Finding new physics using generative models

Petrov Oleg
Supervisor: Mikhail Hushchyn

Laboratory of Methods
for Big Data Analysis

Introduction

New Physics is anything that differs from the Standard Model.

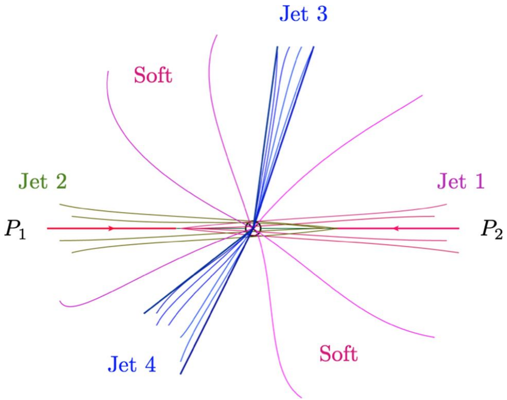
Standard Model is complete, but has a number of drawbacks and contradictions.

One searching approach is to analyze record energies experiments on Large Hadron Collider.

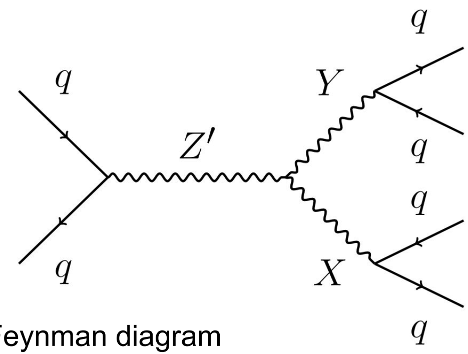
For this task, deep generative models have recently become widely used.

Dataset (LHC Olympics 2020s)

Background: QCD* dijet** events



Signal: (New Physics): Z` → XY



Feynman diagram

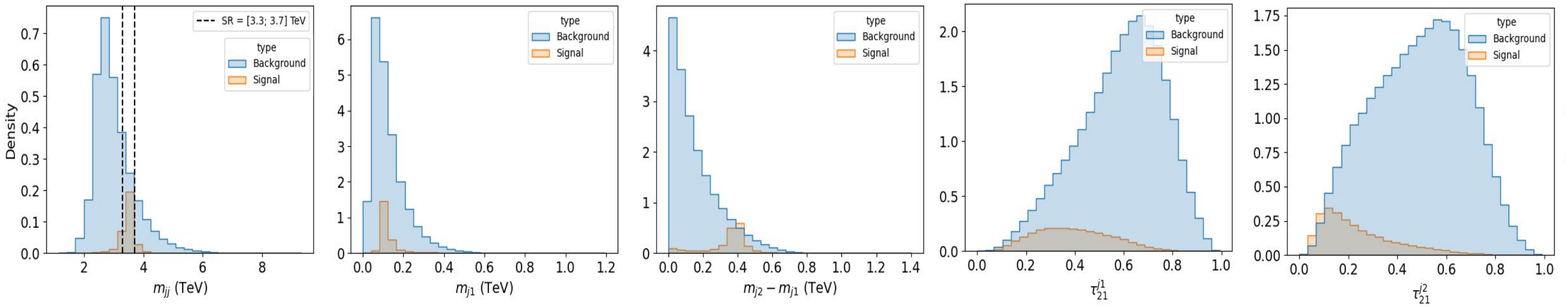
Wanted: to detect Z`

**quantum chromodynamics (QCD)* is the theory of the strong interaction between quarks mediated by gluons
***dijet event* is a collision between subatomic particles that produces two particle jets

Features

Feature space is:

- Invariant mass of dijet system m_{JJ}
- Invariant mass of lighter jet m_{J_1}
- Invariant masses difference Δm_J
- n-subjettiness ratios $\tau_{21}^{J_1},\ \tau_{21}^{J_2}$



Goal

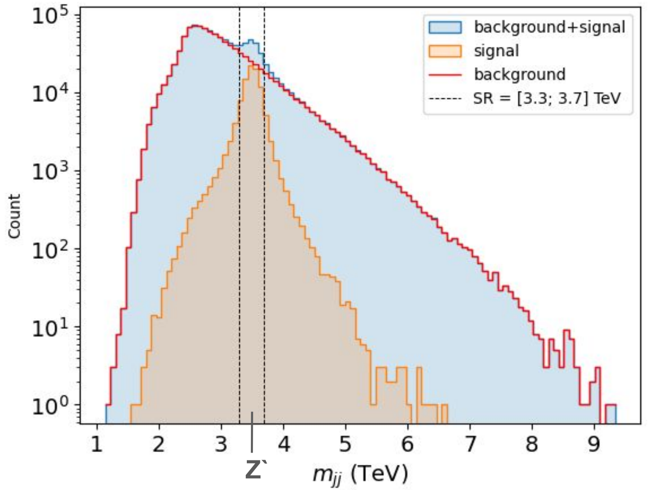
Goal: to distinguish Standard Model *background* events from rare *signal* events

Background: *Standard Model* data distribution (**dijet**)

Signal: supposedly *New Physics* events (particle **Z`**)

Supposed that signal and background have *different distributions*

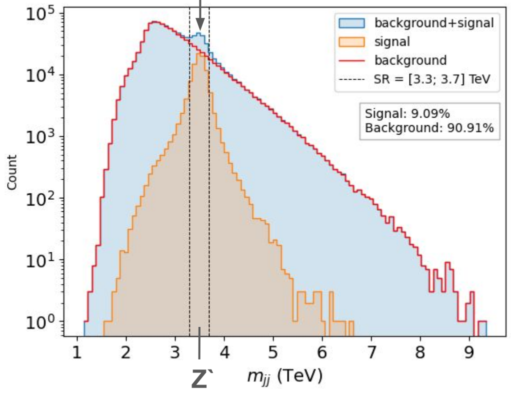
Signal's mass m_{JJ} is located in [3.3; 3.7] TeV

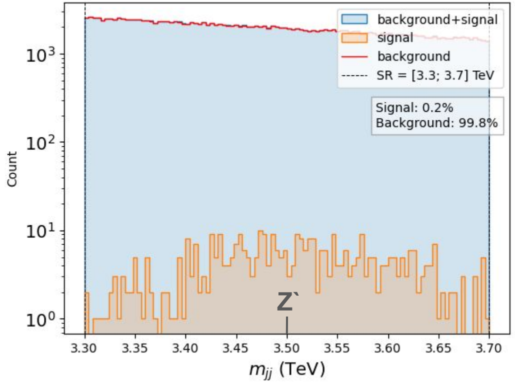


Task Complexity

Why is it hard?

- Cannot be detected as outliers (bump)
- Signals are extremely rare (<1%)
- No applied ground truth



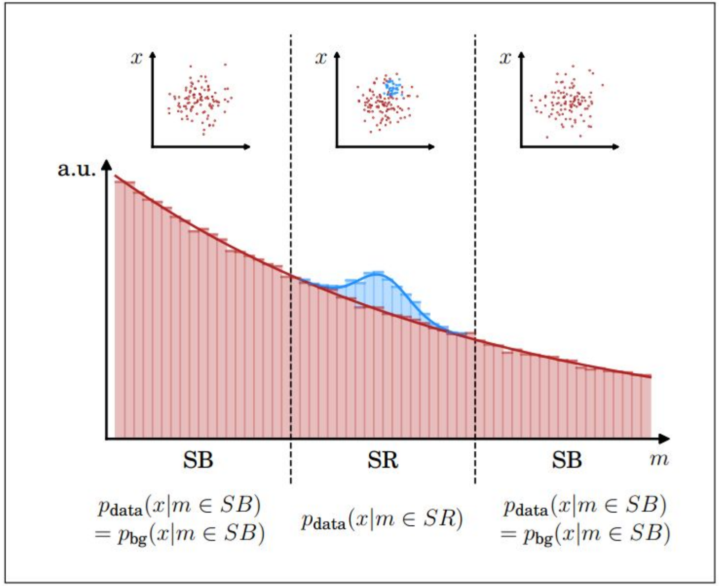


ML task: unsupervised signal detection

CATHODE Approach

An approach consists of 4 steps:

- To train generative network on Side-Band (SB) data with features ***x*** conditioned by ***m***
- To sample data into Signal Region (SR) using the trained network conditioned by ***m***
- To train classifier to distinguish synthetic and real data on SR
- To apply the trained classifier to detect New Physics events



Step 1: Density estimation

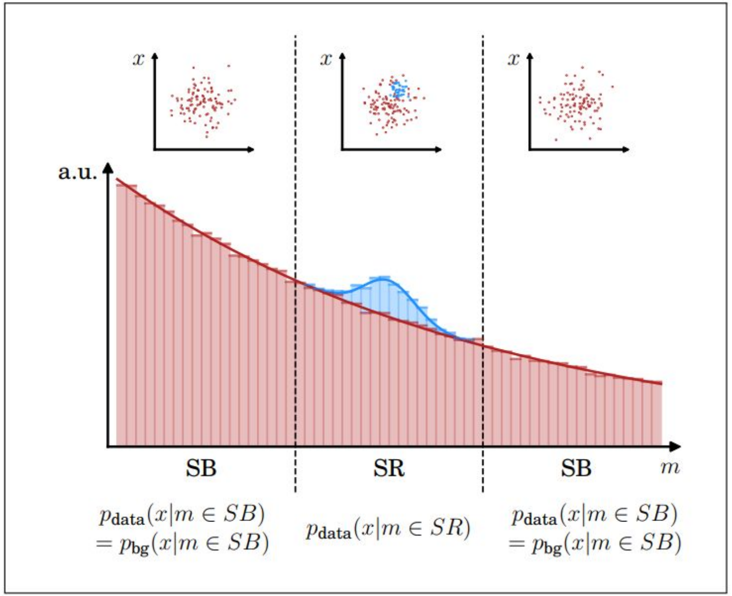
Using **SB** data to learn (non-explicitly) background distribution of features ***x*** conditioned by ***m*** – $p_{\mathrm{data}}(x|m \in SB)$

Estimated distribution should match background distribution into **SB** data:

$$p_\theta(x \mid m \in \mathrm{SB}) = p_{\mathrm{background}}(x \mid m \in \mathrm{SB})$$

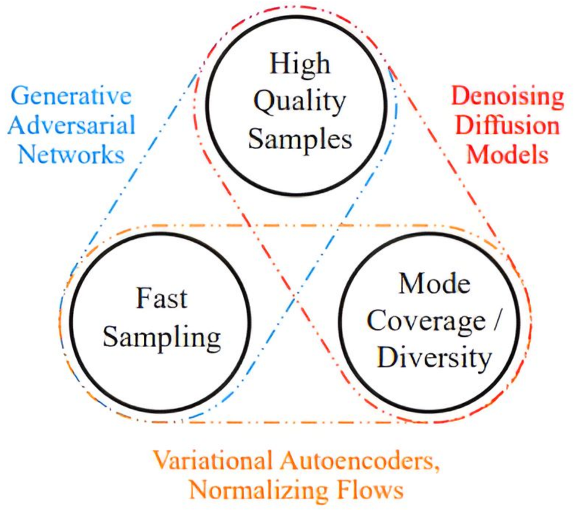
where:

$$x = \left(m_{J_1},\ \Delta m_J,\ \tau_{21}^{J_1},\ \tau_{21}^{J_2}\right)$$
$$m = m_{JJ}$$



Which model to use?

Generative Learning Trilemma



Models generate new data from prior distribution $\mathcal{N}(0,1)$

To condition this, conditional prior $\mathcal{N}(0,1|m)$ is used

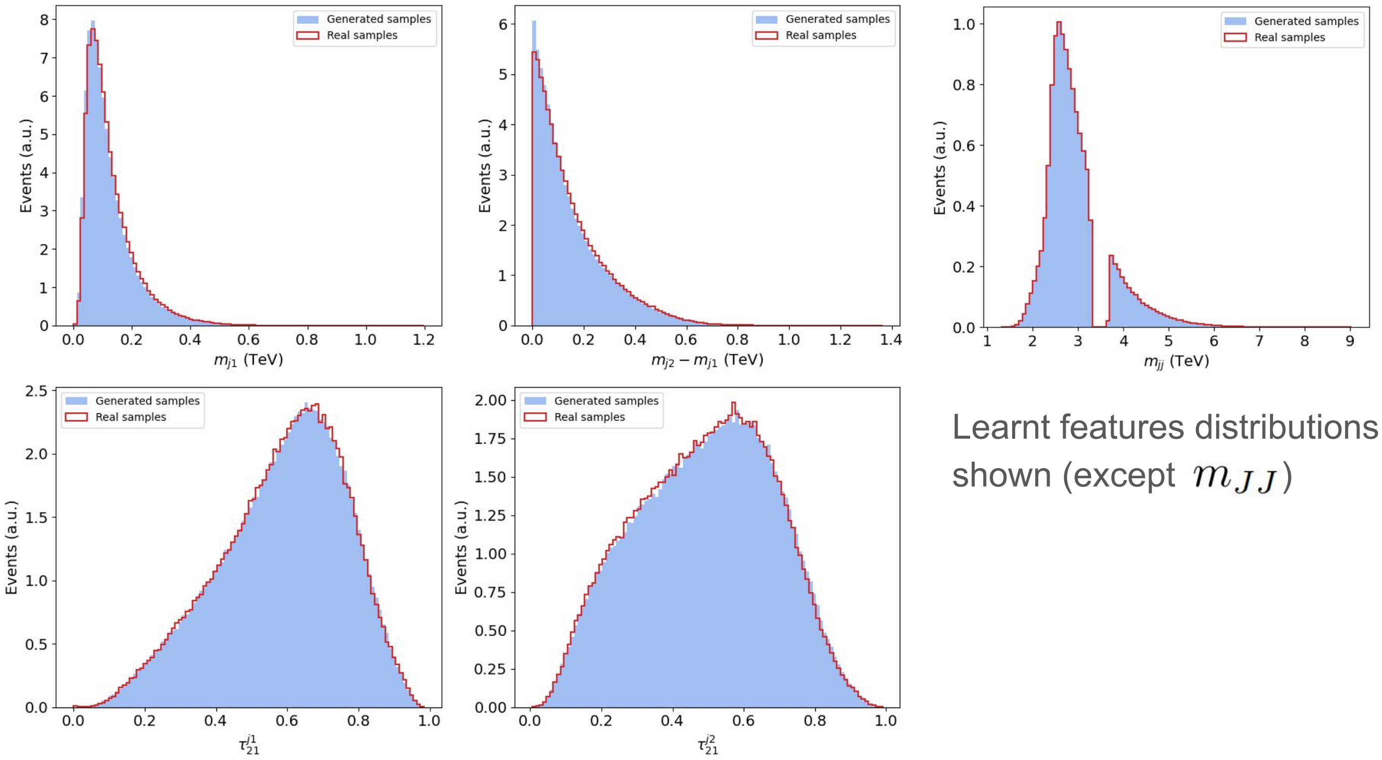
Generative model
Deep Denoising Probabilistic Model
Conditional VAE
Masked Autoregressive Flow

Density estimation results

Metric \ Model	DDPM	MAF	CVAE	Best Possible
Frechet Distance	0.0035 ± 0.0003	**0.0004 ± 0.000084**	0.0031 ± 0.0003	0.00015 ± 0.00005
Kolmogorov-Smirnov	0.012 ± 0.00045	**0.004 ± 0.00036**	0.01 ± 0.0004	0.003 ± 0.00032
Cramer-von Mises	19.75 ± 1.414	**0.76 ± 0.144**	8.15 ± 0.72	0.37 ± 0.13
Anderson-Darling	140.5 ± 9.7	**5.2 ± 0.98**	61.1 ± 3.9	1.4 ± 0.7
Kullback-Leibler	$(216 \pm 23) * 10^{-6}$	$\mathbf{(53 \pm 9) * 10^{-6}}$	$(361 \pm 14) * 10^{-6}$	$(40 \pm 5) * 10^{-6}$
Jensen-Shannon	$(56 \pm 3) * 10^{-6}$	$\mathbf{(12 \pm 3) * 10^{-6}}$	$(92 \pm 5) * 10^{-6}$	$(8 \pm 2) * 10^{-6}$

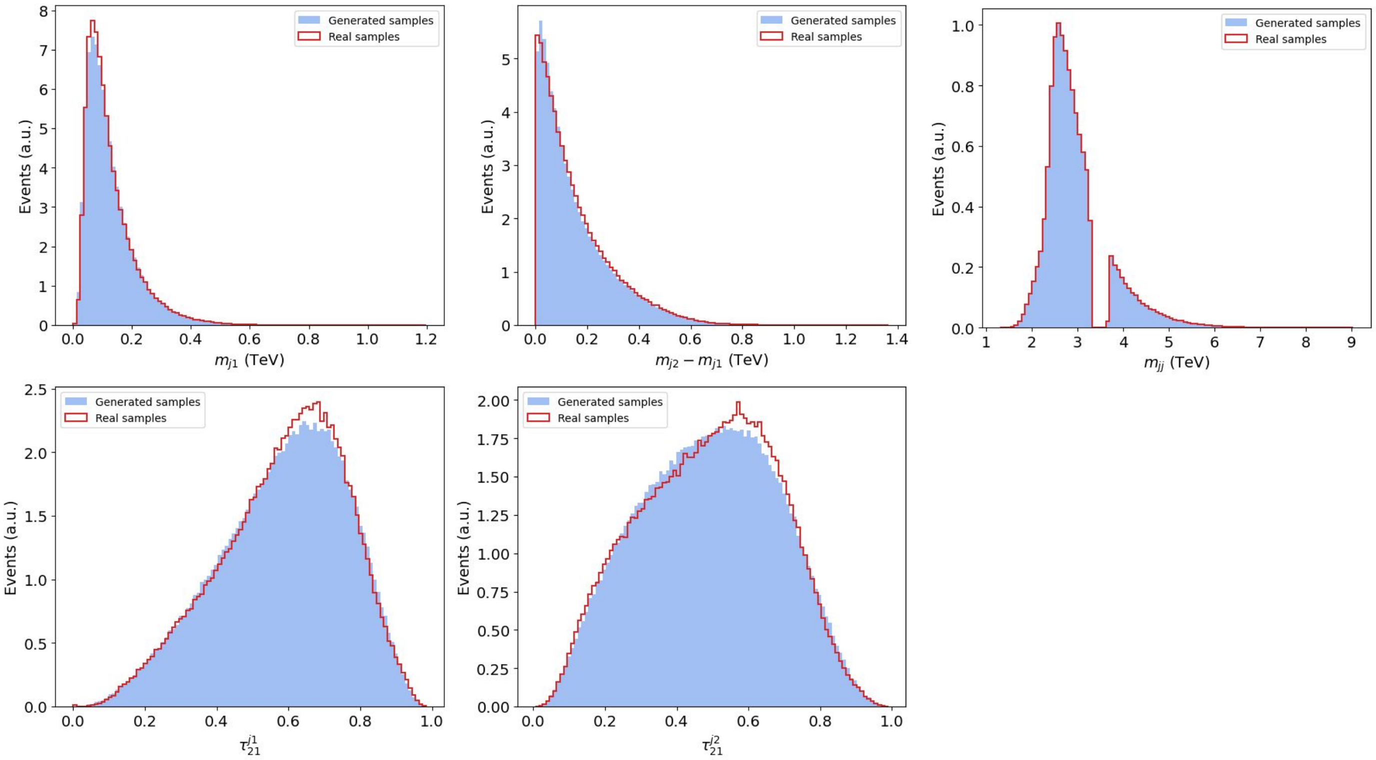
Computed within SB region (sampled background vs real data)

DDPM: density estimation

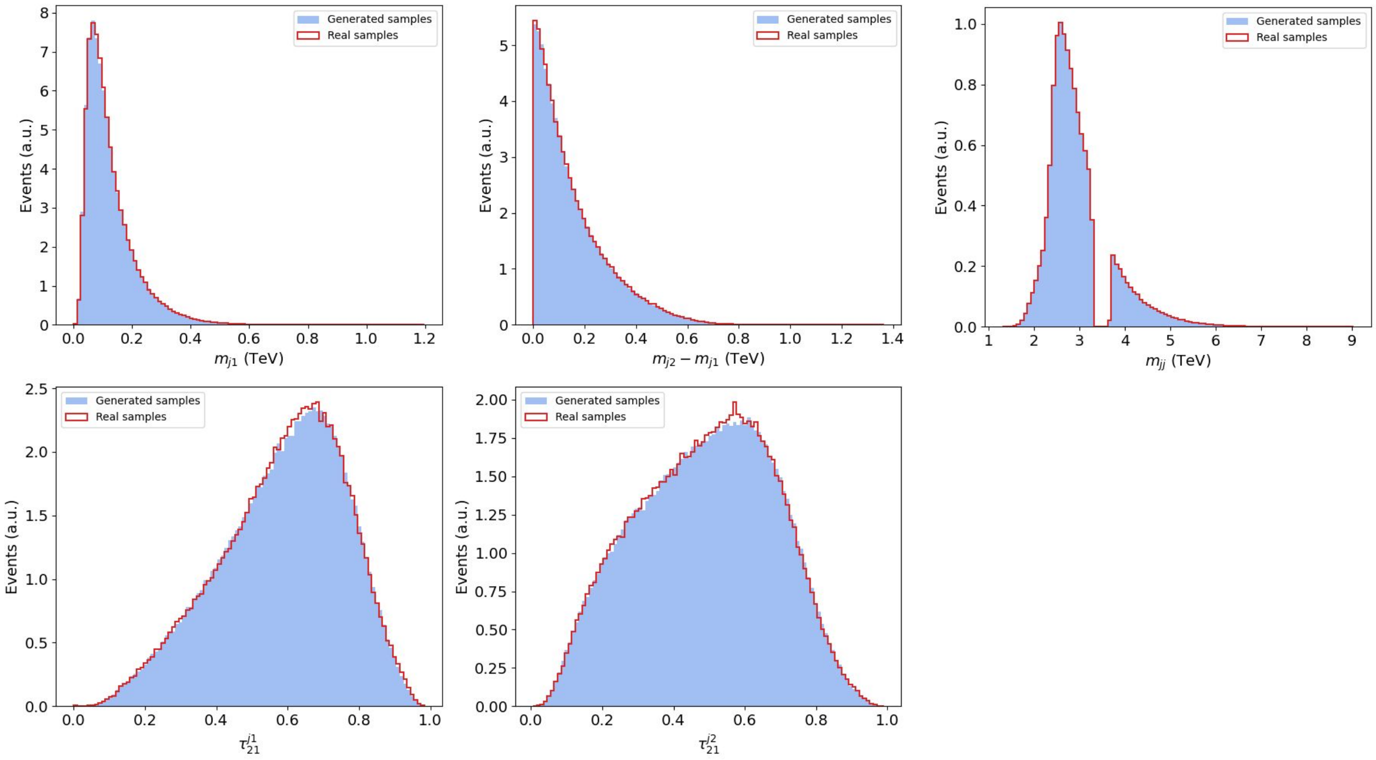


Learnt features distributions on **SB** are shown (except m_{JJ})

CVAE: density estimation



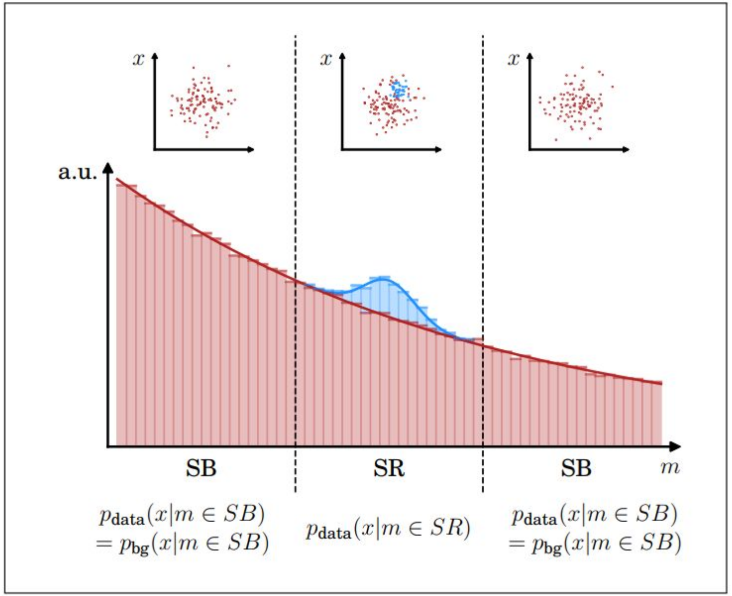
MAF: density estimation



Step 2: Interpolation and conditional sampling

As the generative model is trained on the **SB** region background data, we can sample new background events **x** by interpolating the estimated *PDF* into the **SR**:

$$x \sim p_\theta(x \mid m \in \mathrm{SR})$$



For conditional sampling, a range of values of the invariant mass in the **SR** is used.

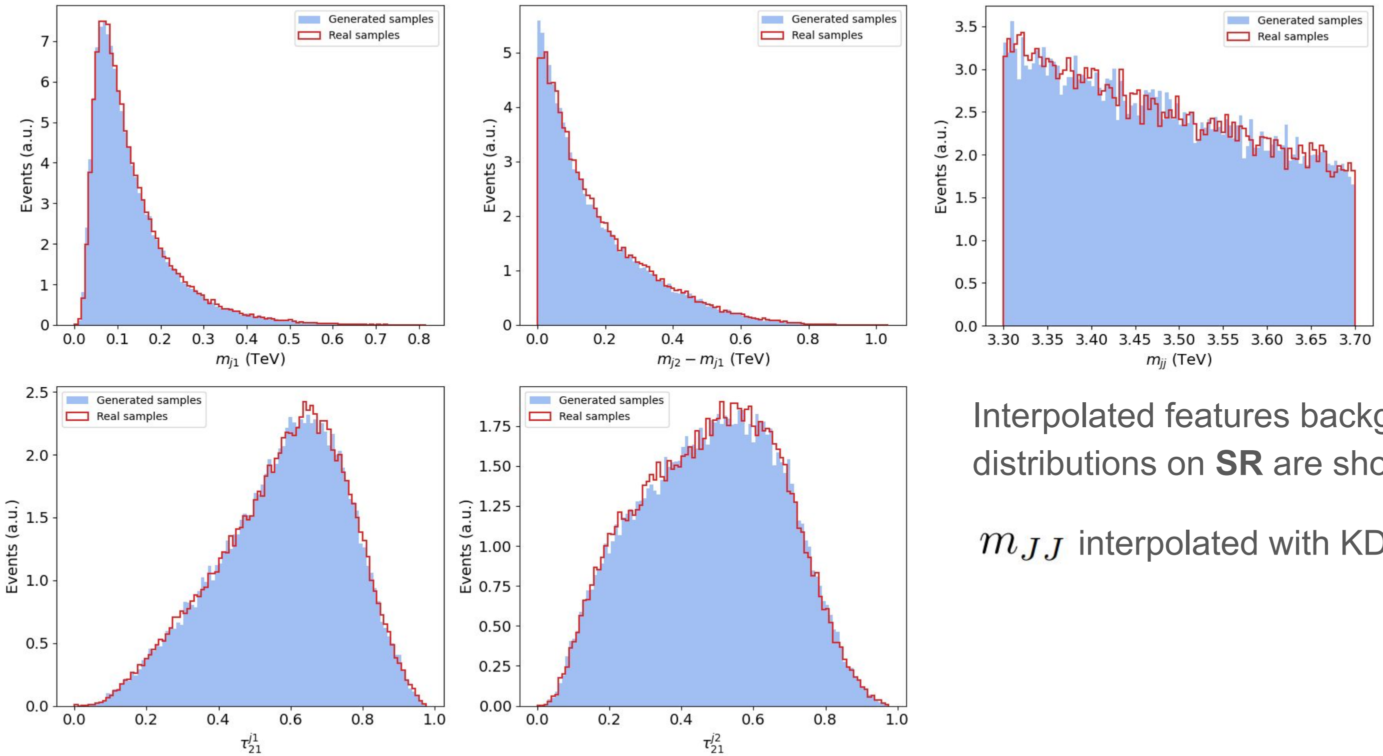
This range comes from $m \sim p_{\mathrm{KDE}}(m \in \mathrm{SR})$.

Interpolation results

Metric \ Model	DDPM	MAF	CVAE	Best Possible
Frechet Distance	**0.00155 ± 0.00044**	0.00159 ± 0.00044	0.0066 ± 0.0009	0.00045 ± 0.00017
Kolmogorov-Smirnov	0.01 ± 0.00112	**0.009 ± 0.0009**	0.016 ± 0.001	0.005 ± 0.00076
Cramer-von Mises	1.2 ± 0.326	**0.6 ± 0.19**	2.9 ± 0.42	0.16 ± 0.064
Anderson-Darling	8.1 ± 2.3	**3.6 ± 1.2**	22.7 ± 2.7	-0.0085 ± 0.47
Kullback-Leibler	(64 ± 21) * 10^{-6}	**(57 ± 11) * 10^{-6}**	(477 ± 32) * 10^{-6}	(16 ± 8) * 10-6
Jensen-Shannon	**(12 ± 4) * 10^{-6}**	(18 ± 7) * 10^{-6}	(121 ± 16) * 10^{-6}	(5 ± 1) * 10-6

Computed within SR region (sampled background vs real *only background* data)
10 models with best validation loss are used for sampling

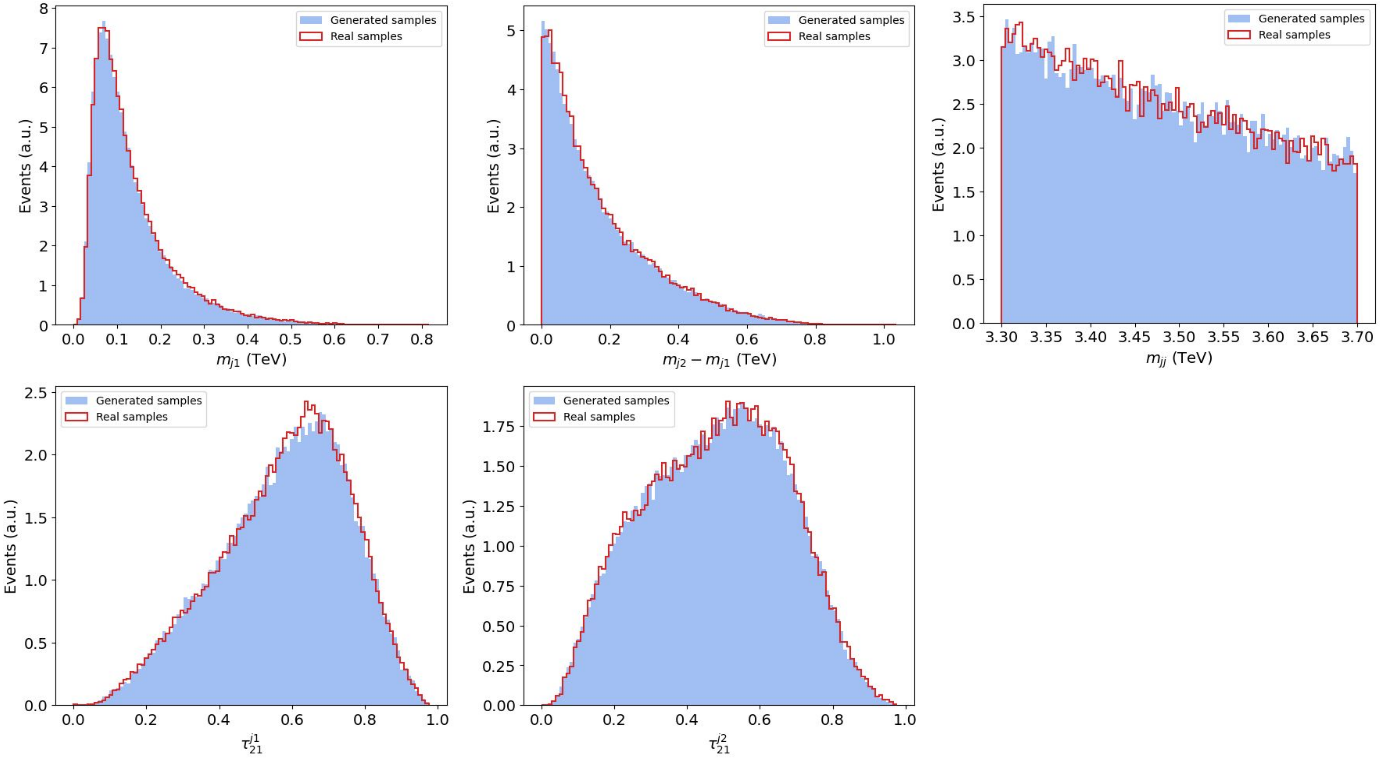
DDPM: BG interpolation on SR



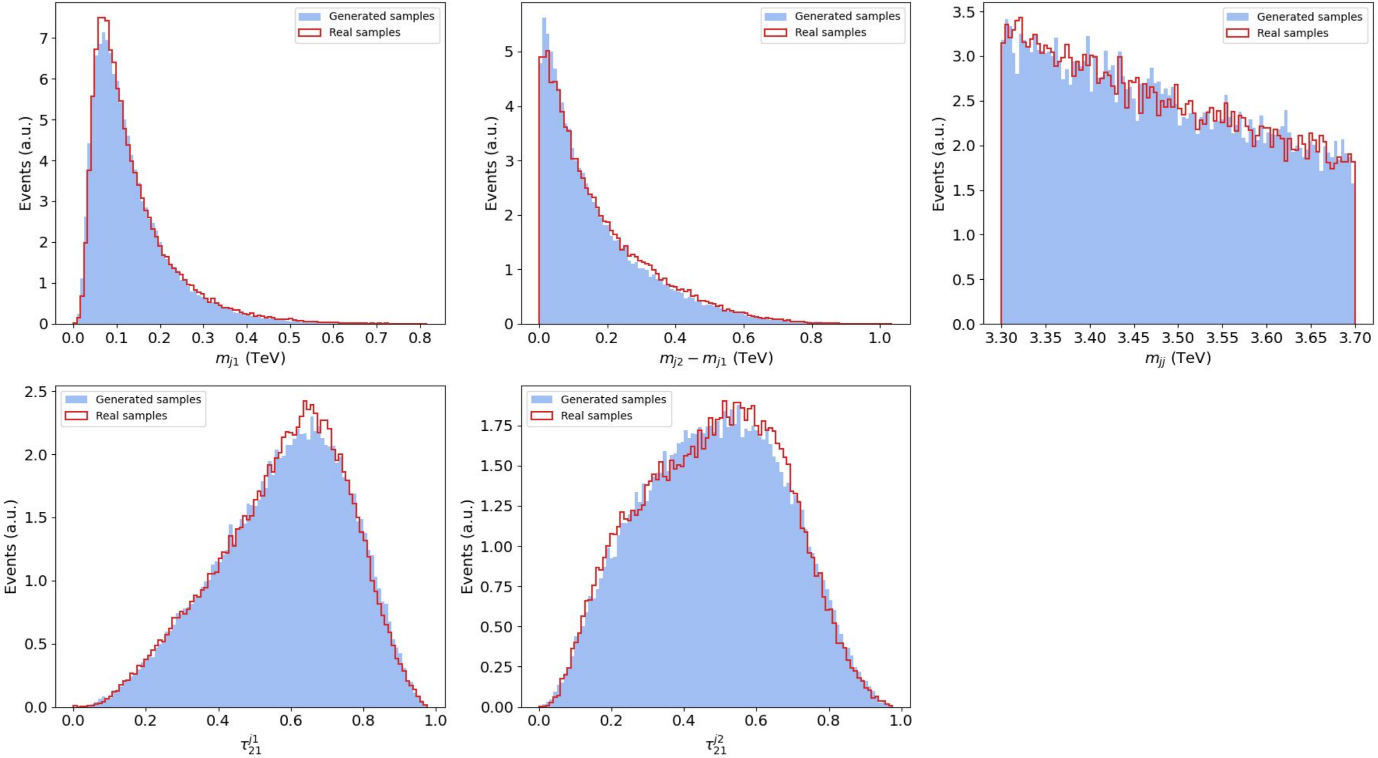
Interpolated features background distributions on **SR** are shown

m_{JJ} interpolated with KDE

MAF: BG interpolation on SR



CVAE: BG interpolation on SR



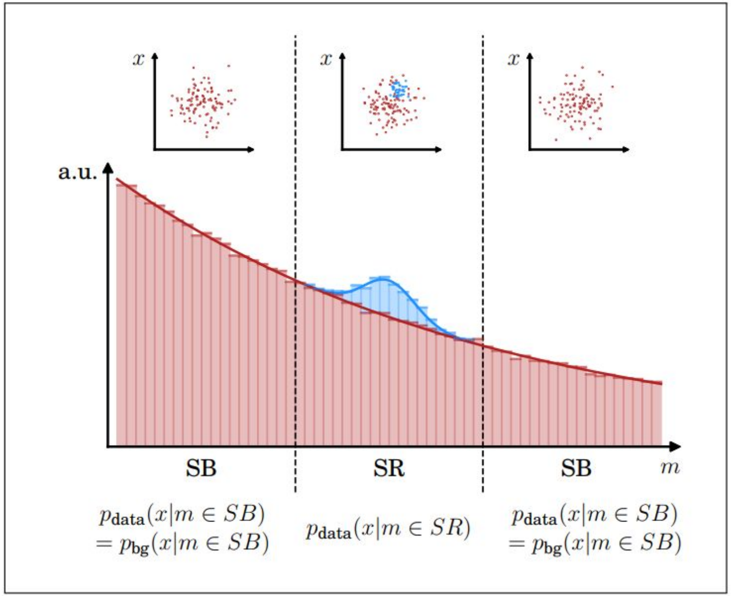
Steps 3: Classification

Classification step is to train a classifier to distinguish the *generated* background events from the *real* data (background + signal) into the **SR**.

According to Neumann-Pearson lemma, optimizations of $\frac{p_{\text{data}}(x)}{p_{\text{background}}(x)}$ and $\frac{p_{\text{signal}}(x)}{p_{\text{background}}(x)}$ are equivalent.

Classifier used: MLP.

Signal fracture: 0.15%.



Step 4: Detection

Detection step is to apply the trained classifier to the real data into the **SR**:

- Positive label is now the signal data
- Negative label is now the background data
- Predict new real-vs-sample labels
- Evaluate metrics on signal-vs-background ground truth

As the real data in the **SR** mostly matches background, positive-tagged objects by classifier are signal-like due to PDFs difference.

Curves metrics

- *Signal efficiency,* or *sensitivity* (**TPR**)
- *Background efficiency* (**FPR**)
- *Background rejection* (*inverse background efficiency*)
- *Significance Improvement Characteristic* (**SIC**)

  for classifier *m* and threshold *t* defined as:

$$\mathrm{SIC}(m,t) = \frac{\mathrm{TPR}(m,t)}{\sqrt{\mathrm{FPR}(m,t)}}$$

The *SIC-curve* is the SIC values calculated at all thresholds and plotted versus the signal efficiency

Detection: results

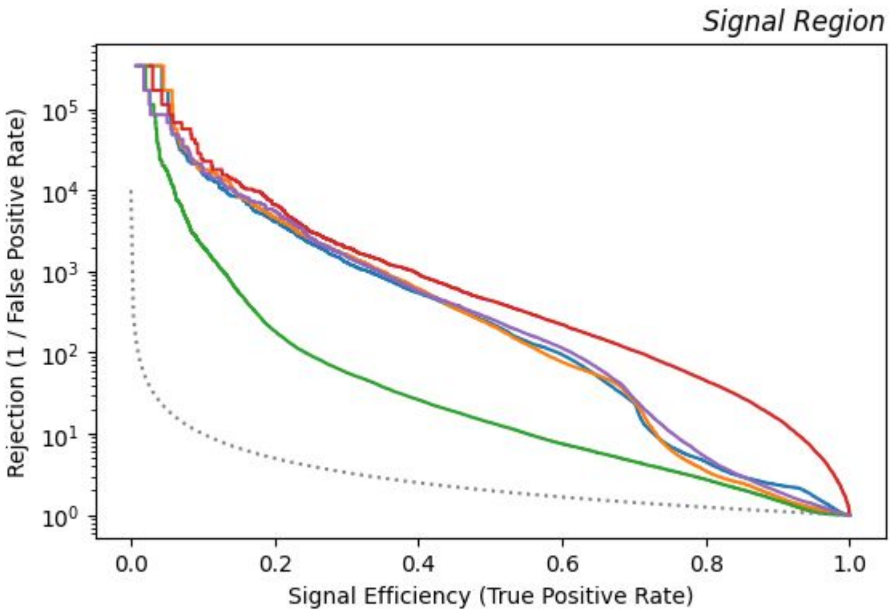
Predictions are aggregated from 10 best model states by mean

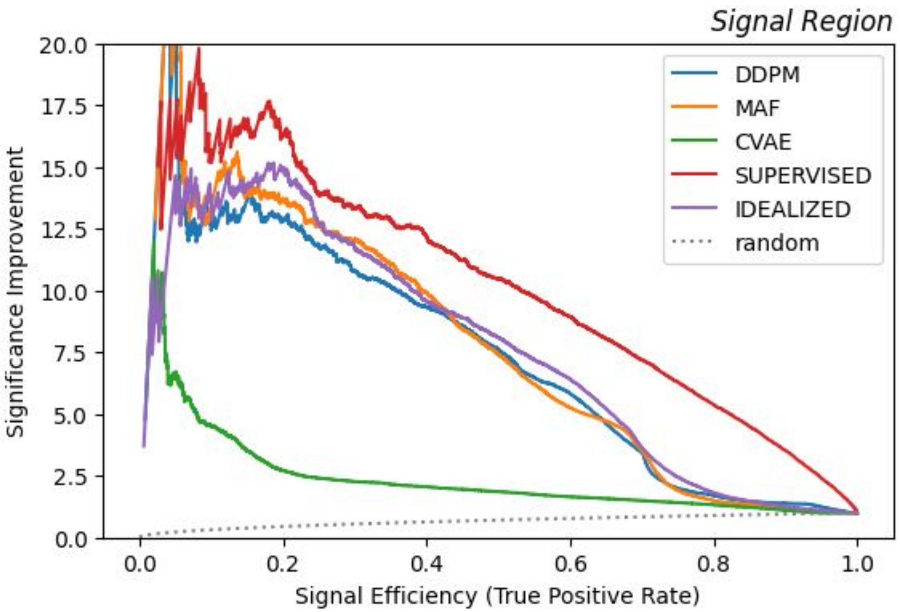
The best run is chosen by median AUC-SIC

Metric \ Model	Generative			Highly-idealized*	
	DDPM	MAF	CVAE	Supervised	Idealized
AUC-PR	**0.675**	0.67	0.387	0.823	0.69
AUC-ROC	**0.886**	0.858	0.798	0.971	0.878
AUC-SIC	7.29	**7.49**	2.32	9.828	7.56

**Highly-idealized* methods set an upper bound on quality. *Supervised* classifier is trained directly on signal-vs-background task, whereas *Idealized* classifier is trained using the perfect simulation dataset, provided by CATHODE's authors

Detection: results





The greater area-under-curve, the better is the method | chosen for the best run

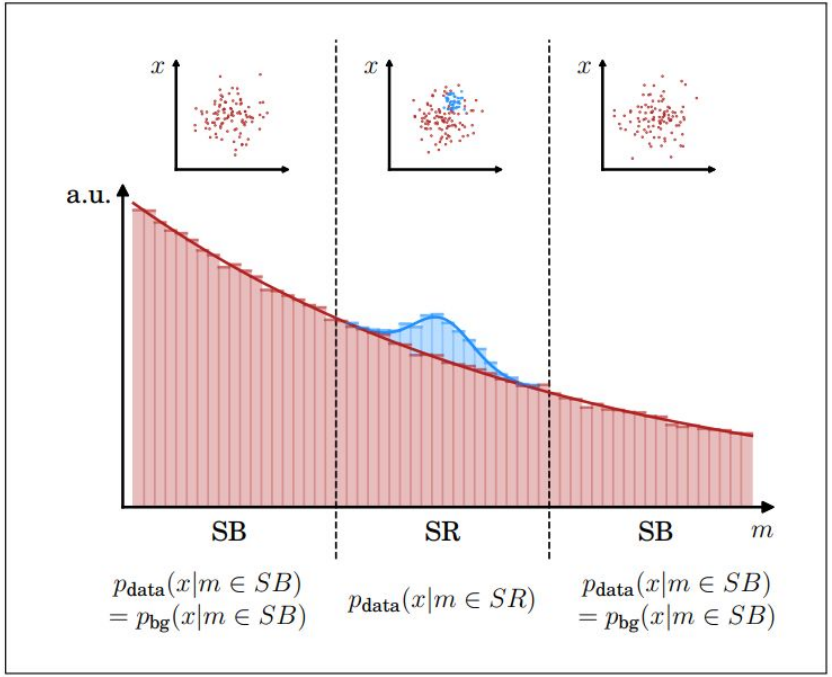
Results

- SOTA approach CATHODE is studied in detail
- Other generative approaches are tested: DDPM, CVAE
- A novel approach with diffusion network:
    - Implemented wrt. tabular data and conditioning
    - Training time 5 times faster than CATHODE’s
    - Performs closely to idealized classifier
    - Has a wide range of improvements

necroshine0/new_physics

Data partition by steps

- Density estimation set (1)
  - 500k/380k **SB** background – real data
- Interpolation (2)
  - sample 200k/200k **SR** background
- Classification set (3)
  - 200k/200k **SR** background – sampled data from (2)
  - 60k/60k **SR** data (not only BG) – real data
- Detection evaluation set (4)
  - 340k **SR** background – real data
  - 20k **SR** (not only BG) – real data



*train/valid