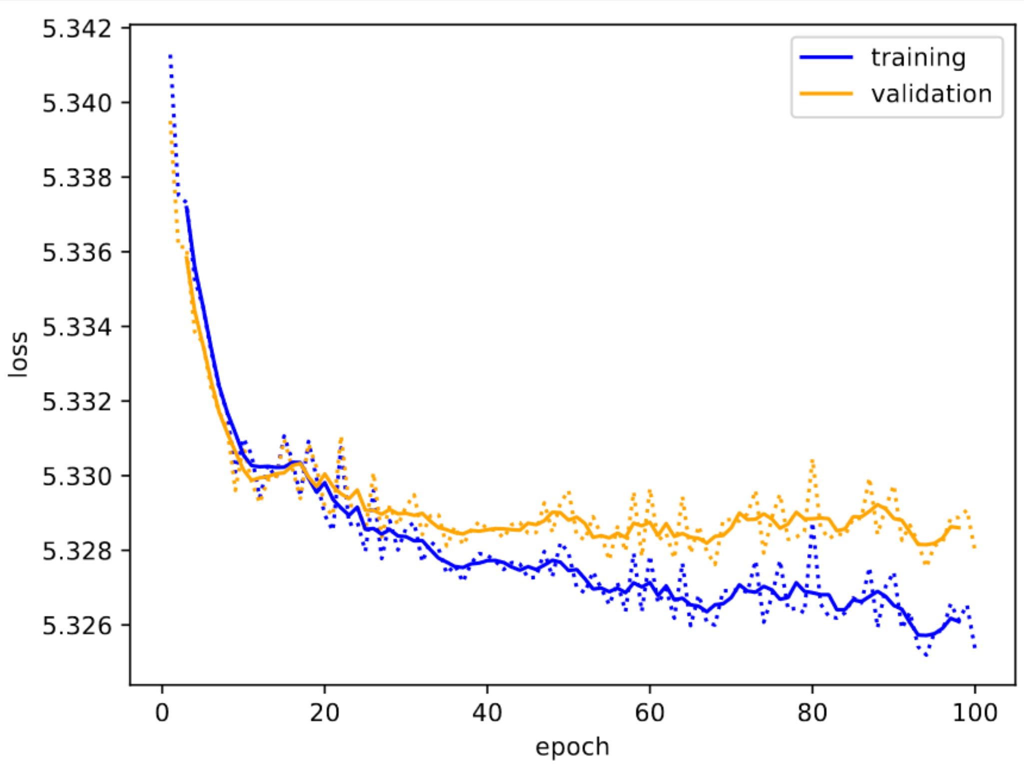
Train the MAF in the **sideband region** for 100 epochs.



MAF training: model selection

Train the MAF in the sideband region for 100 epochs.

Pick the 10 epochs with the **lowest validation loss**. We are going to ensemble these 10 models.



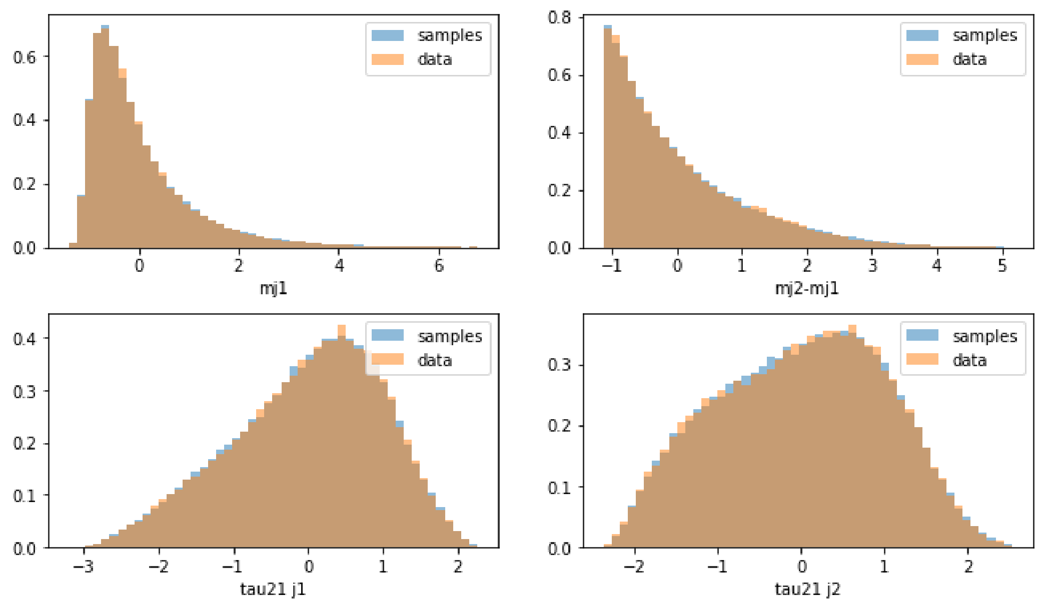
Sampling

Train the MAF in the sideband region for 100 epochs.

Pick the 10 epochs with the lowest validation loss. We are going to ensemble these 10 models.

Draw m_{JJ} values *in the signal region* using a KDE fit to data.

Use these to **sample*** an equal number of events from each of the chosen epochs, and **combine** to one single sample (ensembling).



Comparing samples (background model) to background in data

*We are using the MAF for sampling

Sampling

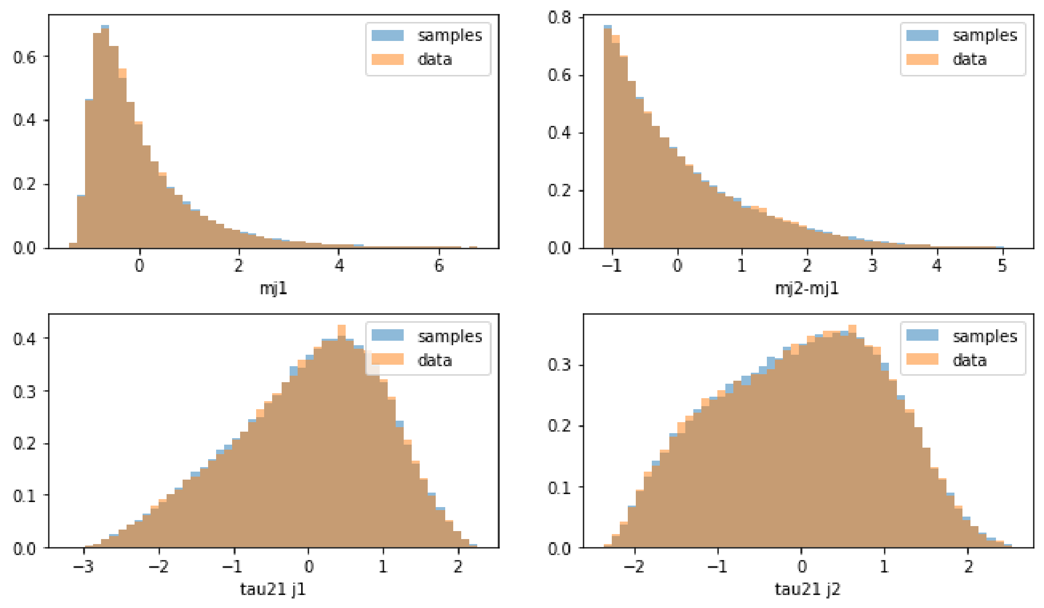
Train the MAF in the sideband region for 100 epochs.

Pick the 10 epochs with the lowest validation loss. We are going to ensemble these 10 models.

Draw m_{JJ} values *in the signal region* using a KDE fit to data.

Use these to **sample*** an equal number of events from each of the chosen epochs, and **combine** to one single sample (ensembling).

At this point we may choose to **oversample**, generate more samples than data, which as we will see improves the performance.



Comparing samples (background model) to background in data

Classification: getting the optimal anomaly detector

Train a classifier to distinguish between the samples we generated, and data in the signal region.

Train Keras with 3 hidden layers with 64 nodes each, ADAM as optimizer, for 100 epochs. Use class weights to re-balance the classes if oversampling has been used.

*The optimal classifier trained to distinguish between two mixed datasets (containing signal and background) is also the optimal classifier for distinguishing signal from background.

Classification: getting the optimal anomaly detector

Train a classifier to distinguish between the samples we generated, and data in the signal region.

Train Keras with 3 hidden layers with 64 nodes each, ADAM as optimizer, for 100 epochs. Use class weights to re-balance the classes if oversampling has been used.

Pick the 10 epochs with the **lowest validation loss**, then **average** the predictions for each data point.

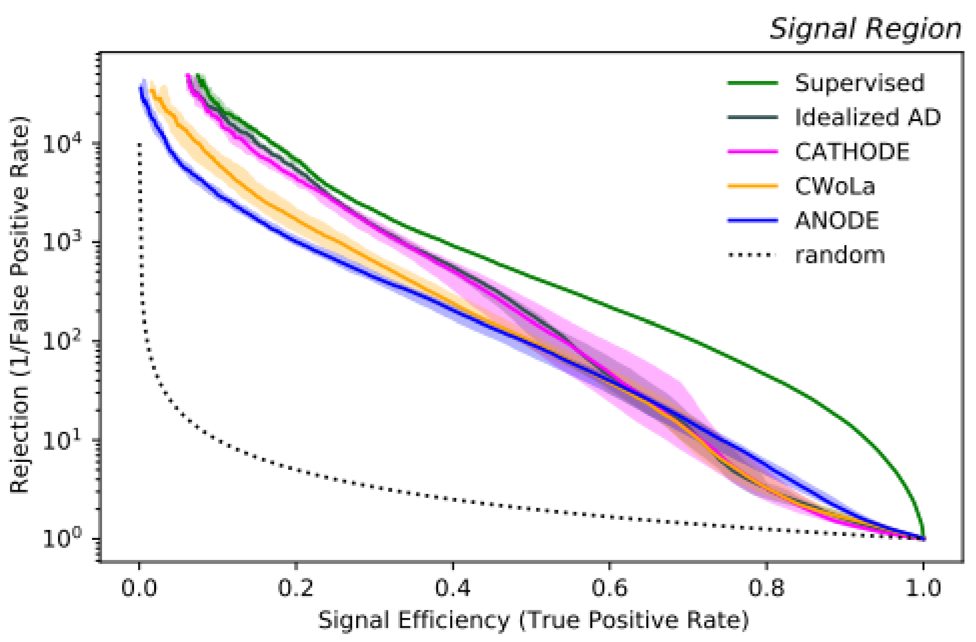
Calculate the true positive rate (TPR) and false positive rate (FPR) from the above average, and then the **significance improvement characteristic** ($\mathrm{SIC} = \mathrm{TPR}/\sqrt{\mathrm{FPR}}$).

Results

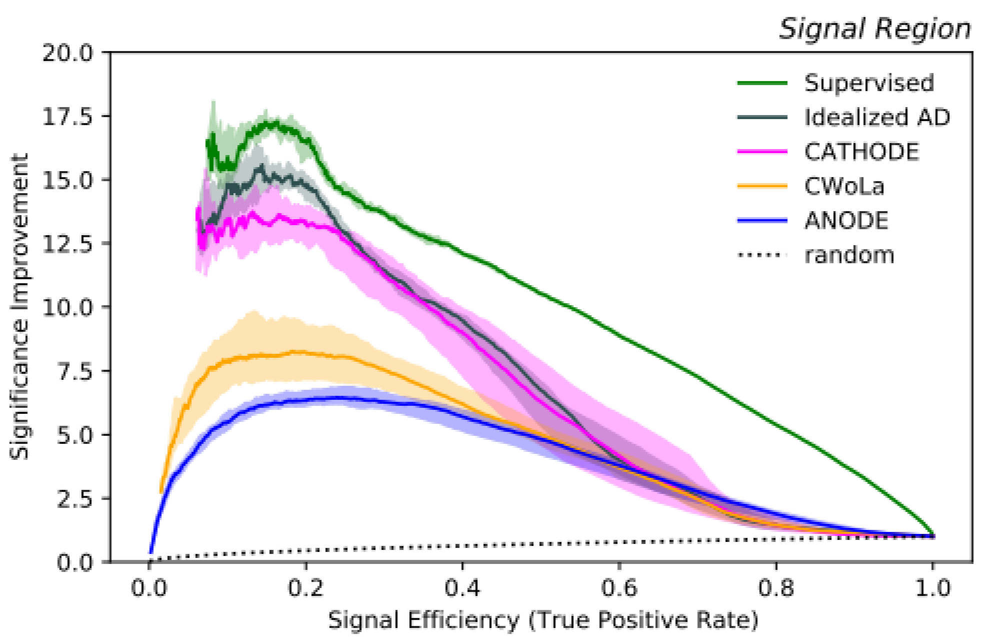
Initial $S/B = 6 \times 10^{-3}$, $S/\sqrt{B} = 2.2$

Curves are medians of 10 independent re-trainings; bands are 1 standard deviation.

ROC



SIC

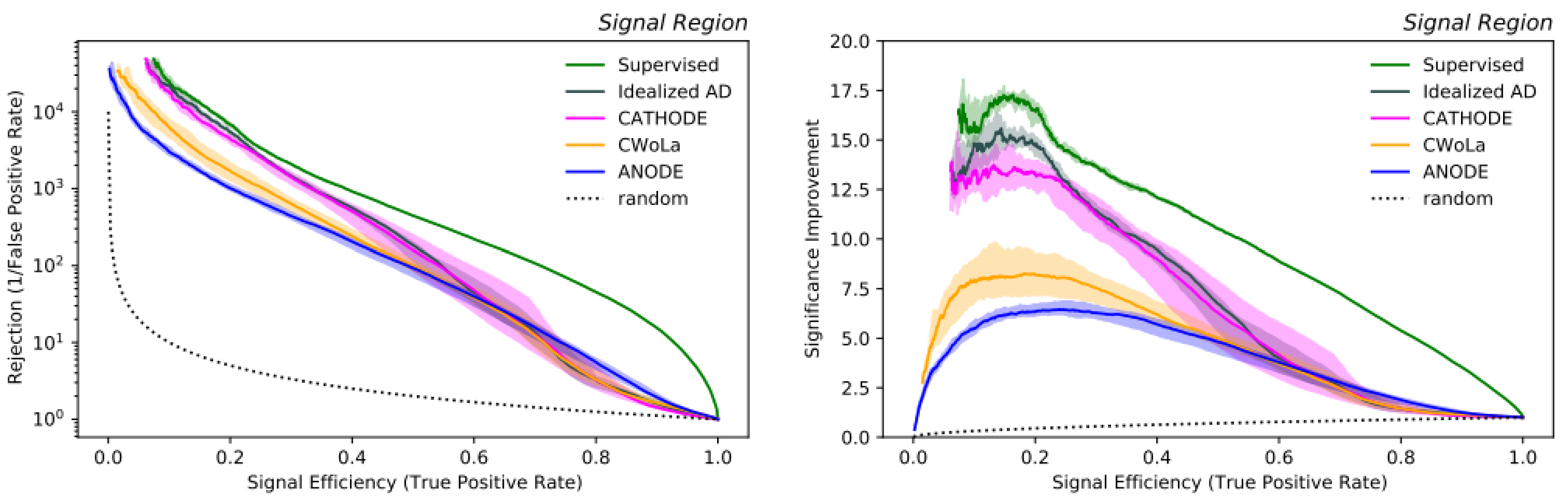


Results

Initial $S/B = 6 \times 10^{-3}$, $S/\sqrt{B} = 2.2$

Curves are medians of 10 independent re-trainings; bands are 1 standard deviation.

Significance improvement with CATHODE: up to **14** – approaches and even overlaps with idealized case.

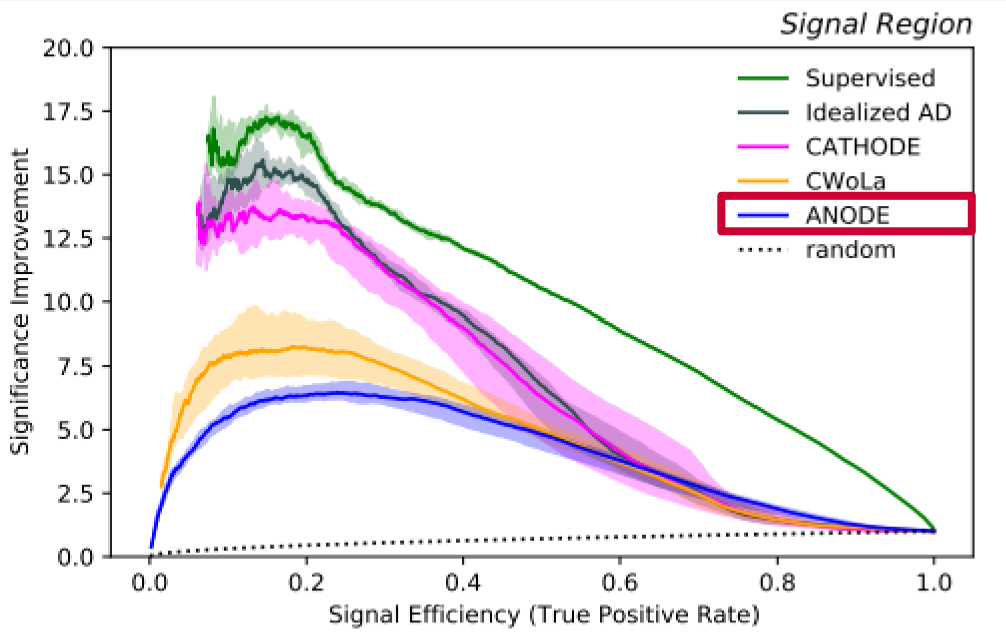


Results

Initial $S/B = 6 \times 10^{-3}$, $S/\sqrt{B} = 2.2$

Curves are medians of 10 independent re-trainings; bands are 1 standard deviation.

Significance improvement with CATHODE: up to **14** – approaches and even overlaps with idealized case.



Outperforms ANODE:

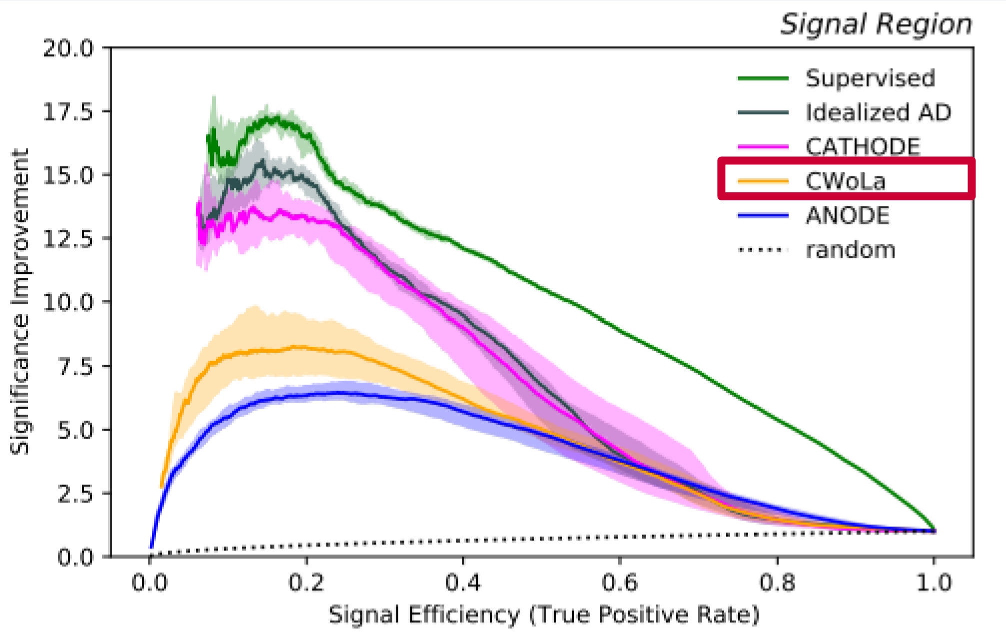
- CATHODE does not have to learn the density in the inner region, including the sharp peak where the signal is localized

Results

Initial $S/B = 6 \times 10^{-3}$, $S/\sqrt{B} = 2.2$

Curves are medians of 10 independent re-trainings; bands are 1 standard deviation.

Significance improvement with CATHODE: up to **14** – approaches and even overlaps with idealized case.



Outperforms CWoLa Hunting:

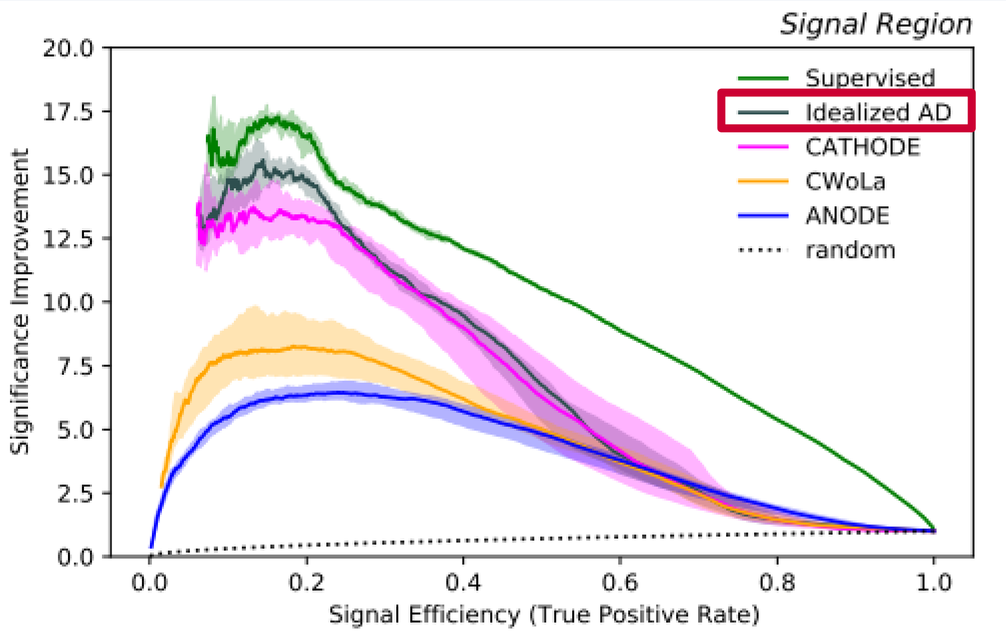
- There is a slight (percent level) correlation in the features
- CATHODE has the ability to oversample, giving the classifier more events to train on

Results

Initial $S/B = 6 \times 10^{-3}$, $S/\sqrt{B} = 2.2$

Curves are medians of 10 independent re-trainings; bands are 1 standard deviation.

Significance improvement with CATHODE: up to **14** – approaches and even overlaps with idealized case.



Comparison to the Idealized Anomaly Detector:

- The Idealized Anomaly Detector trains on “real” background instead of samples
- It is meant to provide an upper bound on the performance of any data vs background anomaly detection method
- CATHODE almost saturates the optimal performance on this dataset

Results: correlated data

Introduce artificial correlations between the features and the conditional variable:

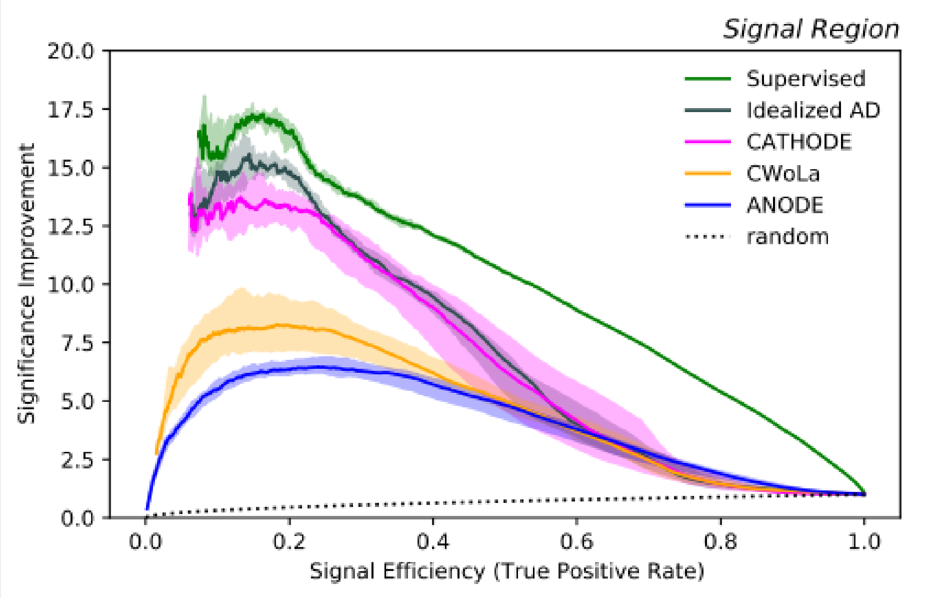
$m_{j1} \to m_{j1} + 0.1 m_{jj}$ $\Delta m_j \to \Delta m_j + 0.1 m_{jj}$

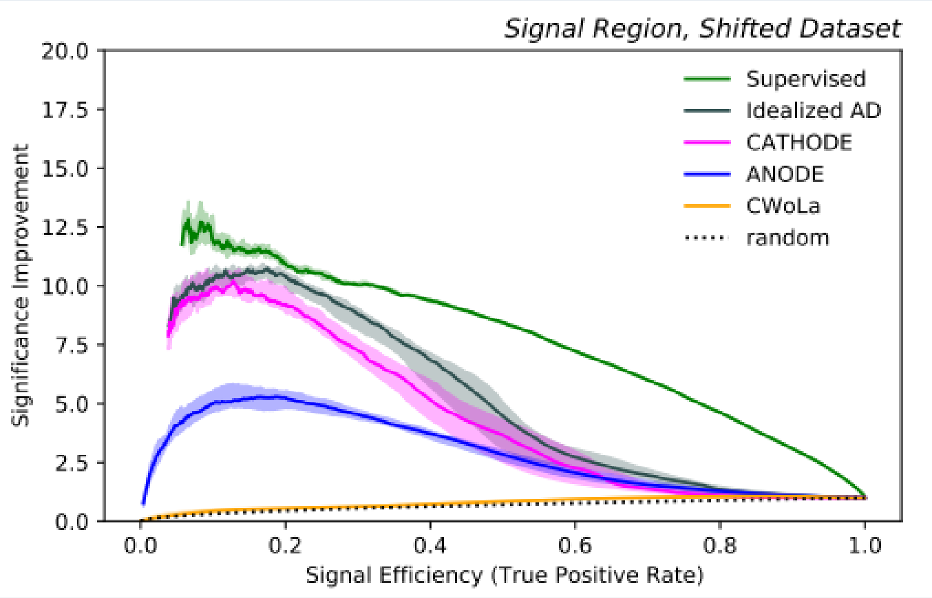
Results: correlated data

Introduce artificial correlations between the features and the conditional variable:

$m_{j1} \to m_{j1} + 0.1 m_{jj}$ $\quad \Delta m_j \to \Delta m_j + 0.1 m_{jj}$

The classifier learns to distinguish data from samples from m_{jj}, instead of learning the likelihood ratio. Since one of CWoLa's necessary conditions is absence of correlations, it breaks down in this test. Right: ratio shifted/regular.

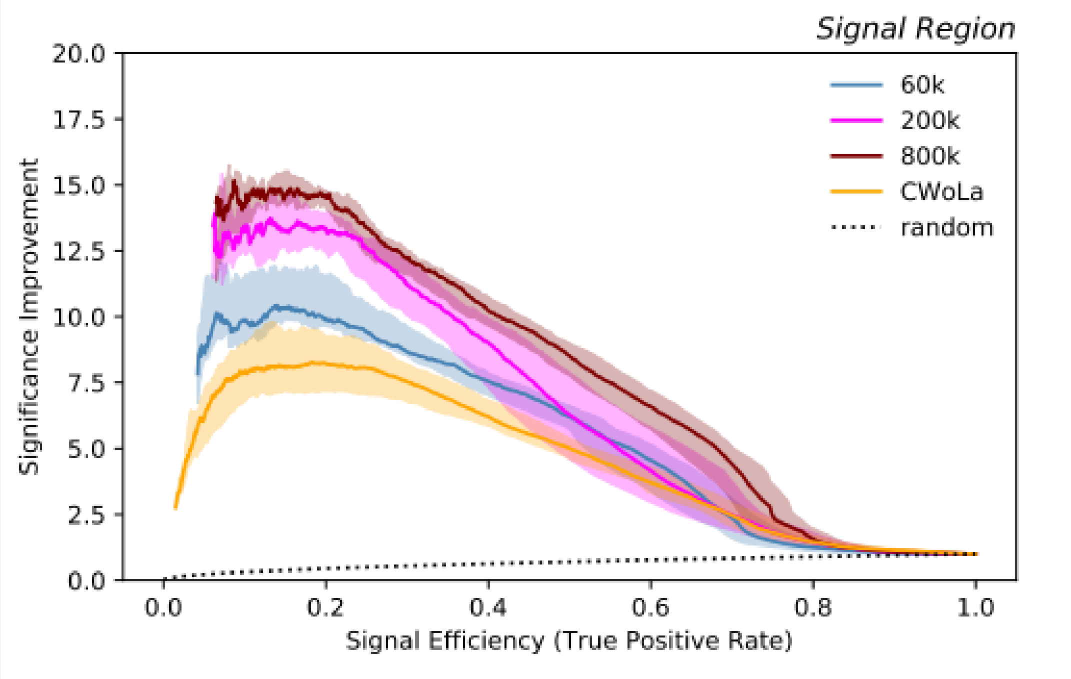




Results: oversampling

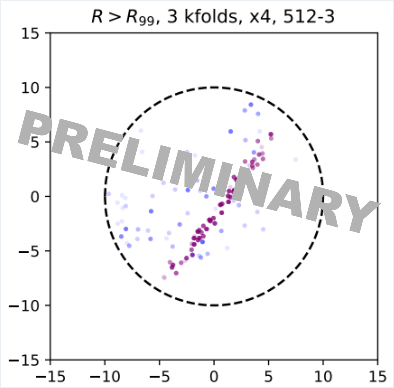
The total number of data events in the training set is fixed at 60,000 while the number of sampled events is varied. Plot legends specify the number of samples used in training.

Oversampling helps to a certain degree, as the classifier has more events to train on. Note that oversampling is not available for methods that rely only on the data.

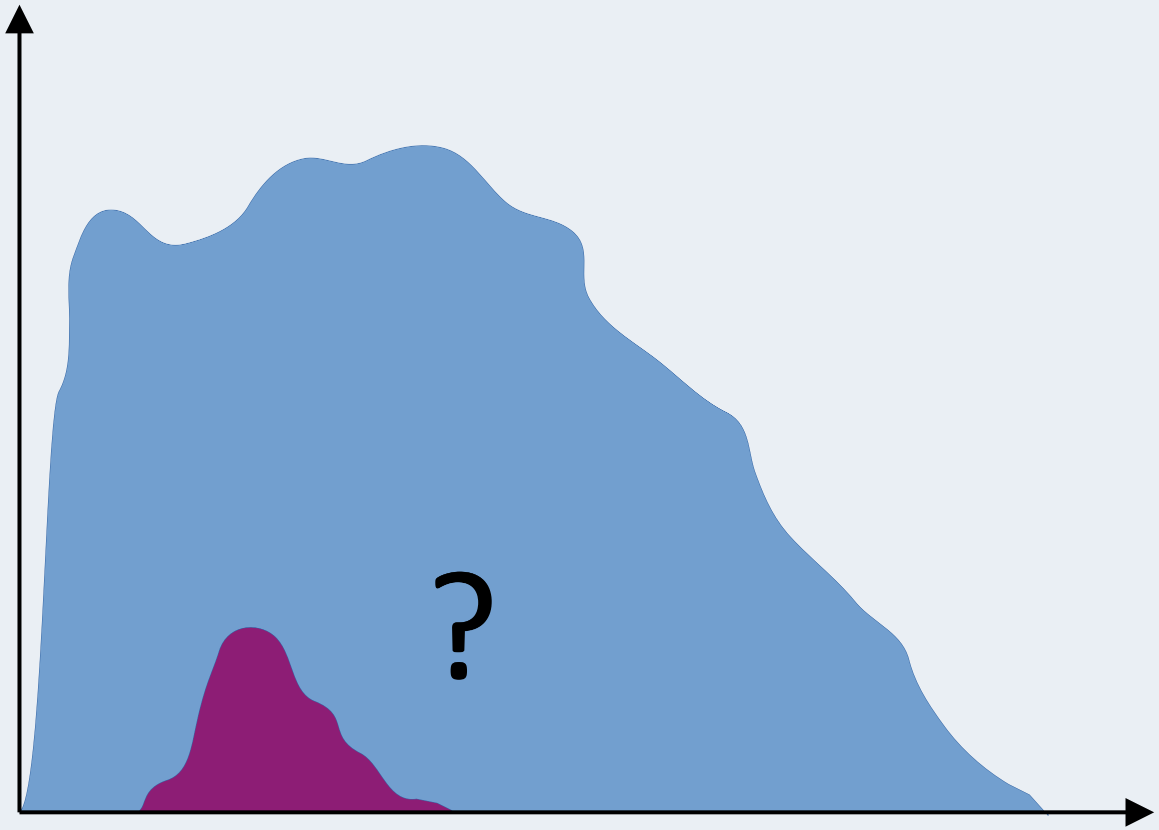


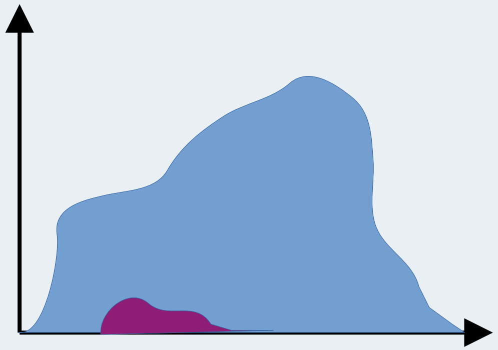
Conclusions

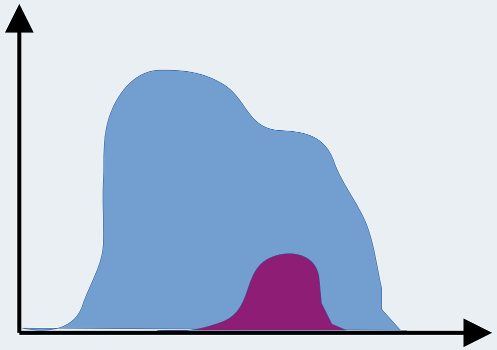
- We have presented **CATHODE**: a new **model agnostic** search strategy for **resonant new physics** at the LHC and beyond.
- CATHODE learns the background density by training a MAF in the sidebands and then interpolating to the signal region. This is a **data driven background estimation** that is less sensitive to correlations.
- The background model is generated through **sampling** in the signal region. By oversampling we can create as much background as we wish, which improves the performance.
- The final step of CATHODE is to train a **classifier** to distinguish between data and samples.
- CATHODE can reach a **significance improvement** of up to **14**, and nearly saturates the optimal performance on this dataset.
- Further work and future directions
  - Other datasets (very strong results so far) →
  - More or other auxiliary features
  - Other density estimators (work in progress)



Have other interesting datasets? Come talk to me!



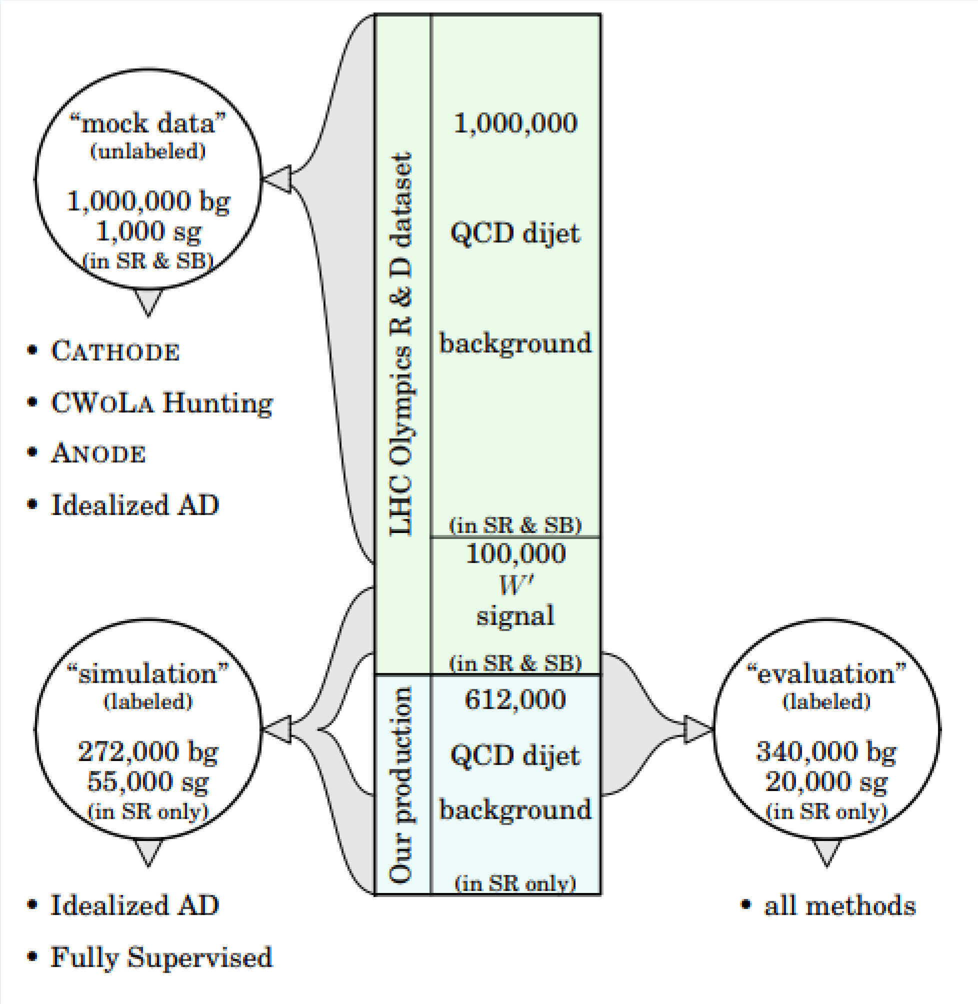




BACKUP

Number of events used

We generated an additional 612,000 QCD dijet events specifically in the SR. Of these, 340,000 were used in evaluation, and 272,000 were used in the simulation-based methods.



Number of events used

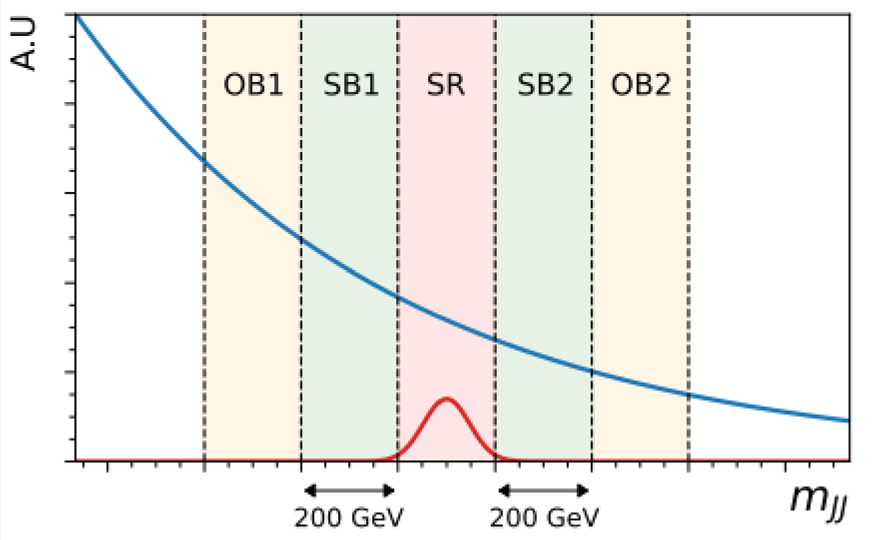
Events used in different methods

Method	Type	Train	Validation (model selection)	Evaluation
CATHODE	density estimator	500k SB data	380k SB data	340k SR background 20k SR signal
	classifier	200k SR background samples 60k SR data	200k SR background samples 60k SR data	
ANODE	density estimator	500k SB data 60k SR data	380k SB data 60k SR data	
CWoLA Hunting	classifier	65k SSB data 60k SR data	65k SSB data 60k SR data	
Idealized AD	classifier	136k SR background 60k SR data	136k SR background 60k SR data	
Fully Supervised	classifier	136k SR background 27k SR signal	136k SR background 27k SR signal	

CURTAINS comparison

arXiv:2203.09470 (Mar 2022); John Andrew Raine, Samuel Klein, Debajyoti Sengupta, Tobias Golling

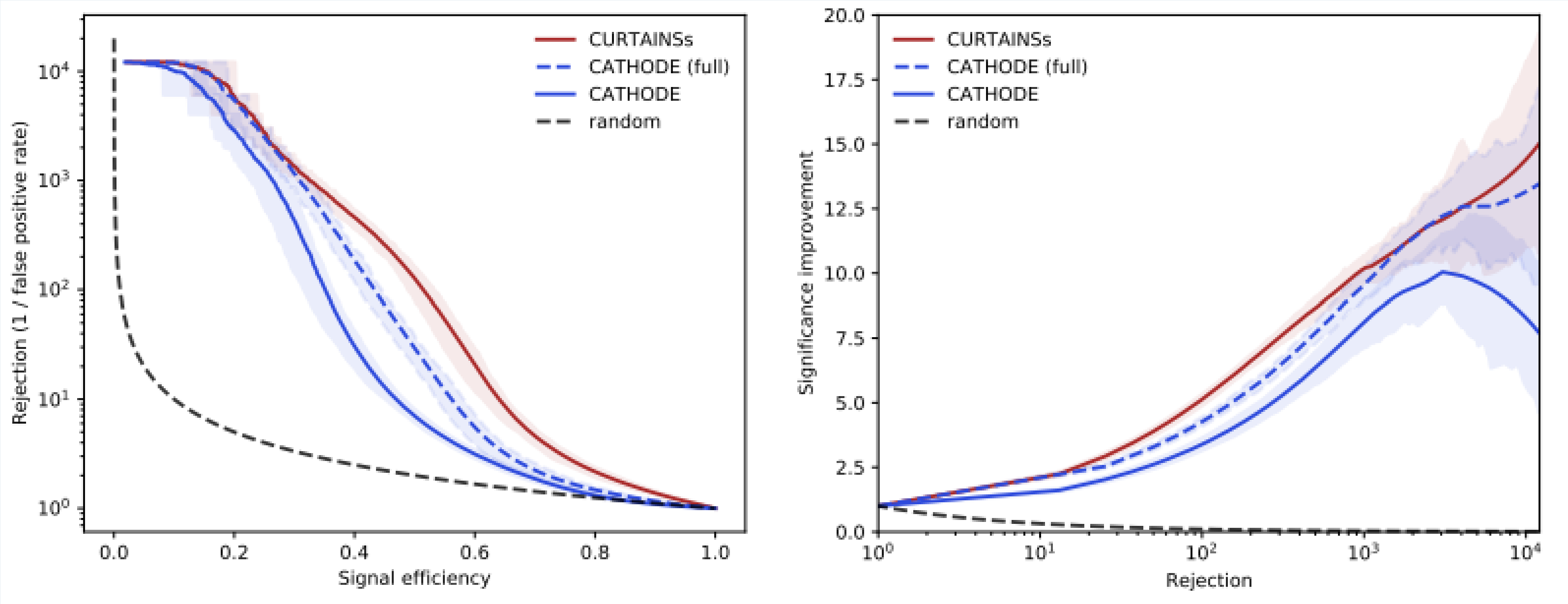
Instead of learning the density in the full sideband region, CURTAINS learns a transformation between two narrow sidebands. The background in the signal region is then estimated by transforming values from both sidebands into the signal region.



CURTAINS comparison

arXiv:2203.09470 (Mar 2022); John Andrew Raine, Samuel Klein, Debajyoti Sengupta, Tobias Golling

The CATHODE we have presented here is what is called “CATHODE full”* in these plots. We see that the performance of CURTAINS overlap with CATHODE in the relevant (lower) signal efficiency range.



* What Raine et al. call “CATHODE” is using only a narrow sideband (as for CWoLa) instead of the full distribution for background estimation.