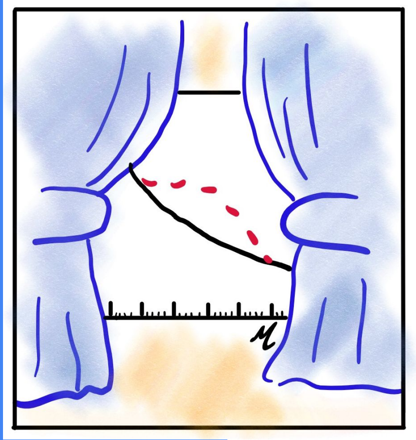
CURTAINs

Constructing Unobserved Regions by Transforming Adjacent Intervals

ML4Jets, 3rd November 2022

Johnny Raine, Sam Klein, Debajyoti Sengupta, Tobias Golling

University of Geneva



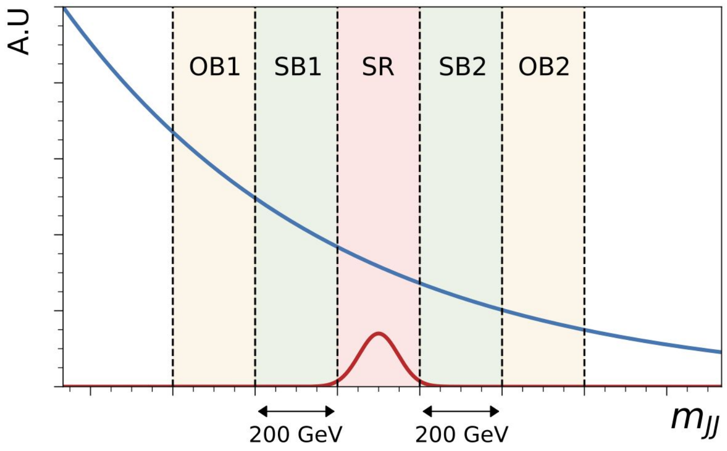
Bump hunts

We expect signal events to be localised in the invariant mass.

⇒ Show up as a **bump** in the spectrum

Method:

1. Split spectrum into sliding ‘side bands’
2. Fit the distribution in sidebands
3. Interpolate into the signal region
4. Look for an excess
5. Slide window and repeat



Extending Bump hunts

But what if the bump is dominated by background…

Is there more information we can use than just mass?

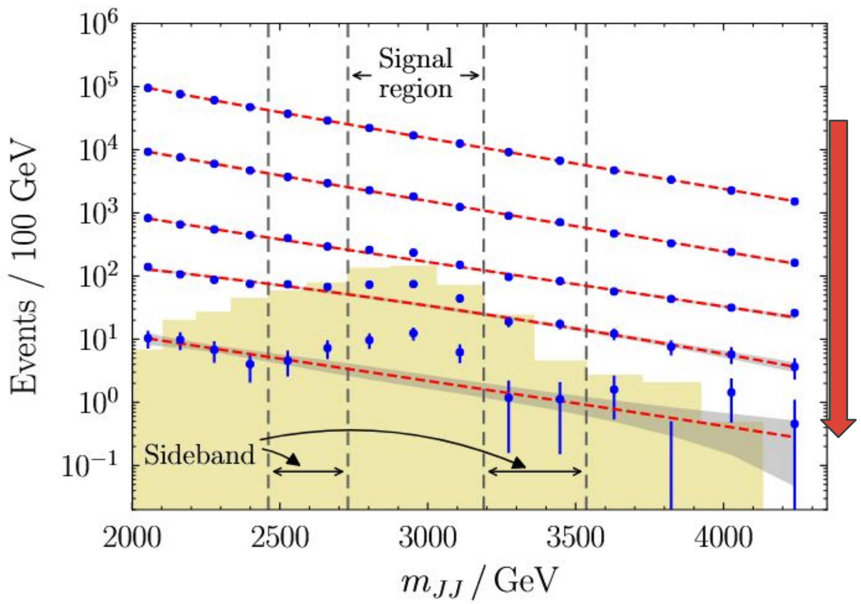
High mass - boosted decays
Look at substructure!

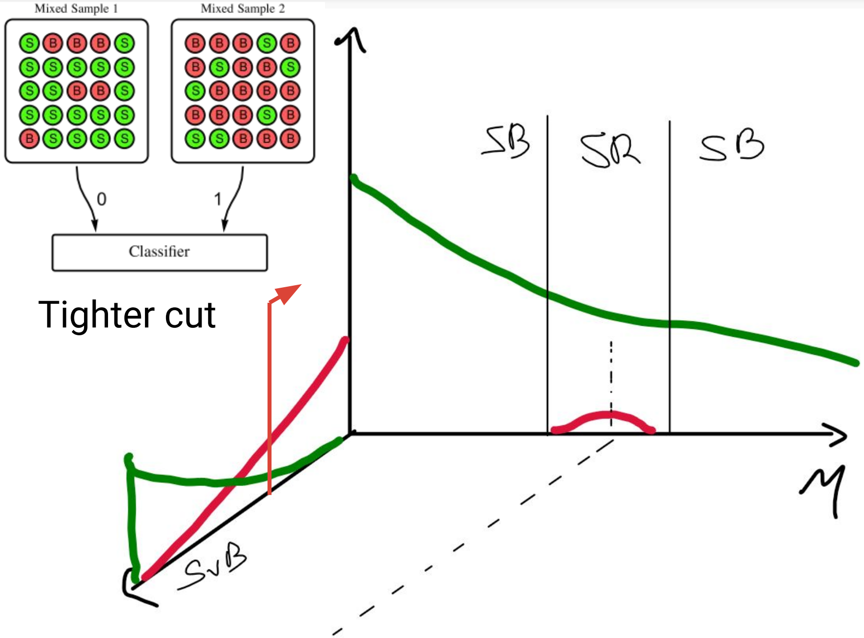
What do we train on?

Control region data
vs
Signal region

Noisy labels - weakly supervised!

Extending Bump hunts with CWoLa





Collins et al Metodiev et al

Extending Bump hunts

Works really well unless observables in classifier are ***correlated with Mass!***

How to take into account?

- Optimise choice of observables
- Bring in additional ML approaches for producing the background

Extending Bump hunts

Approaches showing great improvements over standard CWoLa bump hunt

ANODE (Nachman & Shih)

- Direct density estimation with normalizing flows, using base density for anomaly detection

CATHODE (Hallin et al)

- Normalizing flows trained outside of signal window, generate background data in signal window

SALAD (Andreassen et al) - Not using normalizing flows but also very good performance

- Use simulation to transfer classifier to data with density ratio estimation

Extending Bump hunts

Other approaches build on the idea and show great improvements over standard CWoLa bump hunt

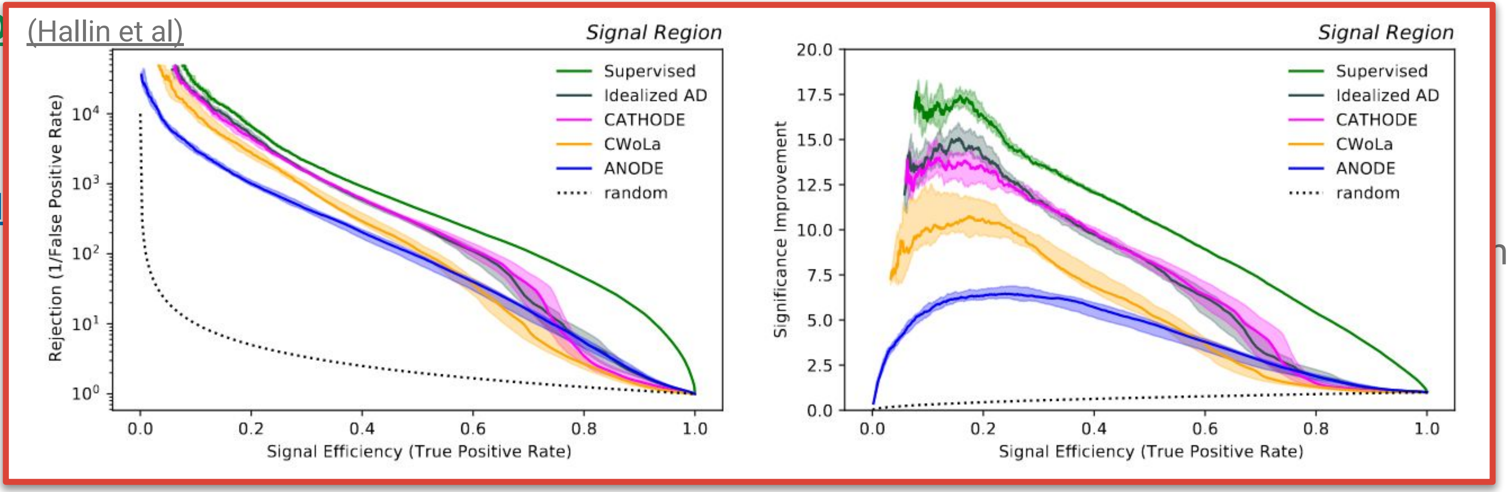
ANOD

●

CATH

● ndow

●



Introducing CURTAINs

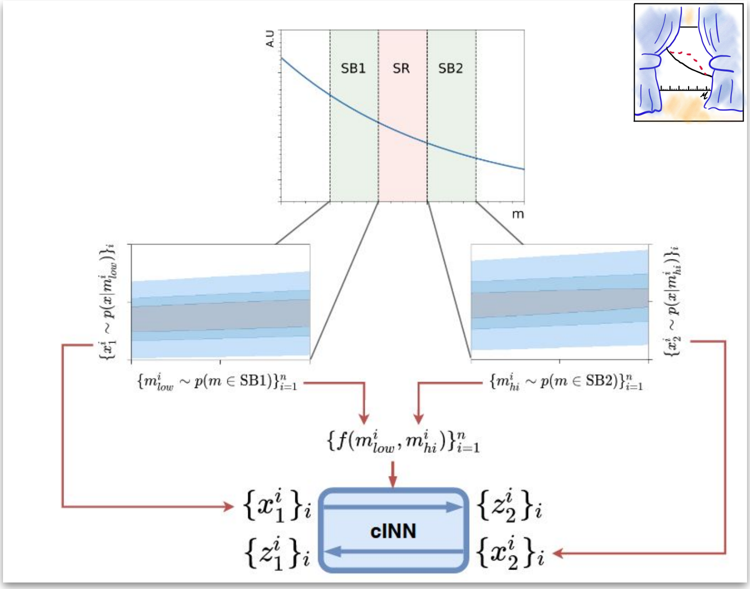
"What would this datapoint look like if it had a different value of mass?"

Train an INN to map data between sidebands

- Condition on the input and target mass
- Learn to account for changing mass

Once trained by choosing target mass values
Transport sideband data into signal region!

- No need to sample from base distribution
- Only estimate mass distribution in signal region



Training CURTAINs

Draw data *x* from SB1 and SB2

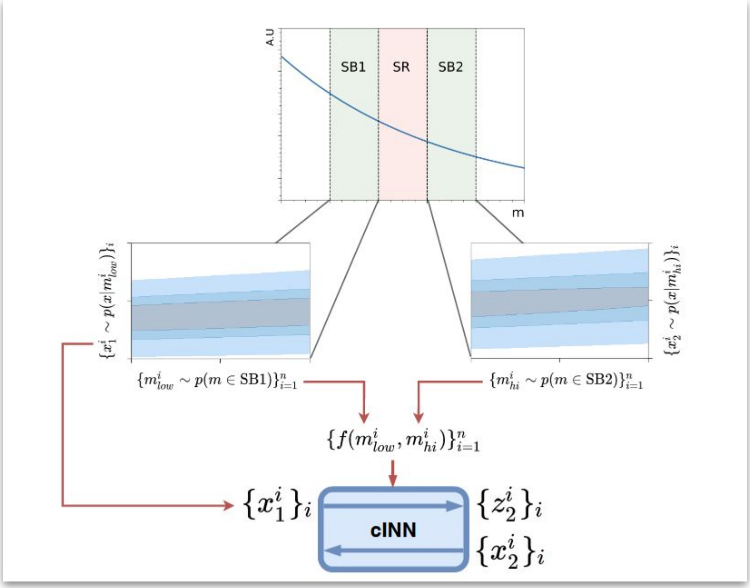
Assign target masses based on batch

Transport data from SB1<->SB2

- Can **train bidirectionally!**

Use an Optimal Transport loss to measure difference between z_2 and x_2

Compares distributions not datapoints



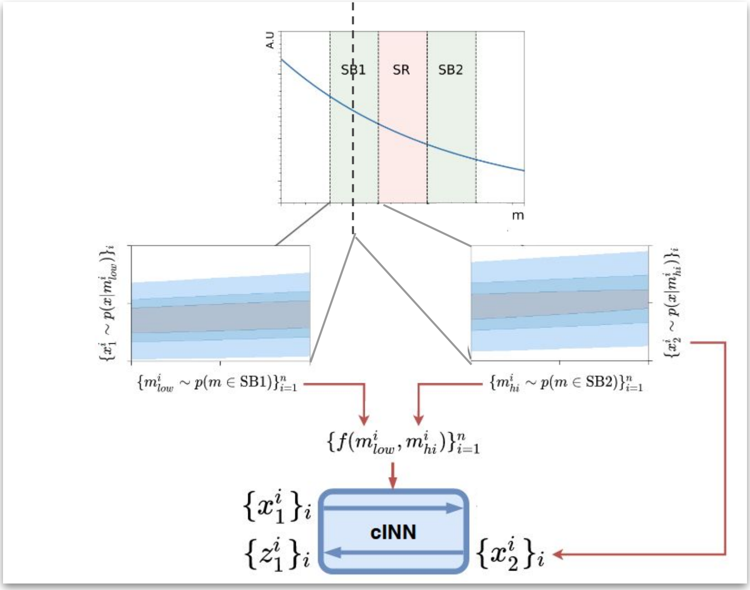
Training CURTAINs - technicality

Using f(m,m')=m'-m

- During training **min value is width of SR**
- To transport data from SB->SR **min value is 0**
- Outside of training domain…

Solution: Split sidebands into two

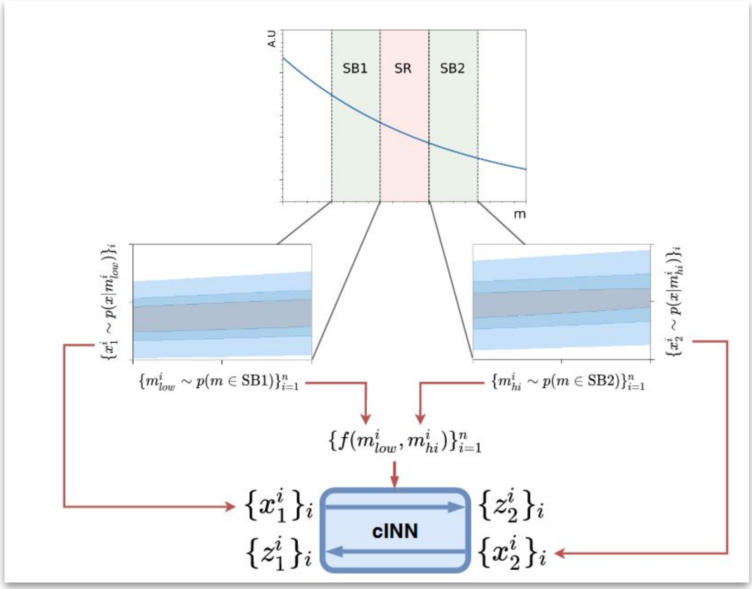
- Train between lower and upper half of SB1/2
- Now min value is also 0



Training CURTAINs - technicality

Using Sinkhorn loss to compare *x* to *z*

- Difference in distribution over events
- Minimised if distributions and correlations correct after transport
- **Does not** strictly enforce correct mass conditioning for each data point
- Slow convergence due to stochastic sampling of target batch



Not ideal but **empirically works**

Stochasticity of data - cannot get true example of a data point at another M value, look for minimum distance in x/z space and ignores M

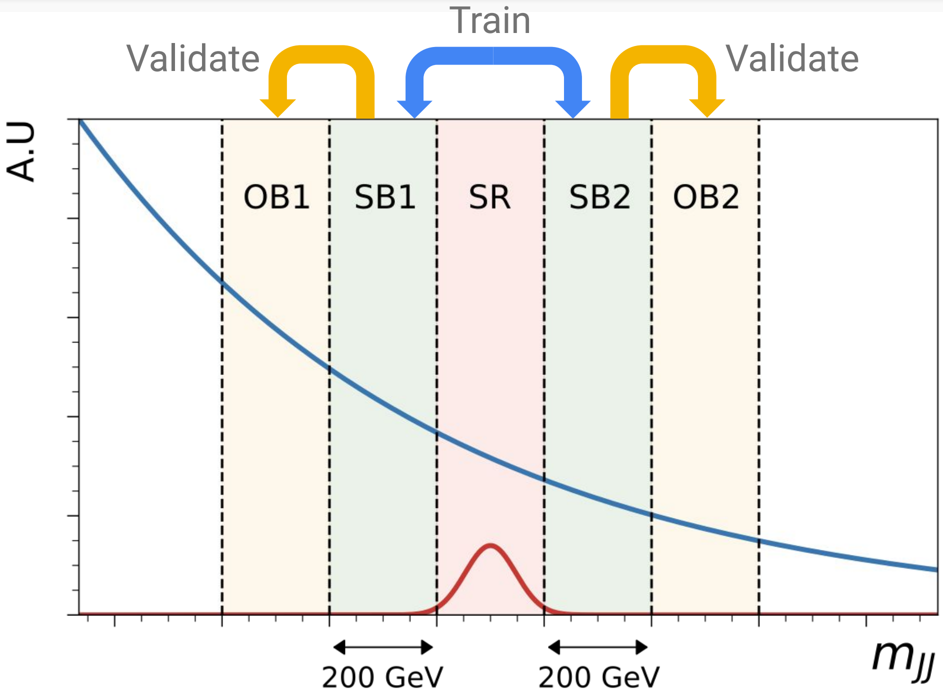
CURTAINs Validation

Fix sidebands

Define Outer-Band (OB) validation regions

Train CURTAINs transformer

Validate on OBs



CURTAINs - Training regions

Training on the LHCO R&D anomaly detection dataset

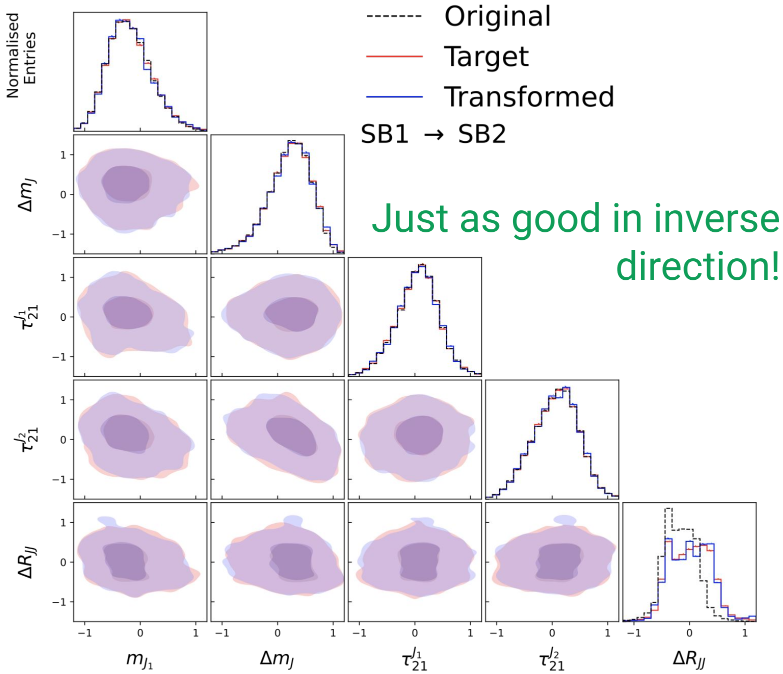
Sideband 1: [3200, 3400]

Sideband 2: [3600, 3800]

Five observables

M_{J1}, M_{J1} - M_{J2}, $\tau^{21}{}_{J1}$, $\tau^{21}{}_{J2}$

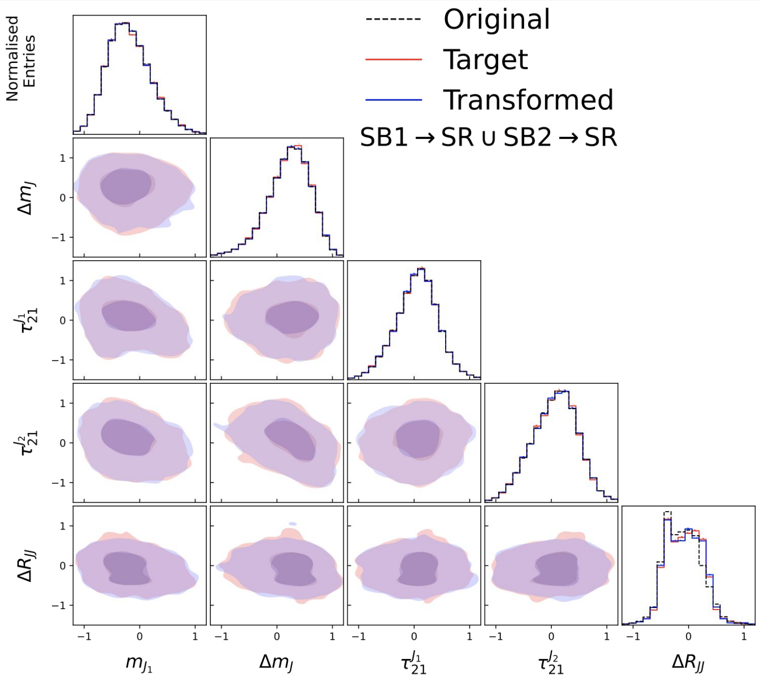
Plus ΔR_{JJ} due to correlation to Mjj



Just as good in inverse direction!

CURTAINs - Signal region

Nearly perfect matching implies near perfect background template!



**Can't look at this in the real analysis or application!*

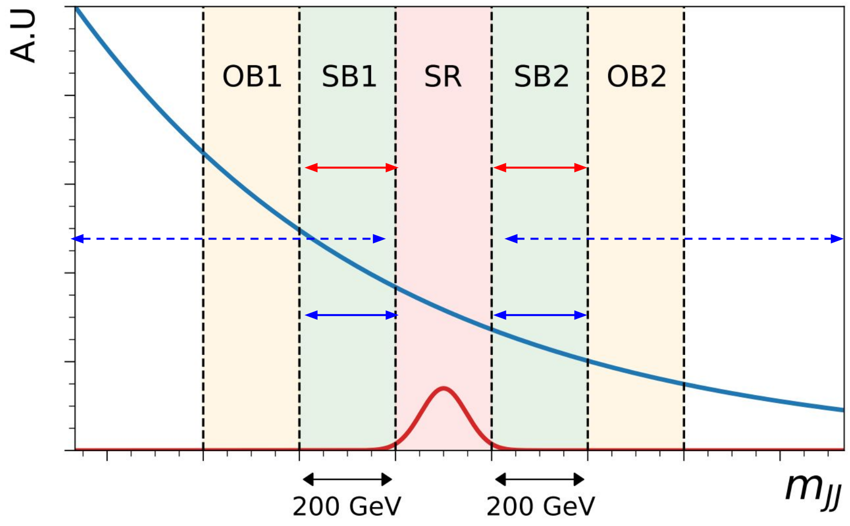
CURTAINs - CWoLa Performance

Compare to CATHODE method

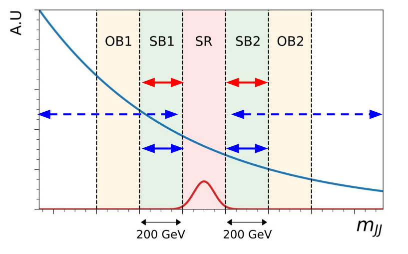
- Equivalent training window (local)
- All available data outside of SR (full)

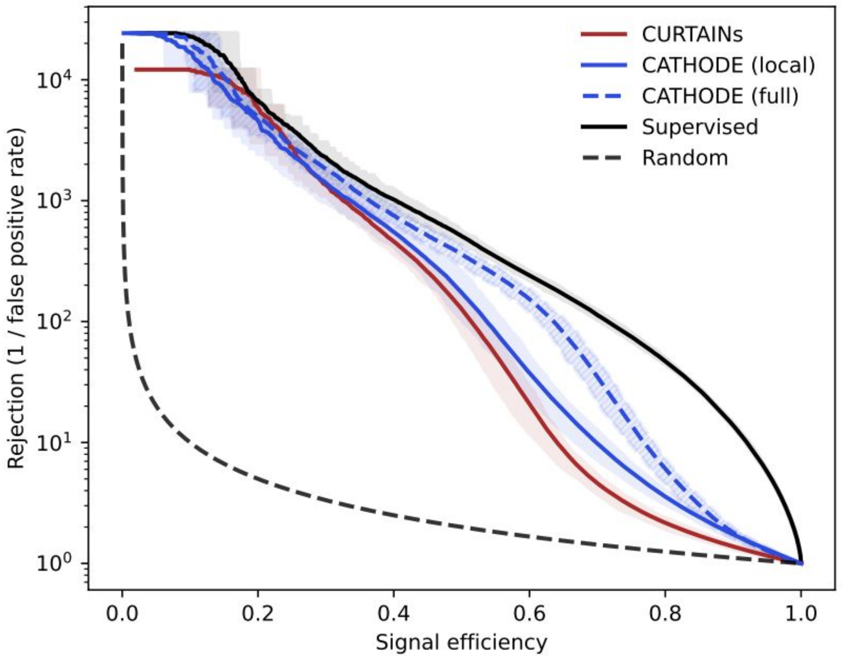
Same number of generated bkg for both methods

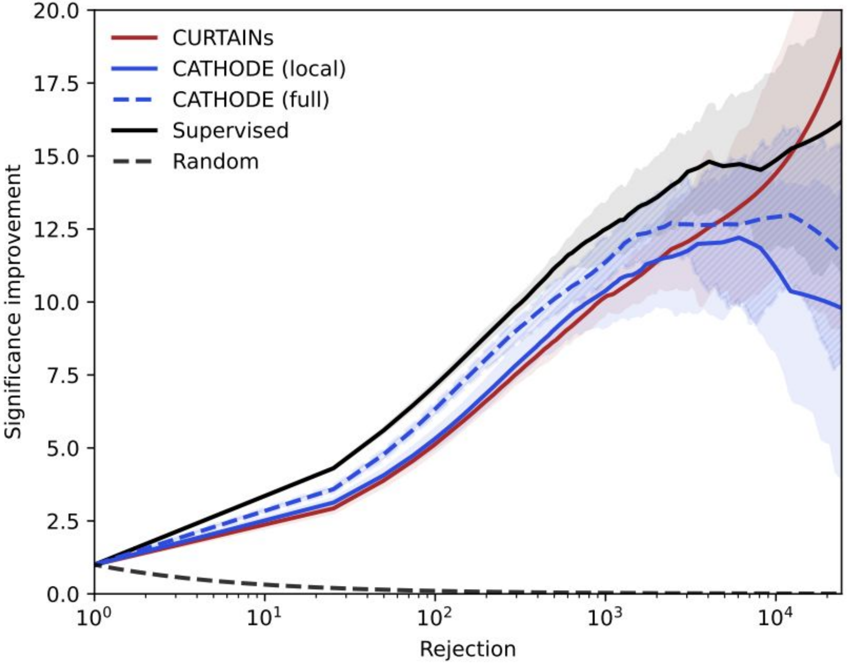
- “Oversample” CURTAINs by transporting same data to multiple values of *m*



CURTAINs - CWoLa Performance

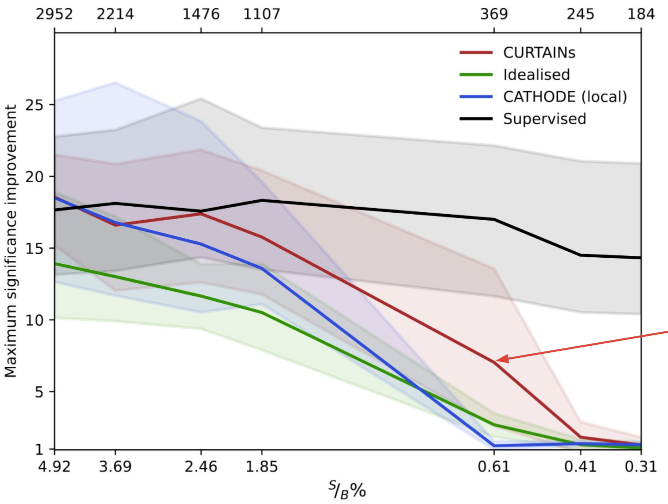


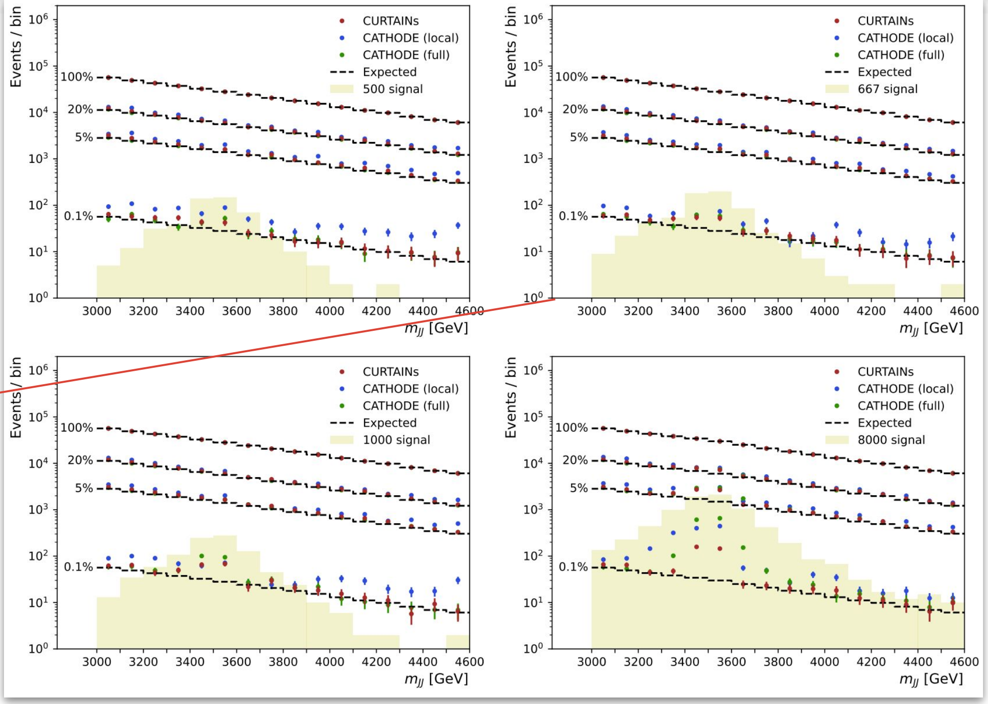




Comparing to full window training CATHODE performs better, but hard to disentangle impact from extra training statistics vs gain to method from extra Bkg to Sig ratio

CURTAINs - Bump hunt





CURTAINs
Flows4Flows

Flows4Flows

Use normalizing flows to parametrise base distribution! No more approximate OT loss!

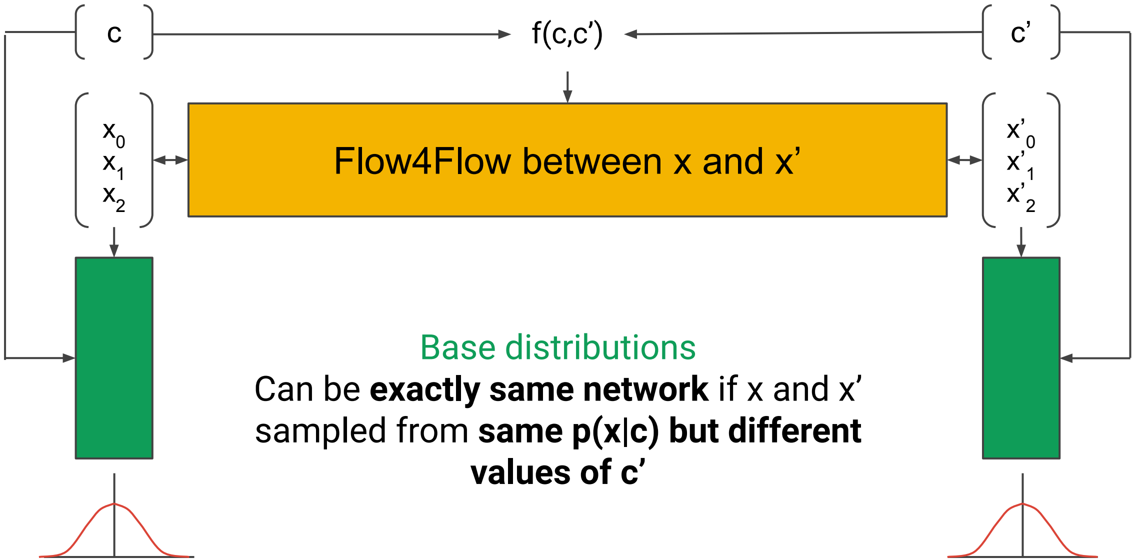
Flows4flows

Train a flow between arbitrary distributions

Simply another change of variables for p(z) in normalizing flows!

Pretrain base distribution(s)

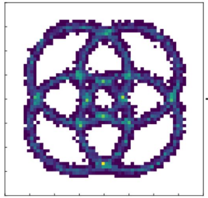
Use base distribution for loss in exact maximum likelihood



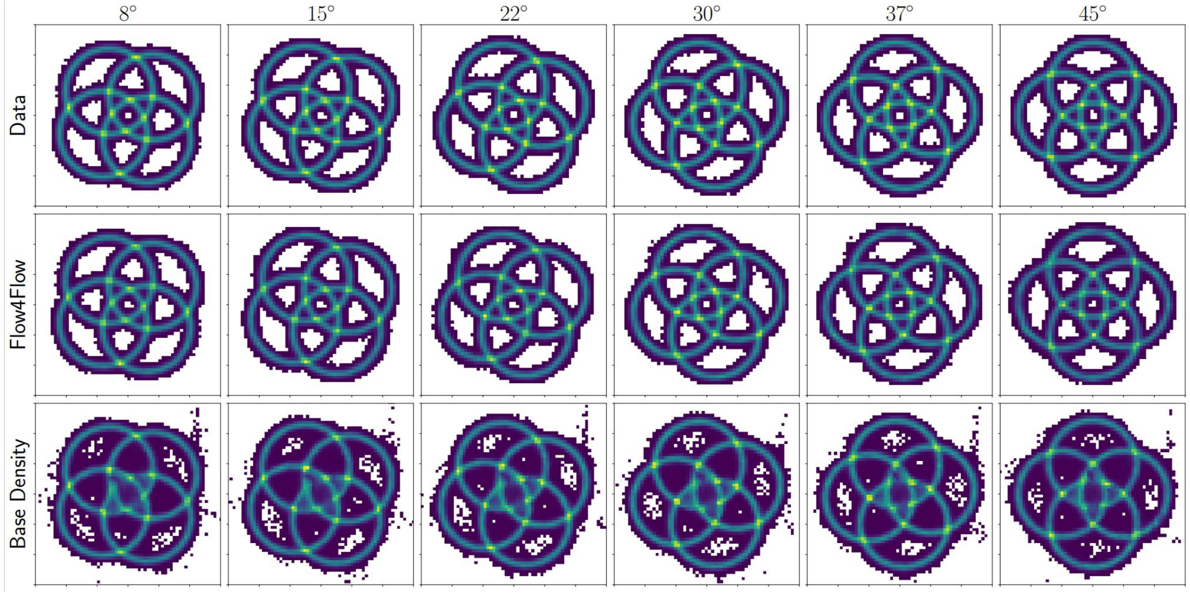
Can now train CURTAINs with exact maximum likelihood!

Why Flows4Flows

Input and transform with F4F

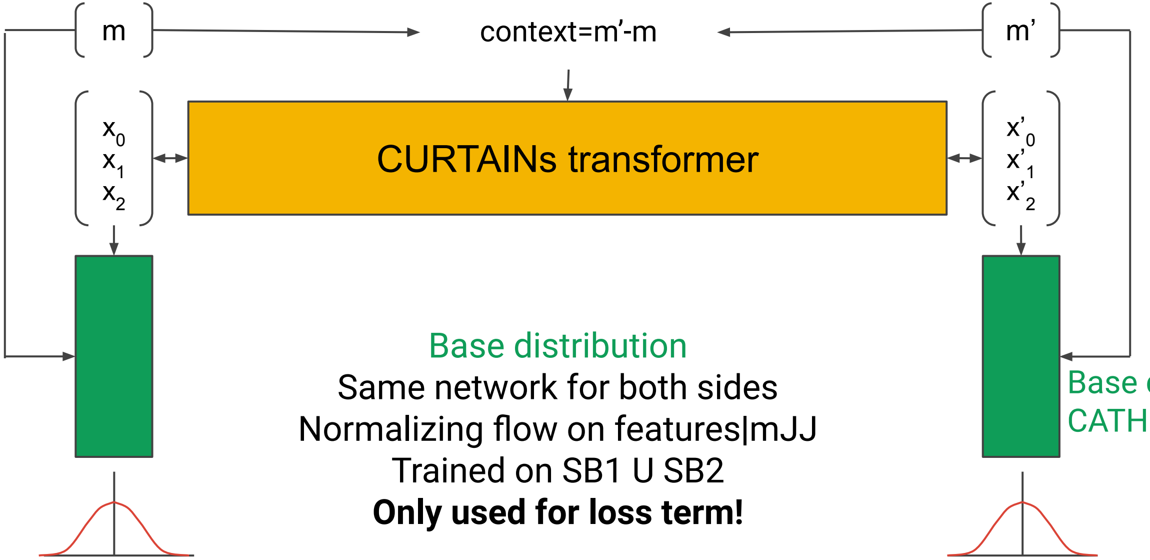


Sample from conditional normalizing flow for desired angle



Much reduced out of distribution points!

CURTAINs - Flows4Flows



Base distribution

Same network for both sides

Normalizing flow on features|mJJ

Trained on SB1 U SB2

Only used for loss term!

During training:
Randomly assign target mass by shuffling input values in batch

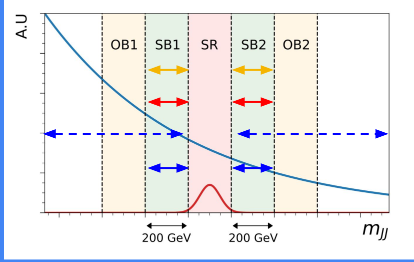
For application use SR mass values

Flow direction depends on if target > input mass

Base dist now similar to CATHODE flow

Loss: $\log(p(x)) = \text{logdet}|J(f(x|m,m'))| + \text{logdet}|J(g(f(x|m,m')|m') + p(g(f(x|m,m')|m'))$

CURTAINs - Flows4Flows



Significant improvement with new loss!

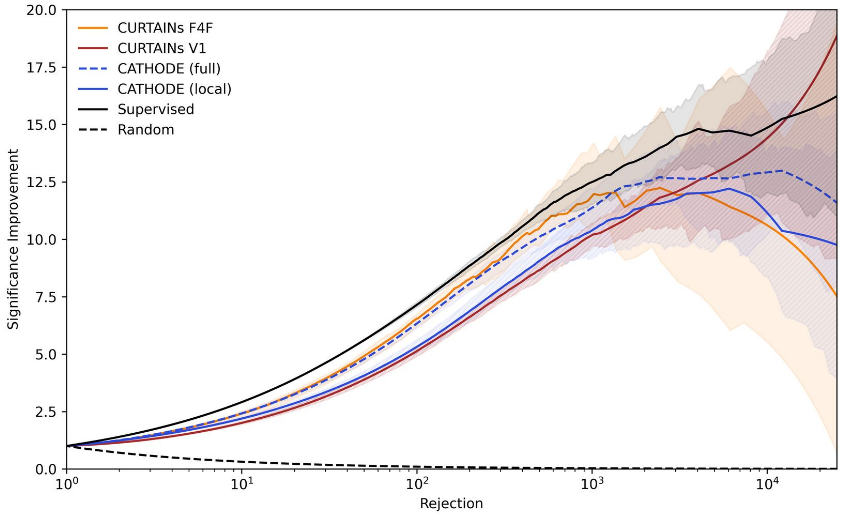
Much faster to train, *including* base density

Still trained on a very local window

- Only 200GeV either side of SR
- Matches CATHODE (full) now over most of the range

Compared to CURTAINs v1

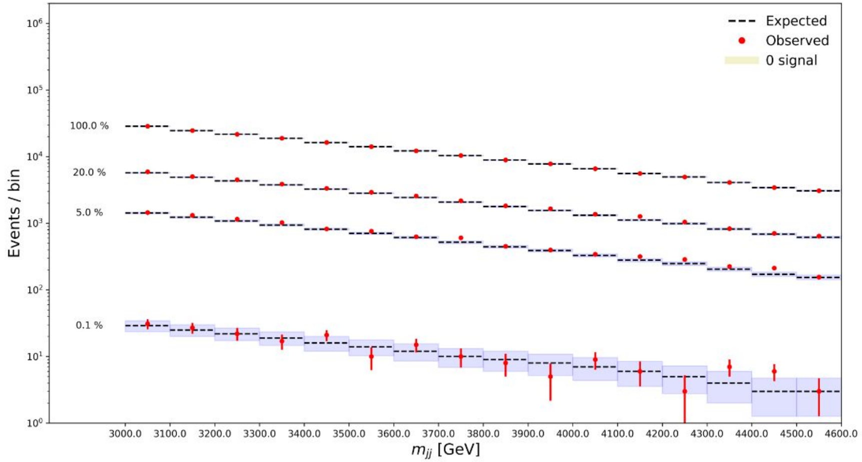
- **Simpler** to set up and train
- Features can be even more strongly correlated to resonant feature



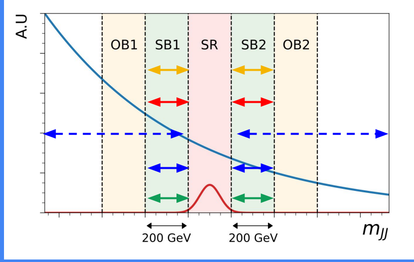
CURTAINs - Flows4Flows

Compared to CURTAINs v1

- **Simpler** to set up and train
- Features can be even more strongly correlated to resonant feature
- Still **robust** to case where there is no signal

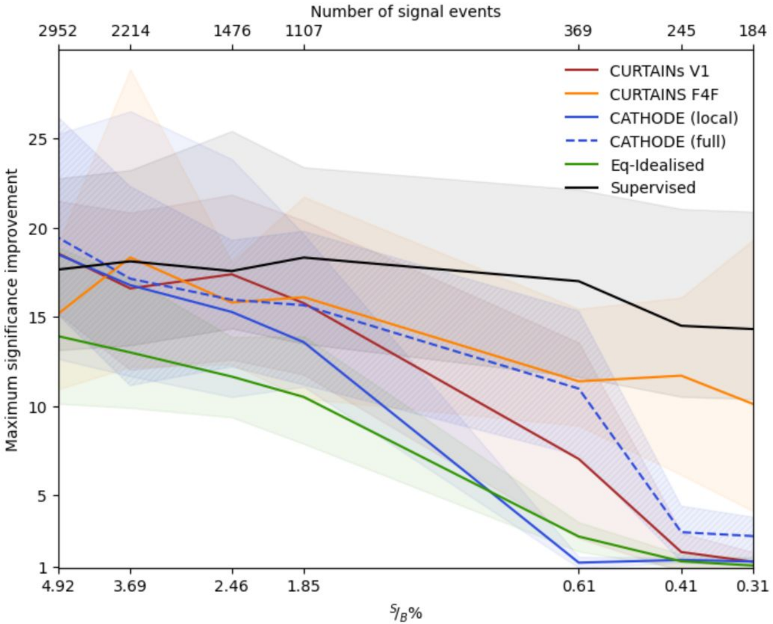


CURTAINs - Flows4Flows



Compared to CURTAINs v1

- **Simpler** to set up and train
- Features can be even more strongly correlated to resonant feature
- Still **robust** to case where there is no signal
- **Much more sensitive** to even small amounts of signal



Summary

CURTAINs is a new method for enhancing the Bump Hunt with CWoLa style classifiers

- Transforms data from sidebands into signal region
- Bypass need of going via an intermediate distribution
- Produces background data in SR and is complementary to other anomaly techniques

CURTAINs matches the performance of leading approaches without needing to train on the full mJJ spectrum

- Leading performance in a local setup

CURTAINs+Flows4Flows can reach even higher levels of performance - preprint soon!

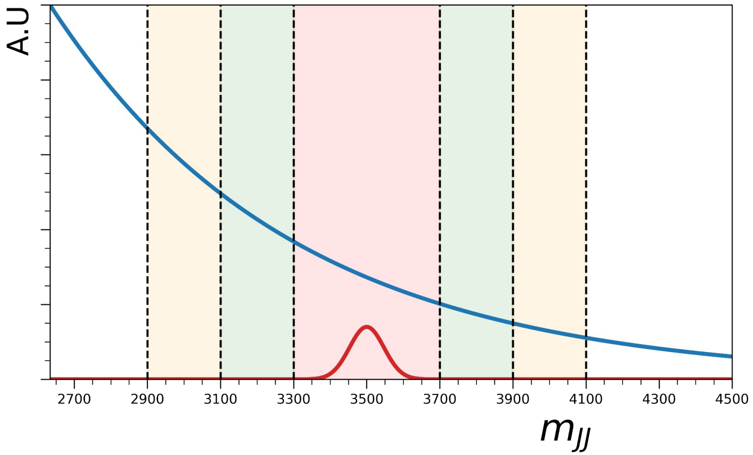
Backup

CURTAINs - CWoLa Performance

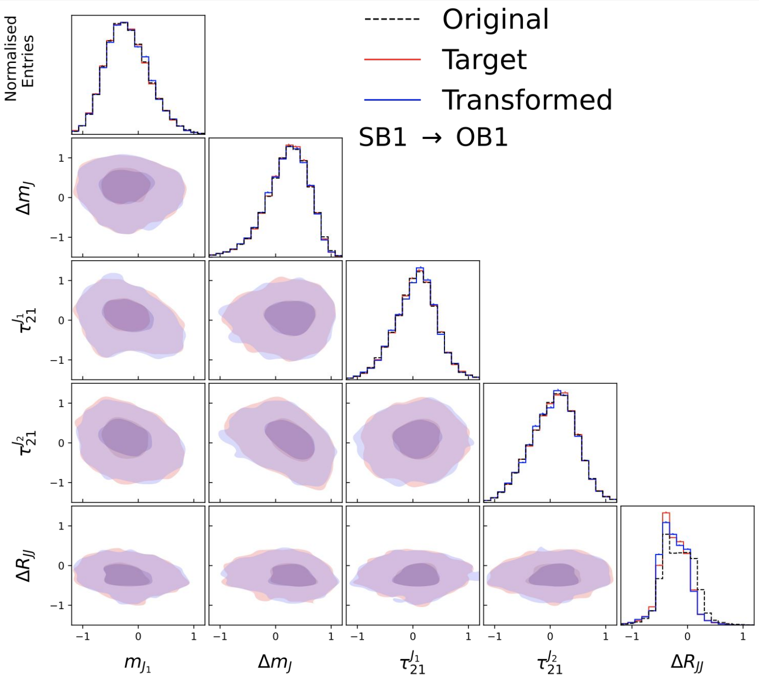
Fix the signal region such that it contains almost all of the signal

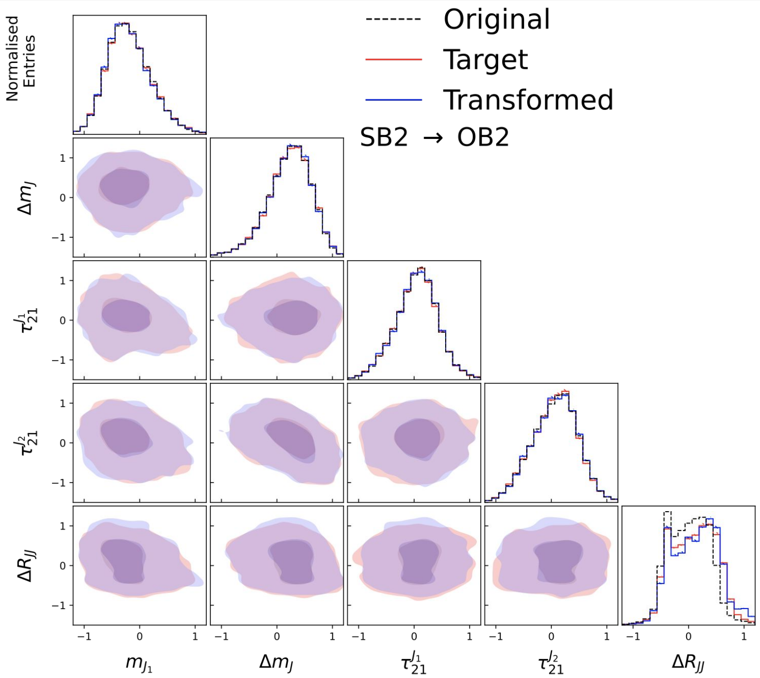
Train a CURTAINs transformer

Train a classifier SR data vs CURTAINs

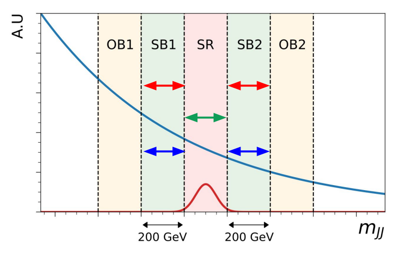


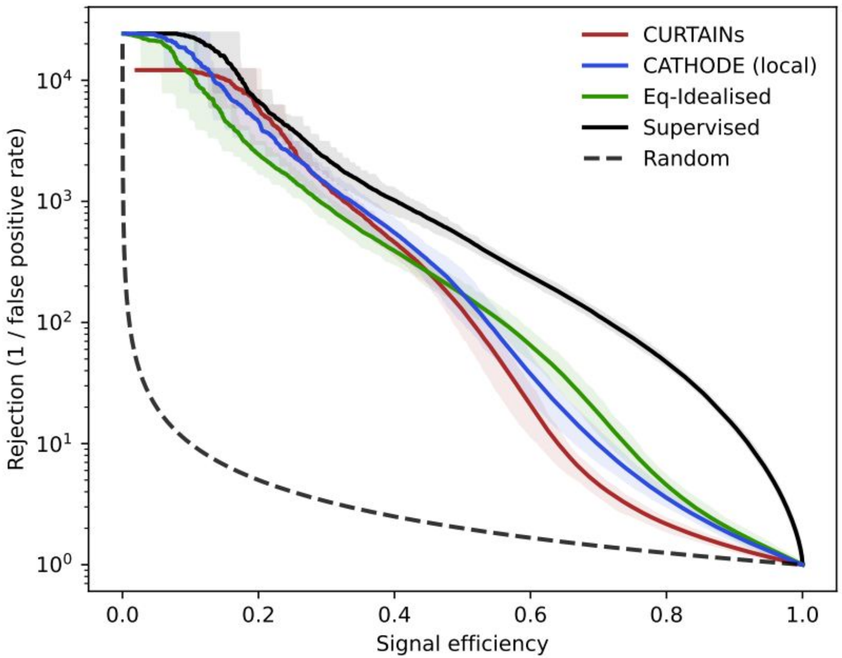
CURTAINs - Validation regions

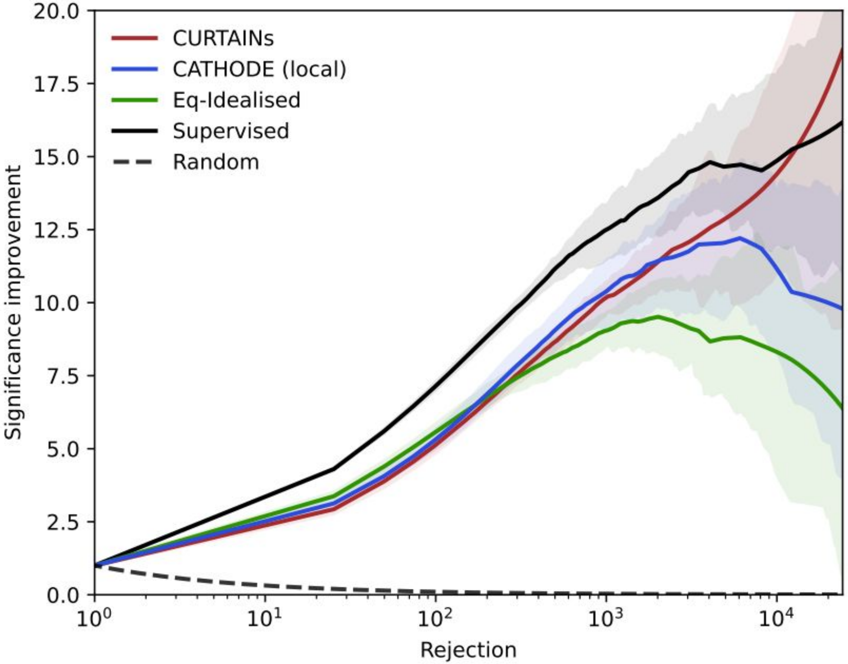




CURTAINs - CWoLa Performance





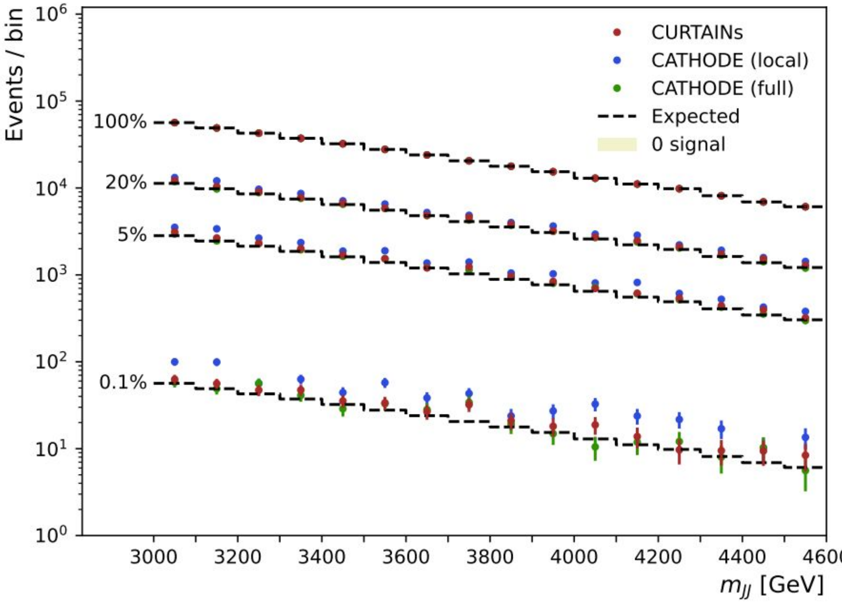


Idealised = take true Background data, and train S+B vs B with equal statistics

CURTAINs - Bump hunt

Repeat CWoLa setup with non-overlapping 200 GeV steps

Apply cuts on classifier trained with e.g. CURTAINs and look for a bump



CURTAINs - Flows4Flows

Train normalizing flow on $p(x, m) = SB1 \cup SB2$ for base distribution - like CATHODE

- But unlike CATHODE, use this for training another flow, not generating samples

Construct a flow4flow from x to x' conditioned on current and target masses (m, m')

- Transform $x \sim p(x|m)$ to $p(x|m' \sim p(m))$ in flow $f(x|m,m')$ - like CURTAINs

But now loss given by maximum likelihood:

$$\log(p(x)) = \underbrace{\text{logdet}|J(f(x|m,m'))|}_{\text{Flow4flow transform}} + \underbrace{\text{logdet}|J(g(f(x|m,m')|m')}_{\text{Base density transform}} + \underbrace{p(g(f(x|m,m')|m'))}_{\text{Base density probability}}$$

CURTAINs - Flows4Flows

If a datapoint has m’ >= m, transform from “left to right”

If a datapoint has m’ < m, transform from “right to left”

No longer need to split sideband, train on combination of all data

- Guaranteed widest support of conditioning variable!
- Leads to much faster training, no longer iterating
    - Forward and inverse pass based on input/target, both passes done per batch

CURTAINs - Flows4Flows