Unsupervised tagging of semivisible jets with normalized autoencoders in CMS

Florian Eble, Annapaola de Cosa, Christoph Ribbe, Roberto Seidita

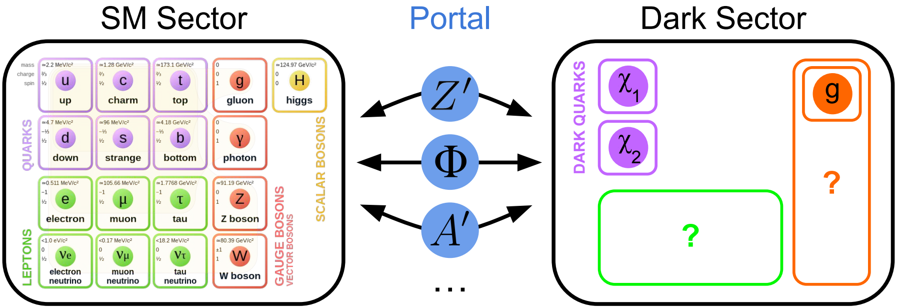
ETH zürich

19/06/2024

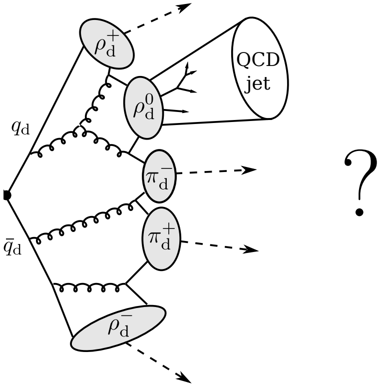
CHIPP 2024 annual meeting

DM as a strongly coupled dark sector

- Hidden Valley [arXiv:hep-ph/0604261] with new particles and forces form the dark sector
- Strongly coupled dark sector
  - ➔ New confining $SU(N)$ force, dark QCD, and dark quarks
  - ➔ Dark hadronic showers and jets
  - ➔ Experimental signature: semivisible jets (SVJs) [arXiv:1503.00009, arXiv:1707.05326]
- Portal between the standard model (SM) and dark sectors via a mediator particle



- ➔ Different jet substructure due to double hadronization
- ➔ Experimental signatures of SVJs very model-dependent
- ➔ Large parameter space to cover



The details of the shower in the dark sector depend on many unknown parameters, e.g.:

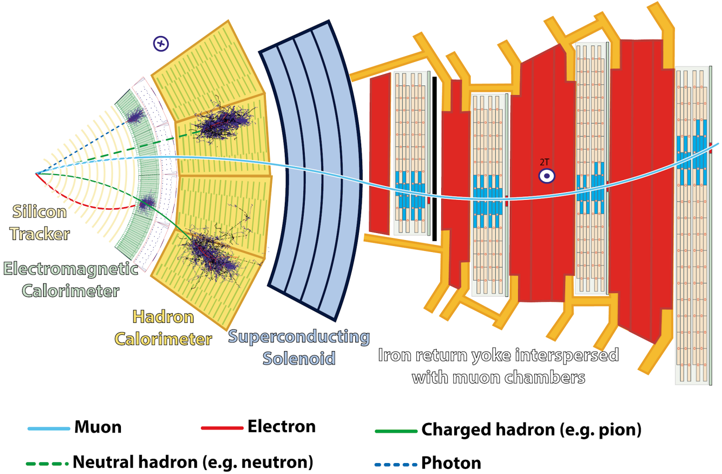
- Number of colors and flavors in the dark sector
- Masses of the dark hadrons

➔ **Use anomaly detection to identify SVJs as anomalies**

CMS detector

- The CMS detector is composed of different subdetectors allowing to identify and measure the properties of photons, electrons, muons and hadrons
- SM decay products of SM jets and SVJs can be reconstructed, and clustered into jets

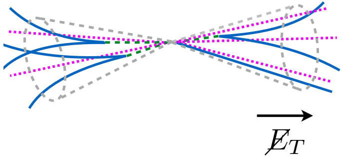
➔ Exploit the different jet substructure of SVJs compared to SM jets to tag them



SVJ experimental signature:

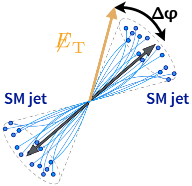
Missing transverse momentum ($\not{E}_{\mathrm{T}}$) aligned with a jet

$$\not{E}_{\mathrm{T}} = \left\| \sum \vec{p_{\mathrm{T}}} \right\|$$



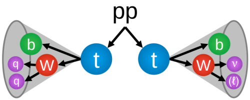
QCD multijet

- Artificial missing transverse energy $\not{E}_{\mathrm{T}}$ aligned with jet from jet energy mismeasurement
- Autoencoder-based anomaly detection proved to work well against QCD jets [arXiv:2112.02864]



$t\bar{t}$ + jets

- Semi-leptonic channel $W(\to l\nu)$ with lost lepton, genuine $\not{E}_{\mathrm{T}}$ from neutrino
- More challenging for anomaly detection



- AEs are trained to minimize the reconstruction error (e.g. MSE) between input and output:

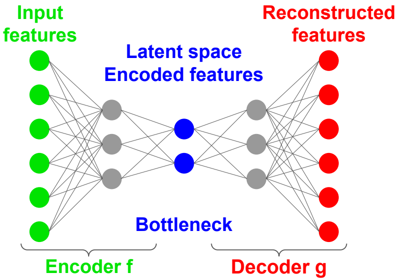
$$L(x) = ||g(f(x)) - x||$$

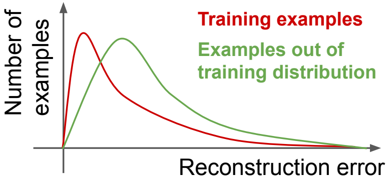
➔ Aim: that examples out of the training distribution, i.e. anomalies, have a higher reconstruction error

➔ Trained on SM data, AEs can perform signal-agnostic searches for new physics [arXiv:1808.08979, arXiv:1808.08992]

➔ Will use interchangeably:
  - "training" and "background"
  - "anomaly" and "signal"

- AE network is a fully connected NN with jet substructure input features (see backup slides 19-21)

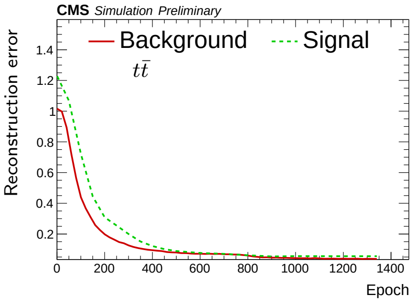


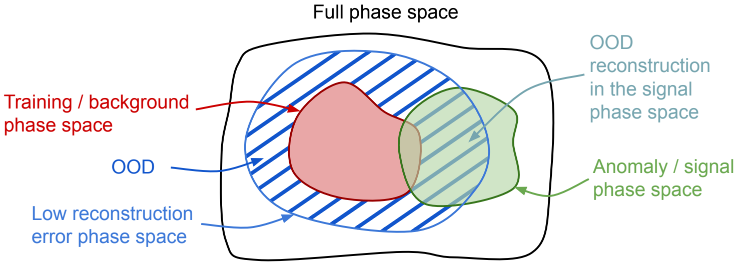


The problem of out-of-distribution (OOD) reconstruction

- Standard AEs are trained to minimize reco error in the background phase space
- but **AEs are free to minimize reco error outside the background phase space!** including the unknown signal phase space...
  ➔ **No classification power**

➔ This is the problem of **out-of-distribution (OOD) reconstruction** / "complexity bias"





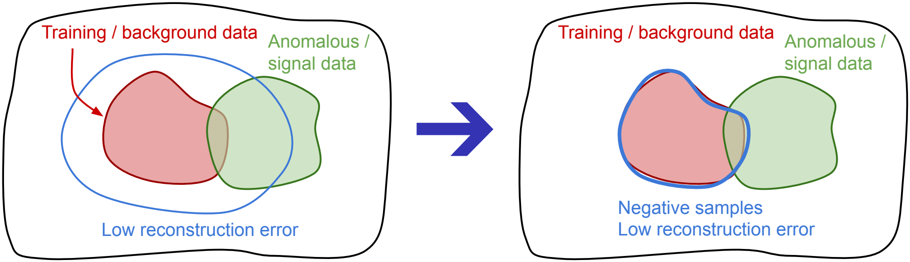
Working principle of the Wasserstein Normalized Autoencoder

Ensure that the low reconstruction error probability distribution matches that of the training data

- Define a probability distribution p_θ so that regions with low reco error E_θ have high probability

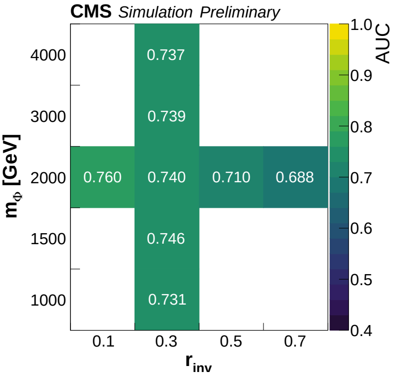
$$p_\theta(x) = \frac{1}{\Omega_\theta} \exp\left(-E_\theta(x)\right)$$

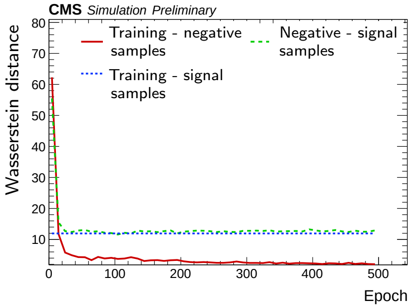
- Minimize the distance between the training and p_θ probability distributions
  - Sample from p_θ via MCMC ➔ "Negative samples"
  - Wasserstein distance between training and negative samples[1]

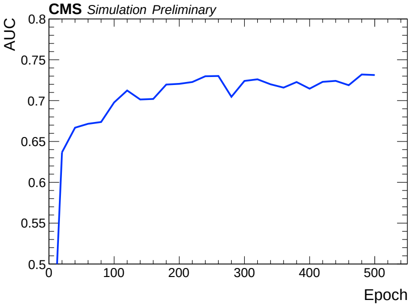


[1] First developments on Normalized Autoencoders in arXiv:2105.05735 and arXiv:2206.14225) with different loss function resulting in several failure modes, see backup slides 23-26

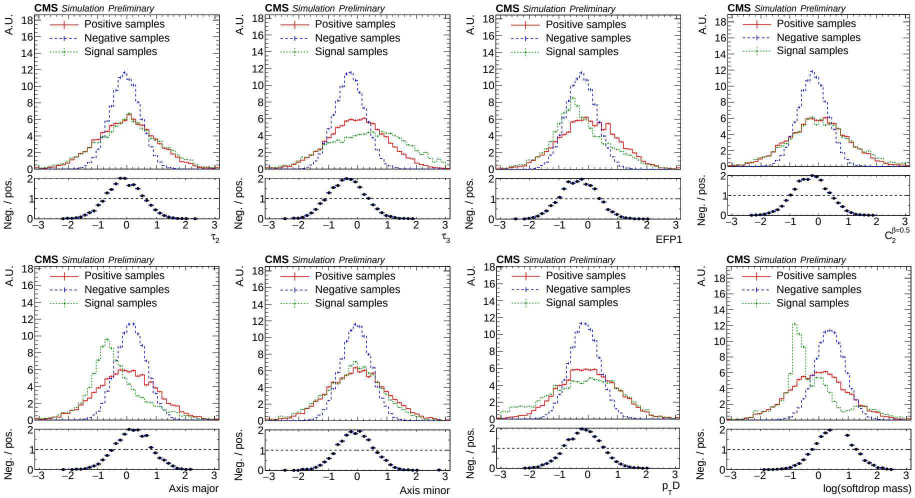
- Direct (anti-)correlation between Wasserstein distance and AUC!
- **Fully signal-agnostic training procedure**: training until minimal Wasserstein distance is achieved
- Drastic improvement over standard AEs!



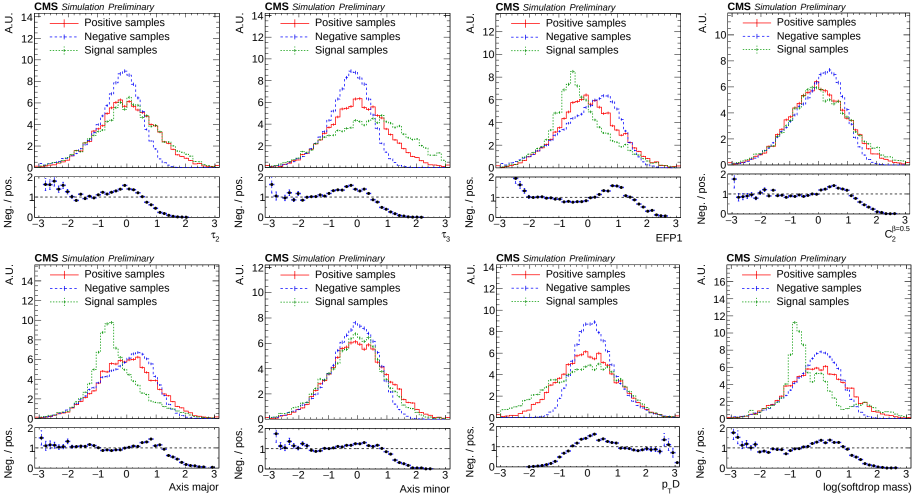




Epoch 1

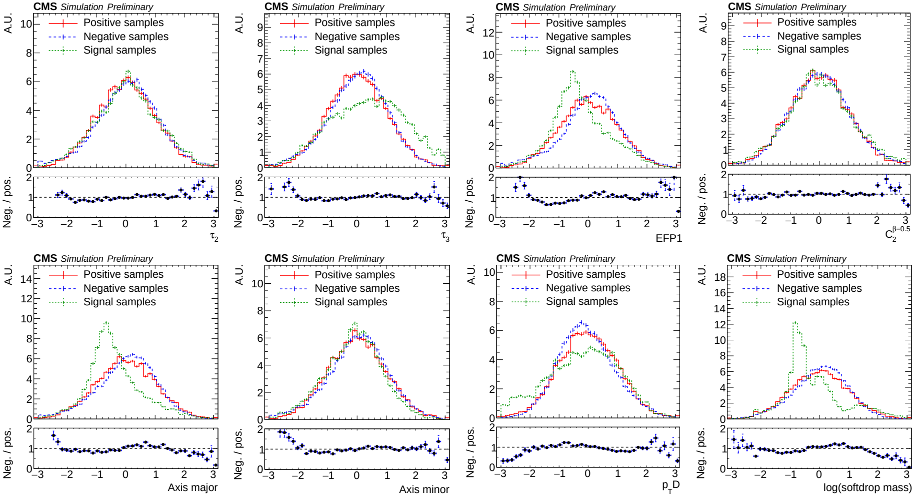

CMS Simulation Preliminary
Positive samples
Negative samples
Signal samples
A.U.
Neg. / pos.
τ_2
τ_3
EFP1
$C_2^{\beta=0.5}$
Axis major
Axis minor
$p_T D$
log(softdrop mass)

Epoch 40


CMS Simulation Preliminary
Positive samples
Negative samples
Signal samples
A.U.
Neg. / pos.
τ_2
τ_3
EFP1
$C_2^{\beta=0.5}$
Axis major
Axis minor
$p_T D$
log(softdrop mass)

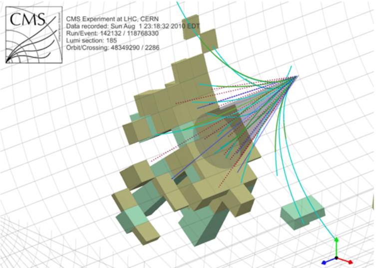
Learning the background probability

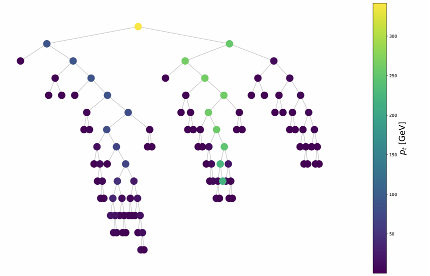
Epoch 500

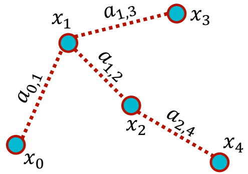


A more natural representation: graphs

- Reconstructed jets are unordered sets of particles
- Can naturally be represented as graphs!







$$X = (x_0, ..., x_N)$$

→ Node features

$$A = \begin{bmatrix} 1 & \dots & x_{0,N} \\ \vdots & \ddots & \vdots \\ a_{N,0} & \dots & 1 \end{bmatrix}$$

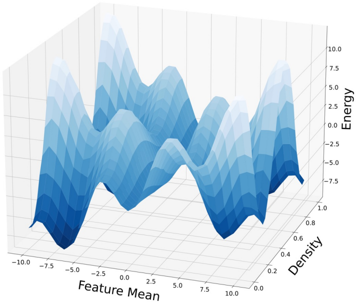
→ Adjacency matrix

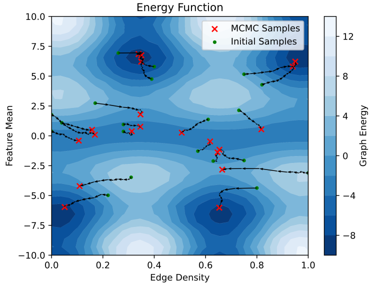
Towards normalized graph autoencoder

- Need to sample from p_θ in a graph space!
- Can run an MCMC on graphs:

$$X_n = X_{n-1} - \alpha \nabla E_\theta(X_{n-1}, A_{n-1}) + \beta \sigma_X$$

$$A_n = A_{n-1} - \gamma \nabla E_\theta(X_{n-1}, A_{n-1}) + \delta \sigma_A$$





➔ **Extends normalized autoencoders to graph networks!**

Conclusions

- Signal-agnostic searches for new physics in HEP can be implemented by learning a score that depends on the probability density of the SM data
- Standard AEs are prone to out-of-distribution reconstruction because they are free to minimize the reconstruction error outside the training phase space
- Normalized AEs propose a mechanism to ensure that the learned probability distribution matches that of the training data
- Wasserstein Normalized AEs is an improvement over Normalized AEs, based on the Wasserstein distance to minimize the distance between the AE probability distribution and that of the training data
- The Normalized AE paradigm can be extended to graph networks

Backup

1. Analysis

2. Normalized autoencoder (theory)

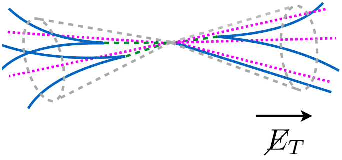
3. Normalized autoencoder (in practice)

- Dark quarks hadronize in the dark sector
- A fraction of dark hadrons promptly decays to SM quarks which hadronize in the SM sector
- Remaining dark hadrons are stable and invisible $\Longrightarrow$ DM candidates

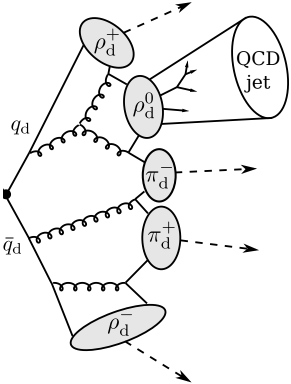
➔ Production of semivisible jets (SVJ) [arXiv:1503.00009, arXiv:1707.05326]

➔ **Different jet substructure due to double hadronization**

$$\not{E}_{\mathrm{T}} = \left|\left|\sum \vec{p_{\mathrm{T}}}\right|\right|$$

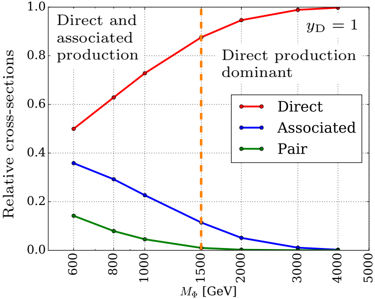


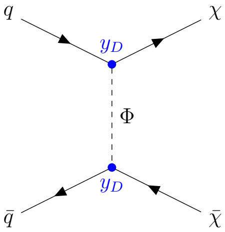
SM hadrons
Stable dark hadrons



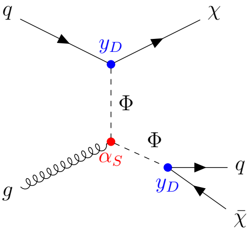
3 production mechanisms:

- **Direct production**: Production of dark quarks without resonance
- **Associated production**: Production of the mediator associated with a dark quark
- **Pair production**: Production of a pair of mediators

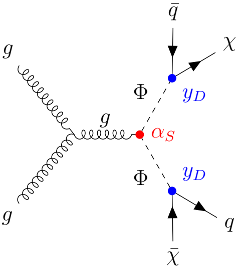




(a) Direct production



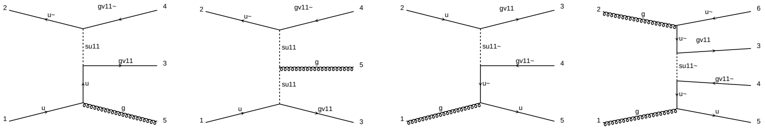
(b) Associated production



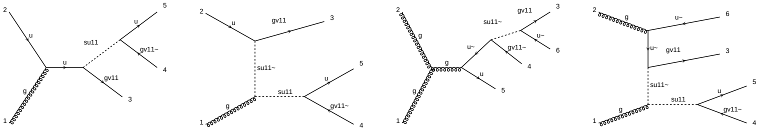
(c) Pair production

Many possible diagrams in the t-channel

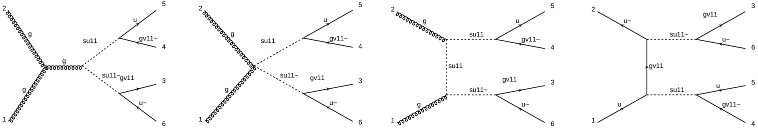
- Direct production



- Associated production



- Pair production



su11 is the mediator Φ, gv11 is a dark quark

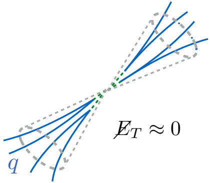
Model parameters

Model parameters:

- m_Φ: Mass of the mediator
- m_D: Mass of the dark hadrons (π_D, ρ_D)
  - Same for all dark hadrons
- y_D: Yukawa coupling between SM and dark quarks
- r_inv: Jet invisible fraction
  - Effective parameter in the simulation Branching ratio DM $\to q\bar{q}$

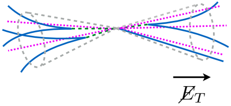
$$r_\mathrm{inv} = \left\langle \frac{\text{Number of stable dark hadrons}}{\text{Number of dark hadrons}} \right\rangle$$

$r_\mathrm{inv} = 0$



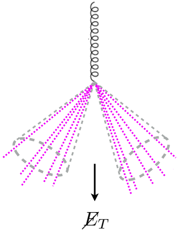
Dijet search

$0 < r_\mathrm{inv} < 1$



SVJ search

$r_\mathrm{inv} = 1$



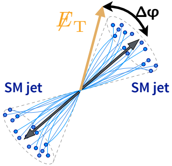
WIMP search

SM hadrons

Stable dark hadrons

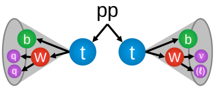
Backgrounds

SVJ experimental signature: $\not\!\!E_{\mathrm{T}}$ aligned with jets!



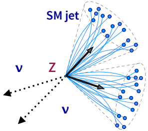
QCD multijet

- Artificial missing transverse energy $\not\!\!E_{\mathrm{T}}$ aligned with jet from jet energy mismeasurement
- Large cross-section



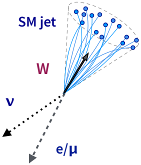
$\mathbf{t\bar{t}}$ + jets

- Large jet from boosted t
- Semi-leptonic channel $W(\rightarrow l\nu)$ with lost lepton, genuine $\not\!\!E_{\mathrm{T}}$ from neutrino



Z + jets

- Genuine $\not\!\!E_{\mathrm{T}}$ from $Z \rightarrow \nu\nu$



W + jets

- $W \rightarrow l\nu$ with lost/not reconstructed lepton or hadronic decay of τ
- Genuine $\not\!\!E_{\mathrm{T}}$ from neutrino

Energy-based models (EMBs)

- EBMs are models where the probability is defined through the Boltzmann distribution
- Let θ denote the model parameters
- The model probability p_θ is defined from the energy E_θ

$$p_\theta(x) = \frac{1}{\Omega_\theta} \exp\left(-E_\theta(x)/T\right) \tag{1}$$

where the normalization constant Ω_θ is

$$\Omega_\theta = \int \exp\left(-E_\theta(x)/T\right) dx \tag{2}$$

- The EBM loss for a training example x is the negative log-likelihood:

$$L_\theta(x) = -\log p_\theta(x) = E_\theta(x)/T + \log \Omega_\theta \tag{3}$$

- The gradient of the EBM loss is thus:

$$\nabla_\theta L_\theta(x) = \nabla_\theta E_\theta(x) - \mathbb{E}_{x' \sim p_\theta}\left[\nabla_\theta E_\theta(x')\right] \tag{4}$$

- The expectation value over the training dataset, with probability p_{data} is:

$$\mathbb{E}_{x \sim p_{\mathrm{data}}}\left[\nabla_\theta L_\theta(x)\right] = \mathbb{E}_{x \sim p_{\mathrm{data}}}\left[\nabla_\theta E_\theta(x)\right] - \mathbb{E}_{x' \sim p_\theta}\left[\nabla_\theta E_\theta(x')\right] \tag{5}$$

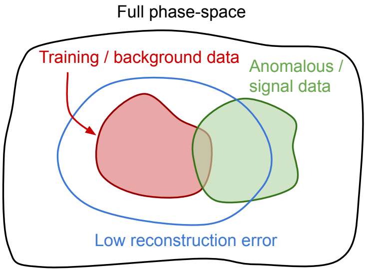
Normalized Autoencoder (NAE) paradigm

- **Ensure that the low reconstruction error probability distribution matches that of training data**

➔ Need a way to sample from the low reco error probability, independent of the training dataset

- The network probability distribution p_θ is constructed from the reco error E_θ via the Boltzmann distribution[1]:

$$p_\theta(x) = \frac{1}{\Omega_\theta} \exp\left(-E_\theta(x)\right)$$



- Low reco error probability distribution sampled via Langevin Markov Chain Monte Carlo (MCMC)[2] to obtain "negative examples" and compute their reconstruction error E_-
- The positive energy E_+ is the reconstruction error of the training ("positive") examples
- The loss is designed to learn $p_\theta = p_{\text{data}}$:

$$\mathbb{E}_{x\sim p_{\text{data}}}\left[L_\theta(x)\right] = \underbrace{\mathbb{E}_{x\sim p_{\text{data}}}\left[E_\theta(x)\right]}_{\text{positive energy } E_+} - \underbrace{\mathbb{E}_{x'\sim p_\theta}\left[E_\theta(x')\right]}_{\text{negative energy } E_-}$$

[1] More on Energy Based Models in backup slide 9

[2] More on MCMC in backup slide 13

Loss

$$\mathbb{E}_{x\sim p_{\text{data}}}[L_\theta(x)] = \underbrace{\mathbb{E}_{x\sim p_{\text{data}}}[E_\theta(x)]}_{\text{positive energy } E_+} - \underbrace{\mathbb{E}_{x'\sim p_\theta}[E_\theta(x')]}_{\text{negative energy } E_-}$$

Positive energy

- Simply the reconstruction error over the training dataset
- Take examples from training dataset and compute the reconstruction error!

Negative energy

- Reconstruction error of the "negative samples" x' from the probability distribution p_θ
- Need to sample from the model to get the "negative samples"

➔ Monte Carlo Markov Chain (MCMC) employed

MCMC

- Start from an initial point x'_0
- Run n Langevin MCMC steps:

$$x'_{i+1} = x'_i - \underbrace{\lambda_i \nabla_x E_\theta(x'_i)}_{\text{drift}} + \underbrace{\sigma_i \epsilon}_{\text{diffusion}} \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

- Repeat with several points $x_0'^{(j)}$, the negative samples are the $x_n'^{(j)}$

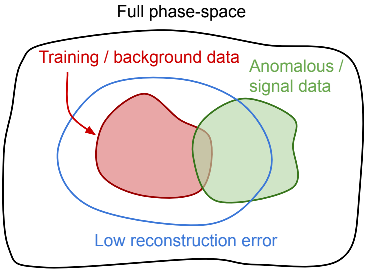
Wasserstein Normalized Autoencoder (WNAE) paradigm

- **Ensure that the low reconstruction error probability distribution matches that of the training data**

➔ Need to sample from the low reco error probability distribution, independent from the training dataset

- The network probability distribution p_θ is constructed from the reco error E_θ via the Boltzmann distribution[1]:

$$p_\theta(x) = \frac{1}{\Omega_\theta} \exp\left(-E_\theta(x)\right)$$



- Low reco error probability distribution sampled via Langevin Markov Chain Monte Carlo (MCMC)[2] to obtain "negative examples"[3]

- The loss is the Wasserstein distance (a.k.a. Energy Mover's Distance) between negative examples and training examples to learn $p_\theta = p_{\text{data}}$:

$$L_\theta(x) = \inf_{\gamma \in \Pi(p_{\text{data}}, p_\theta)} \mathbb{E}_{(x,x')\sim\gamma}[\|x - x'\|]$$

- The WNAE learns the probability distribution of the training data

[1] More on Energy Based Models in backup slide 9

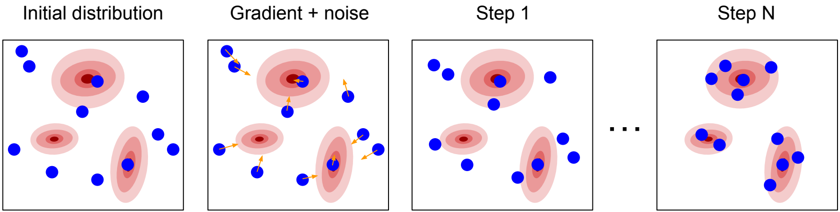
[2] More on MCMC in backup slide 13

Principle of MCMC (Langevin Monte Carlo)

- Let p be a probability distribution on $\mathbb{R}^d$
- Consider x_0 a random initial set of n points in $\mathbb{R}^d$
- With the update rule:

$$x_{t+1} = x_t + \lambda \nabla \log\left(p(x_t)\right) + \sqrt{2 \cdot \lambda} \cdot \epsilon_t$$

  where ϵ_t is a sample of n points drawn from a multivariate normal distribution on $\mathbb{R}^d$
- Let ρ_t denote the probability distribution of x_t
- In the limit $t \to \infty$, ρ_t approaches a stationary distribution ρ_∞, and $\rho_\infty = p$



Understanding the MCMC hyper-parameters

- Recall the MCMC equation:

$$x'_{i+1} = x'_i - \lambda \nabla_x E_\theta(x'_i) + \sigma\epsilon \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

- A theoretically motivated choice[1] for the MCMC hyper-parameters is:

$$2 \cdot \lambda = \sigma^2$$

- The MCMC is run on every batch: in practice, for training in a reasonable amount of time, the MCMC is rather short

- To speed up the convergence of the MCMC, the temperature T is introduced:

$$x'_{i+1} = x'_i - \frac{\lambda}{T} \nabla_x E_\theta(x'_i) + \sigma\epsilon \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

- Tweaking the gradient step size can be seen as adjusting the temperature T: the strength of the gradient term is increased for $T < 1$

- The parameter space where σ and T are set independently, with $T < 1$ and $\lambda = \sigma^2/2$ is in theory a good region

[1] For an infinitely long chain, see backup slide 13

MCMC initialization:

- In theory, MCMC convergence independent on the initial point
- However, in practice with short chain, initialization is crucial

Several commonly used initialization algorithms of the MCMC:

- Contrastive Divergence[1] (CD)
- Persistent CD[2] (PCD)

CD[3]

- Initial distribution from training data
- Re-initialization after each parameter update (*i.e.* epoch)

PCD[4]

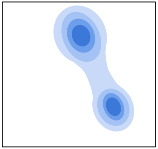
- Random initial distribution for first MCMC
- The model changes only slightly during parameter update
- Thus, for subsequent chains, initialize chain at the state in which it ended for the previous model
- Possibility to randomly re-initialize a small fraction of the samples

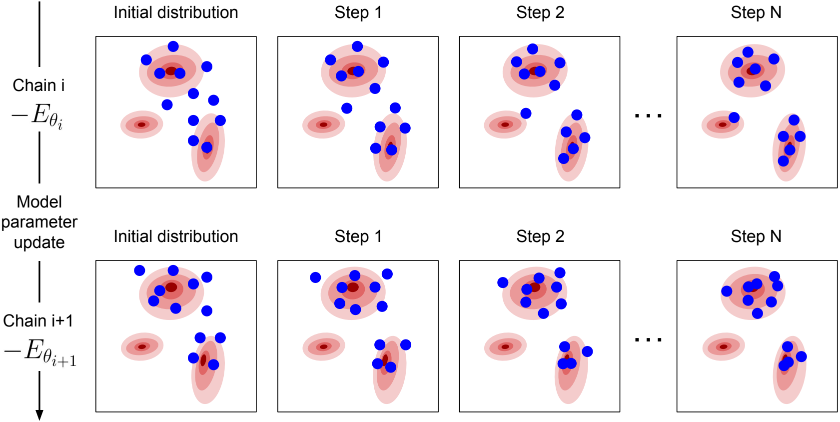
[1] Neural Comput 2002; 14 (8)
[2] PCD paper
[3] Illustration in backup slide 16
[4] Illustration in backup slide 17

Principle of CD

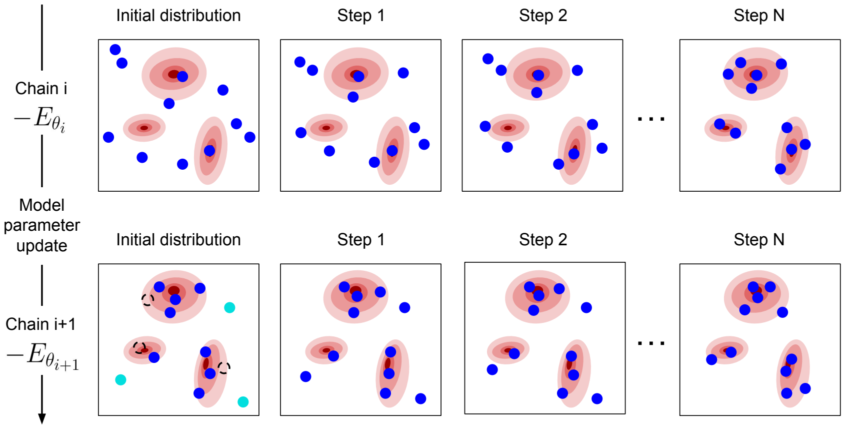
Example of a failure mode of CD: High probability mode far from training data distribution is not sampled

Training data distribution:



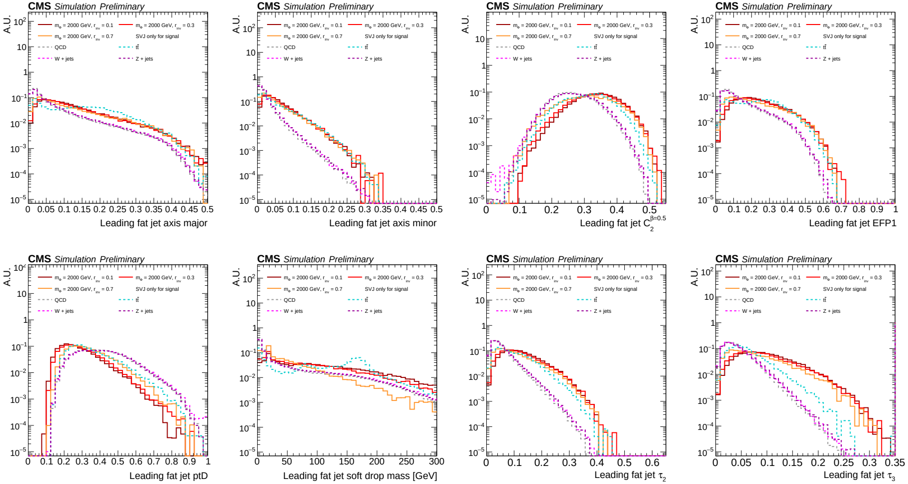


Principle of PCD

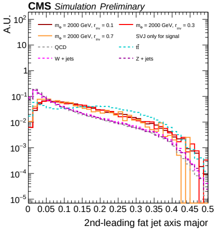


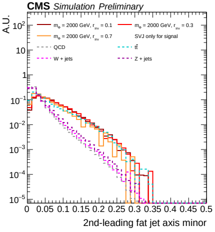
1. Analysis
2. Normalized autoencoder (theory)
3. Normalized autoencoder (in practice)

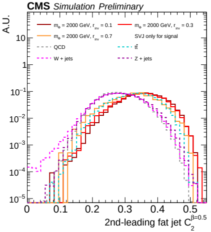
Input features

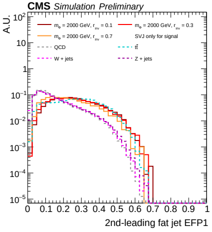


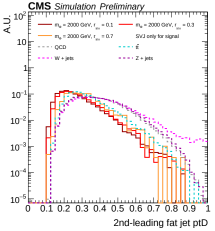
Input features

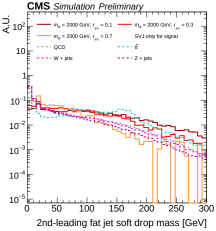


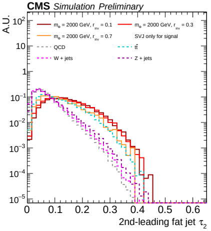


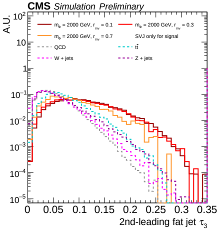






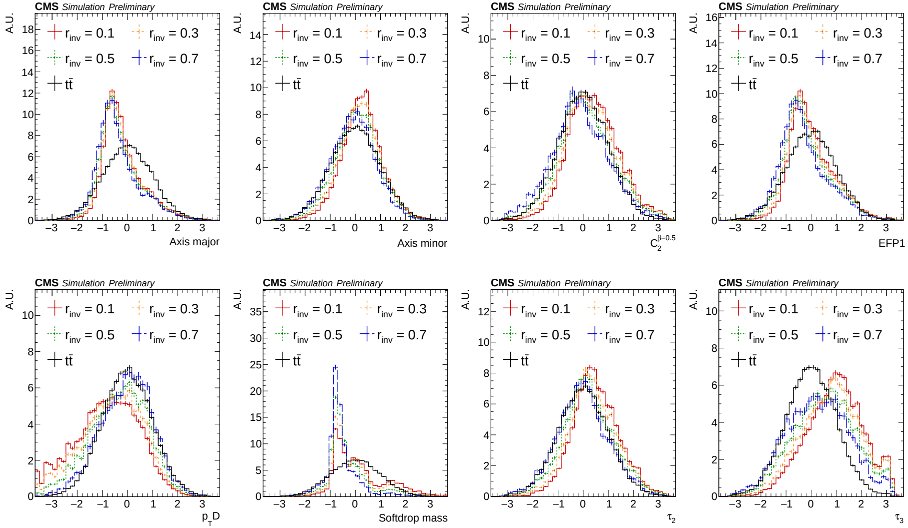






Input features ($t\bar{t}$ + jets trainings)

- Input features used for the training on top jets at pre-selection level
- Leading two jets
- Truth-tagged SVJ only for signals



Training samples and hyper-parameters

Input features
Using AK8 jets because SVJ are expected to be wide

Jet width	Axis major axis minor
N-pronginess	τ_2, τ_3 $C_2^{\beta=0.5}$
Other	p_T^D, EFP1 log(softdrop mass)

Architecture
Fully connected neural net
Hidden layers: $10, 10, 6, 10, 10$

Hyper-parameters

Hyper-parameter	Value
Batch size	256
Reconstruction loss	MSE
Activation	ReLU
Output encoder/ decoder activation	Linear
Optimizer	Adam
Learning rate	1e-5
Dropout	0.
MCMC	PCD
Sampling phase space	[-3, 3] hypercube

Number of events

m_Φ [GeV]	1000	1500	2000				3000	4000	QCD	$t\bar{t}$
r_{inv}	0.3	0.3	0.1	0.5	0.3	0.7	0.3	0.3		
Number of events	23k	25k	23k	18k	16k	11k	14k	14k	83k	23k

Number of AK8 jets

Background jets	Leading 2 jets
Signal jets	Only SVJ in leading 2 jets

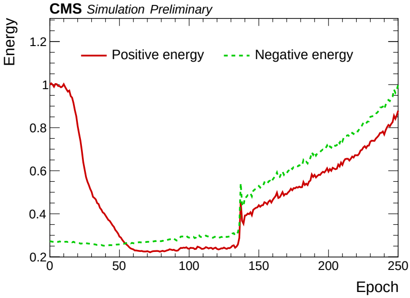
Train/validation/test splitting

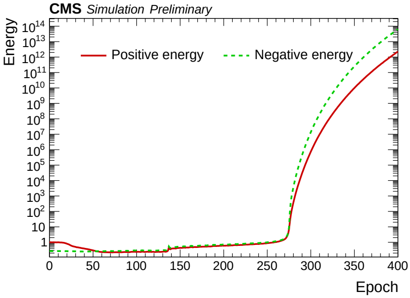
0.7/0.15/0.15

Failure modes of NAEs

Observed two failure modes when training a NAE:

- **Negative energy difference**: the loss function can be < 0
  - ➔ $p_\theta = p_{\text{data}} \implies L = 0$
  - ➔ $L \neq 0 \implies p_\theta \neq p_{\text{data}}$!
  - ➔ Incentive to learn $p_\theta \neq p_{\text{data}}$ as it has lower loss ($L < 0$) than $p_\theta = p_{\text{data}}$ ($L = 0$)
- **Divergence of energies**

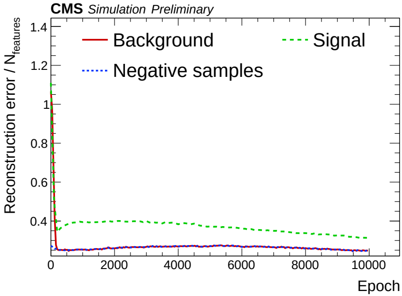


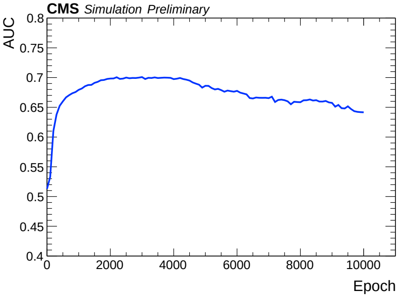


Modified default loss function, compared to arXiv:2105.05735, to:

- discourage the network to converge to negative energy difference configurations
- prevent the divergence of the energies

$$L = \log\left(\cosh\left(E_+ - E_-\right)\right)$$

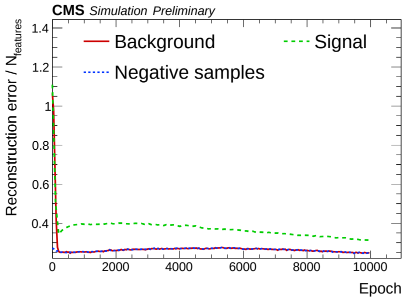


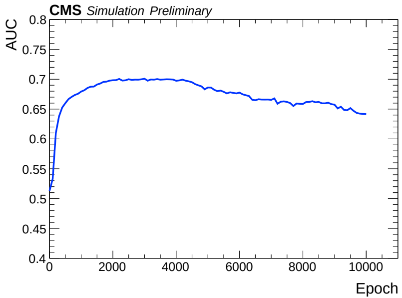


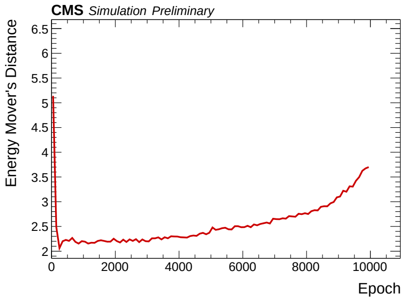
➔ **Signal SVJ reconstruction is efficiently suppressed!**

➔ **How to define stopping condition in a fully signal-agnostic way?**

Wasserstein distance: signal-agnostic metric for optimal performance







- **The Wasserstein ditance (a.k.a. Energy Mover's Distance, EMD) between the training and negative samples is a measure of the distance between the background and NN probabilities directly in the input feature space**
- Always observing a "collapse": the energy difference stays zero but background and NN probabilities differentiate

EMD versus energy difference: illustration of the collapse

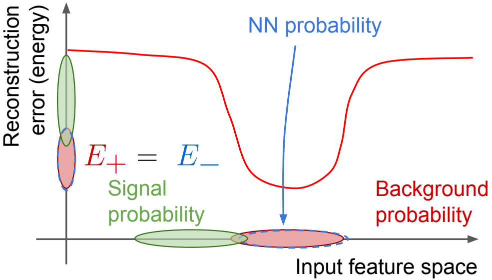


Illustration before collapse:

- Background (positive) and NN (negative) probability distributions match

➔ **Low EMD and low energy difference** between negative and positive probability distributions

➔ Anomalies have large reco error

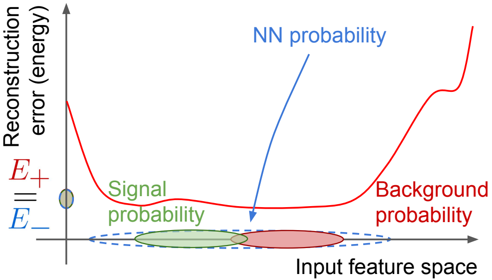


Illustration after collapse:

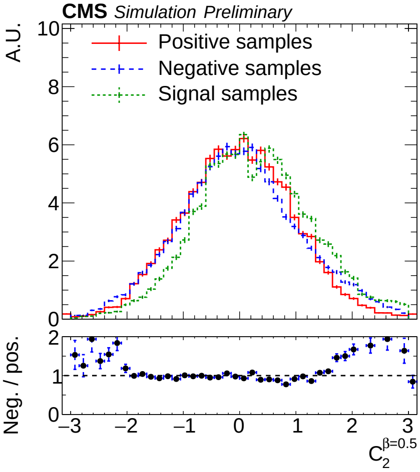
- Large discrepancy between background and NN probability distributions

➔ **Large EMD but low energy difference** between negative and positive probability distributions

➔ Anomalies are not distinguishable from background

- Can visualize negative samples as 1D histograms in the feature space!
- $C_2^{\beta=0.5}$ negative samples distribution is wider and offset after the collapse

Before phase space collapse



After phase space collapse