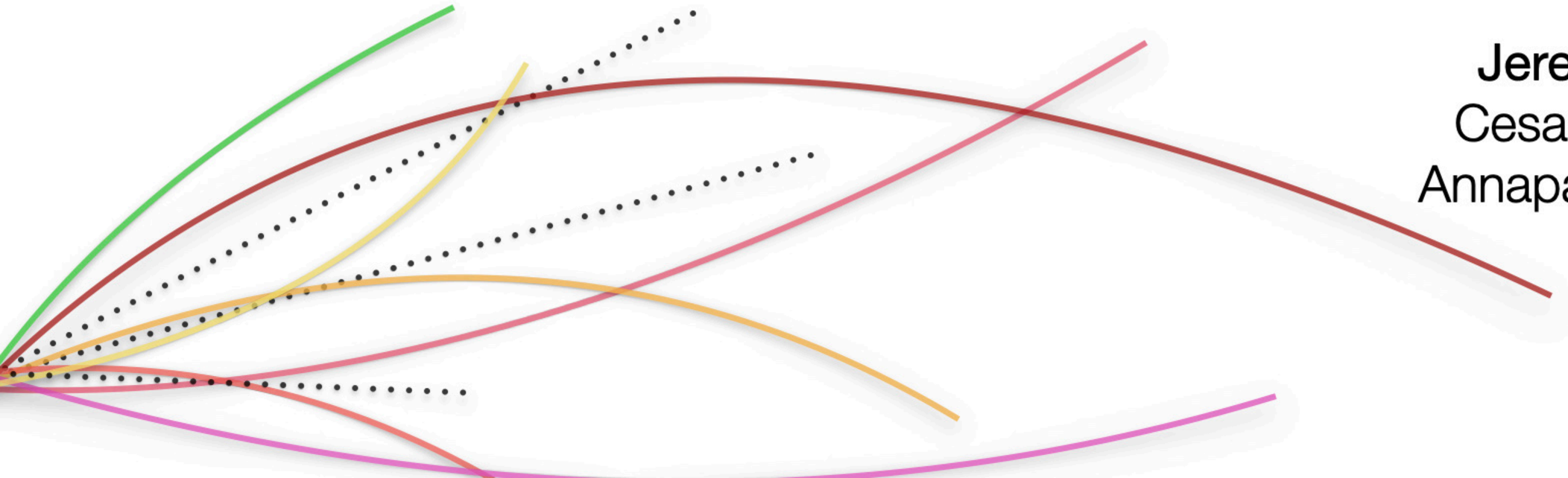
ETH zürich

# AUTOENCODERS
## FOR ANOMALOUS JET SEARCHES



**Jeremi Niedziela**
Cesare Cazzaniga
Annapaola de Cosa
Florian Eble

# Outline

1. Introduction
   - semivisible jets
   - signature & challenges

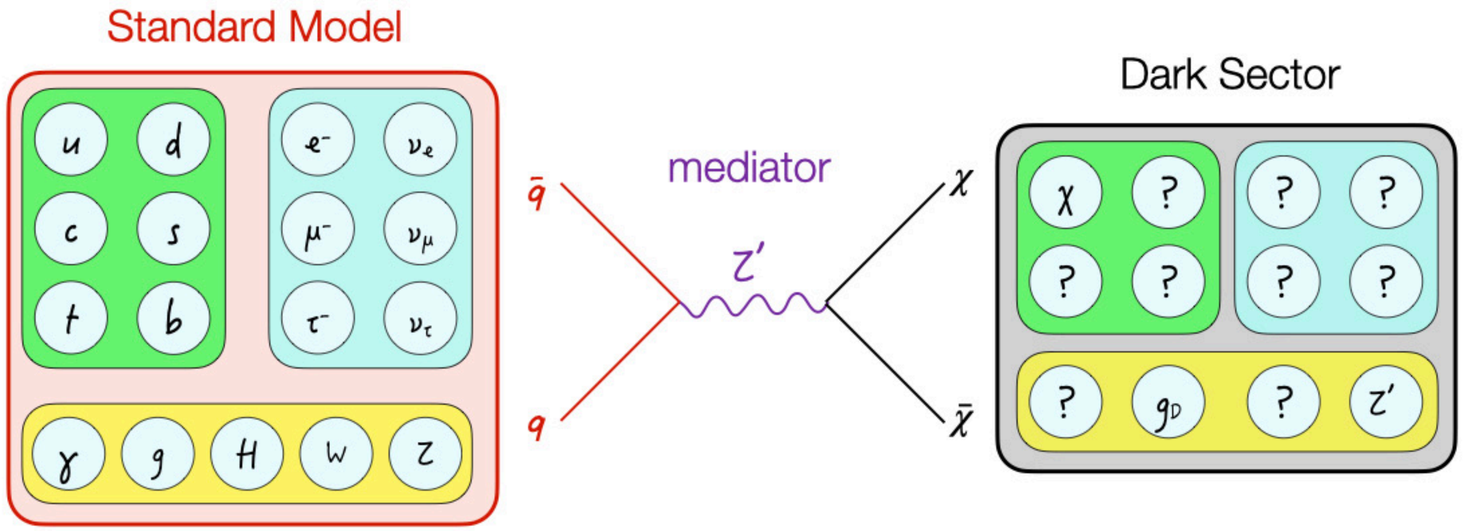
2. Anomalous jet tagging
   - Autoencoders,
   - performance against QCD and $t\bar{t}$,
   - Graph Autoencoders
   - other signatures.

3. Summary

# INTRODUCTION

## Hidden Valley

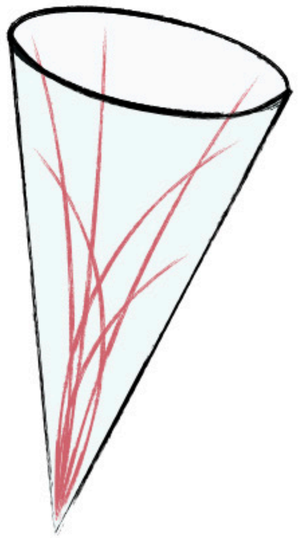
- Dark Matter (DM) is part of a larger **dark sector** containing **many particles & interactions**,
- dark sector is connected with the Standard Model (SM) through a **mediator Z'**,
- one of the dark interactions could be **similar to the SM strong interaction**
  → dark sector quarks fragment to form **jets.**



# INTRODUCTION

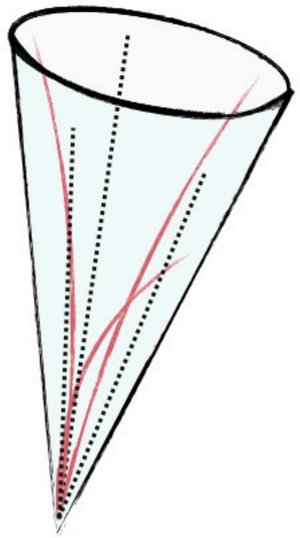
## Semivisible Jets (SVJ)

- a fraction of constituents decays back to **SM hadrons**, others are stable (invisible) **Dark Matter particles**,
- **internal structure of the jet** depends on model parameters:
  - $\alpha_{Dark}$ — coupling constant of the Dark Sector strong interaction equivalent,
  - $m_{Dark}$ — mass of the dark hadron,
  - $r_{inv}$ — fraction of invisible particles inside a jet,
- jet kinematics also affected by the Z' mass $m_{Z'}$.

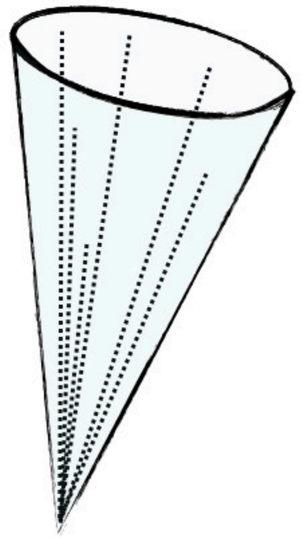


$r_{inv} = 0.0$

(di-jet like search)



$r_{inv} = 0.5$



$r_{inv} = 1.0$

(WIMP like search)

Standard Model

Dark Sector

# SVJ — SIGNATURE

## Focus on internal jet structure

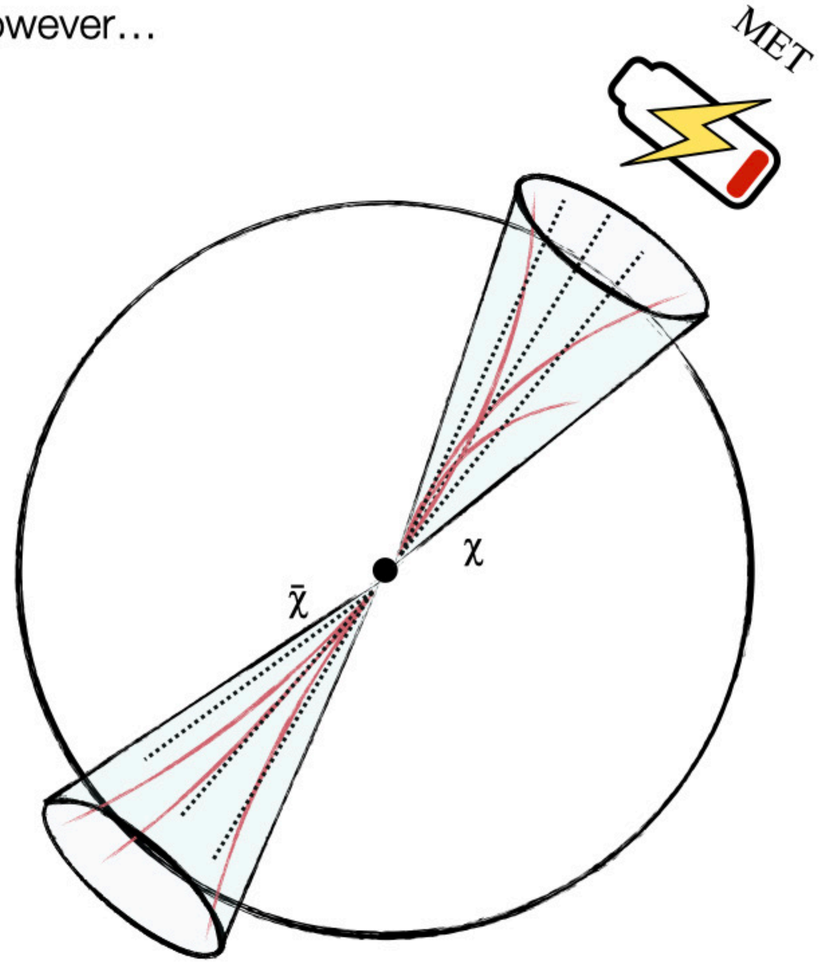
- event-level features, such as **di-jet $m_T$ or MET $p_T$** can be exploited to some extent. However…
- bump hunt not possible in the **t-channel**,
- the true discriminating power comes from **different shower dynamics**,
- to fully exploit this type of models we should look at the **internal structure of jets**.

## MET aligned with jet

- MET mainly results from **mismeasurement** of jet energy, or **detector failure** in certain region,
- $\Delta\phi$(MET, jet) $\approx 0$ → **most analyses reject** such events,
- for us, that's the **region of interest**.

## Other challenges

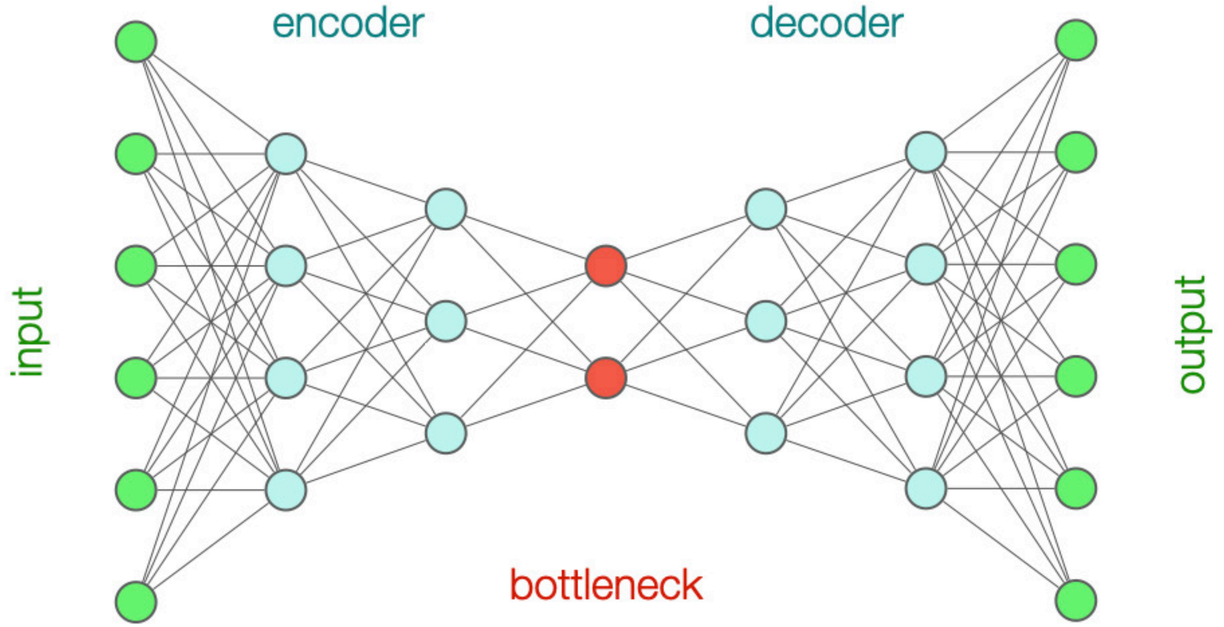
- signal models → **unknown theory parameters**,
- imperfect **background modeling**,
- imperfect **detector simulation**.



# AUTOENCODERS

## Autoencoder (AE)

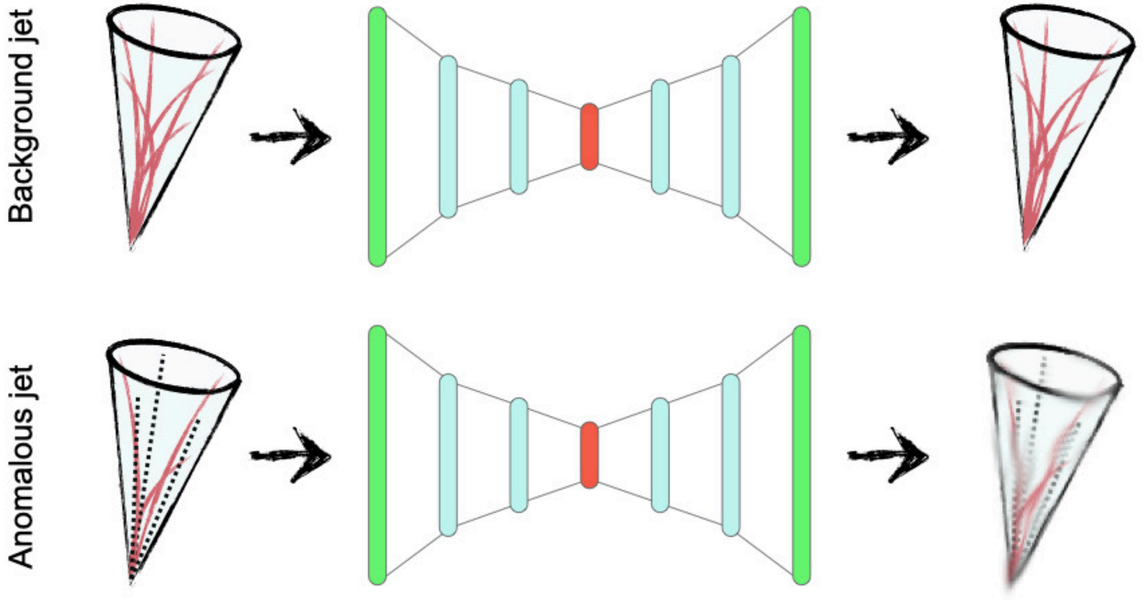
- **unsupervised** neural network,
- architecture: **encoder** → **bottleneck** (latent space) → **decoder**,
- trained to **minimize the difference** (loss) between output and input,
- bottleneck: network has to learn how to **compress** and then **de-compress** input information,
- great tool for **anomaly detection** - when presented with something that it didn't see during the training, **loss is exceptionally high**.

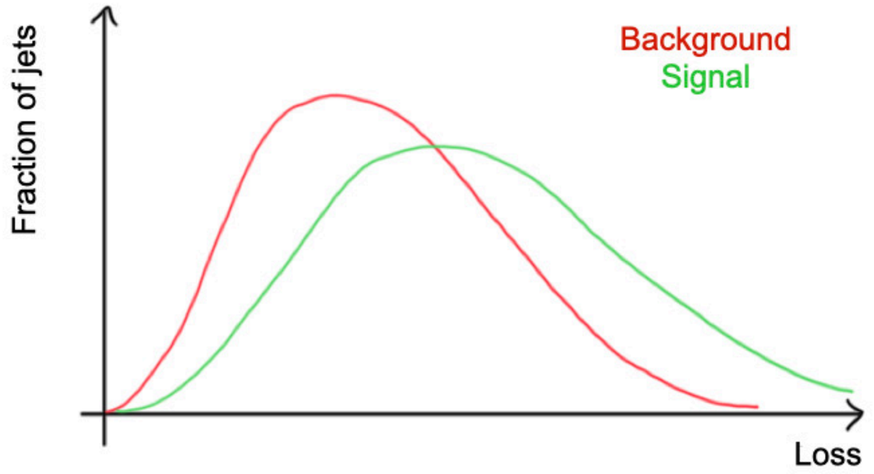


# AUTOENCODERS FOR ANOMALOUS JETS

## AE for anomalous jets

- **train** on background jets → networks learns to **minimize loss**,
- the model struggles to reconstruct **signal jets** → **high loss**.




## Benefits of AE

- **training on background only,**
- can be **trained on collision data,**
  - → largely **independent from signal hypothesis,**
  - → independent from **background modeling,**
  - → independent from **detector simulation.**

# SAMPLES & INPUT FEATURES

## Samples

- QCD and SVJ signal generated with **Pythia,**
- reconstructed with **Delphes** with CMS settings,
- applied **simple preselection** to:
  - mimic the trigger,
  - reduce QCD background.

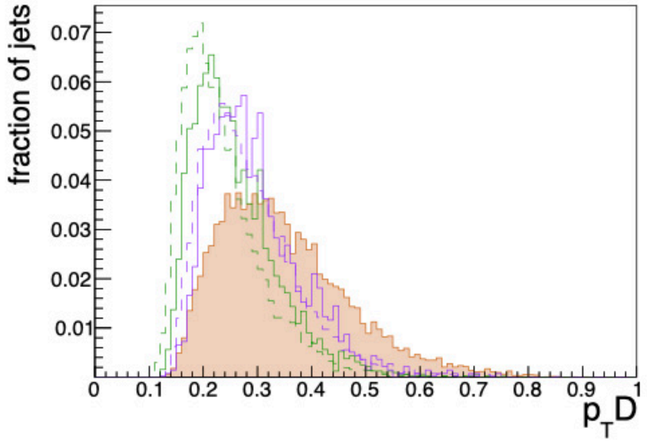
## Input features

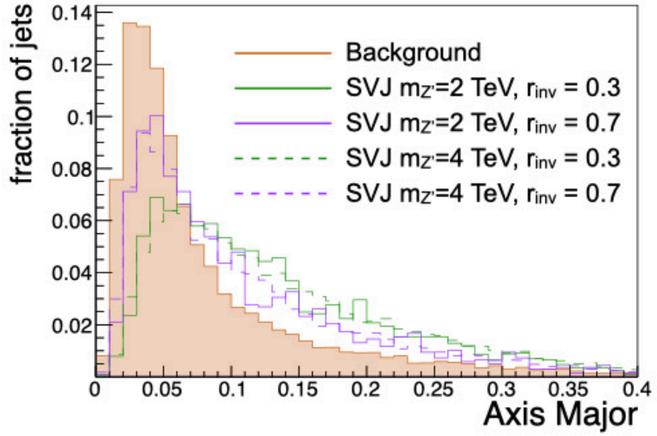
- good **signal-background discrimination,**
- **small correlation** with each other,
- **selected jet features:**
  - η, ϕ → information about detector effects,
  - jet mass,
  - substructure variables: $p_T D$, jet axes, Energy Flow Polynomials ($EFP_1$), Energy Correlation Functions ($C_2$, $D_2$).

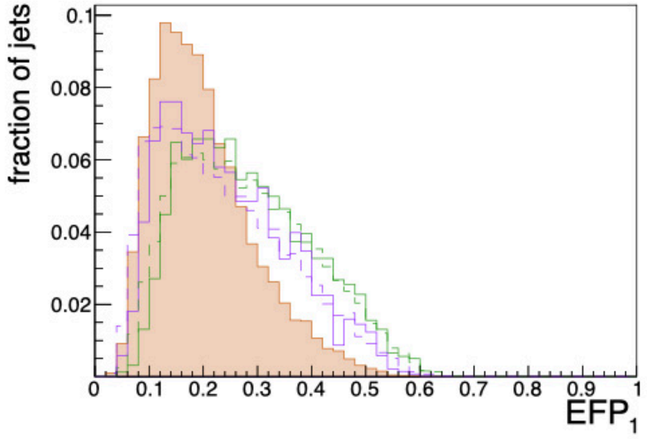
doi.org/10.1007/JHEP02(2022)074

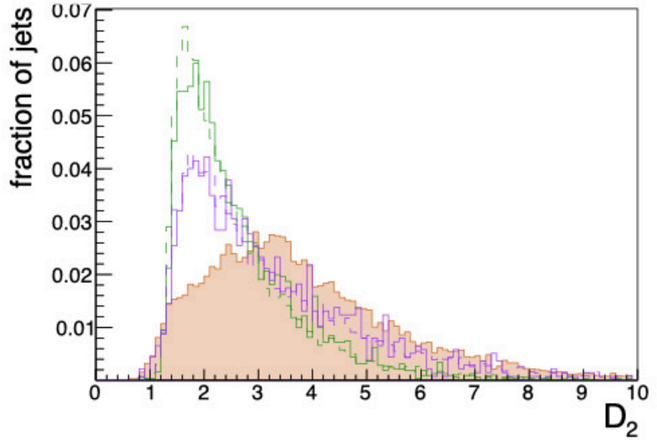
*Autoencoders for semivisible jet detection*

F. Canelli, A. de Cosa, J. Niedziela, K. Pedro, M. Pierini, L. Le Pottier



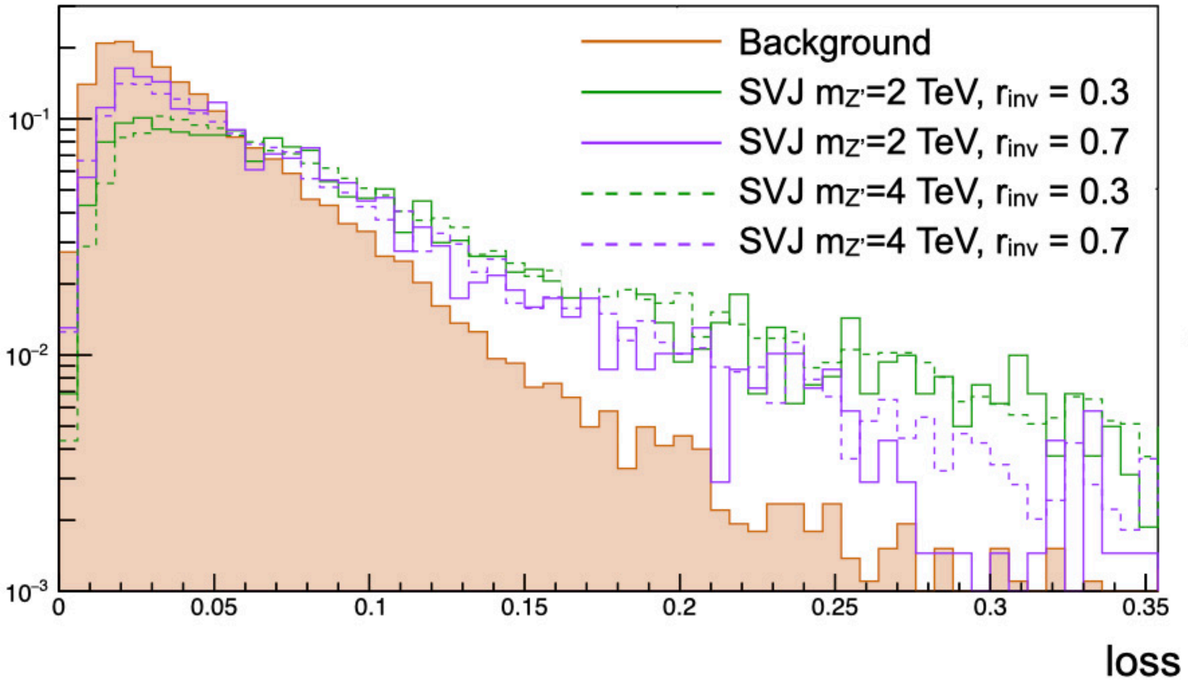



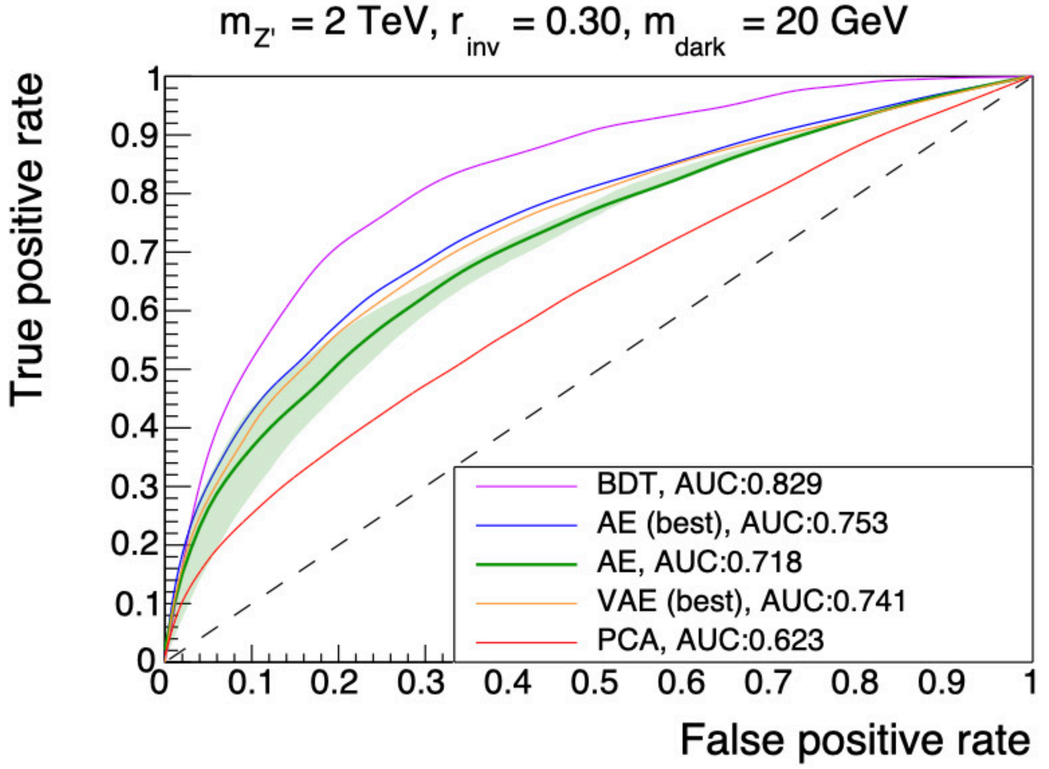
# AE — ANOMALY DETECTION CAPABILITIES

## Anomaly detection capabilities

doi.org/10.1007/JHEP02(2022)074

- losses of **QCD jets smaller** than for SVJ → the method works!
- allows to **discriminate** between background and signal, despite AE had **no prior knowledge of the signal**!



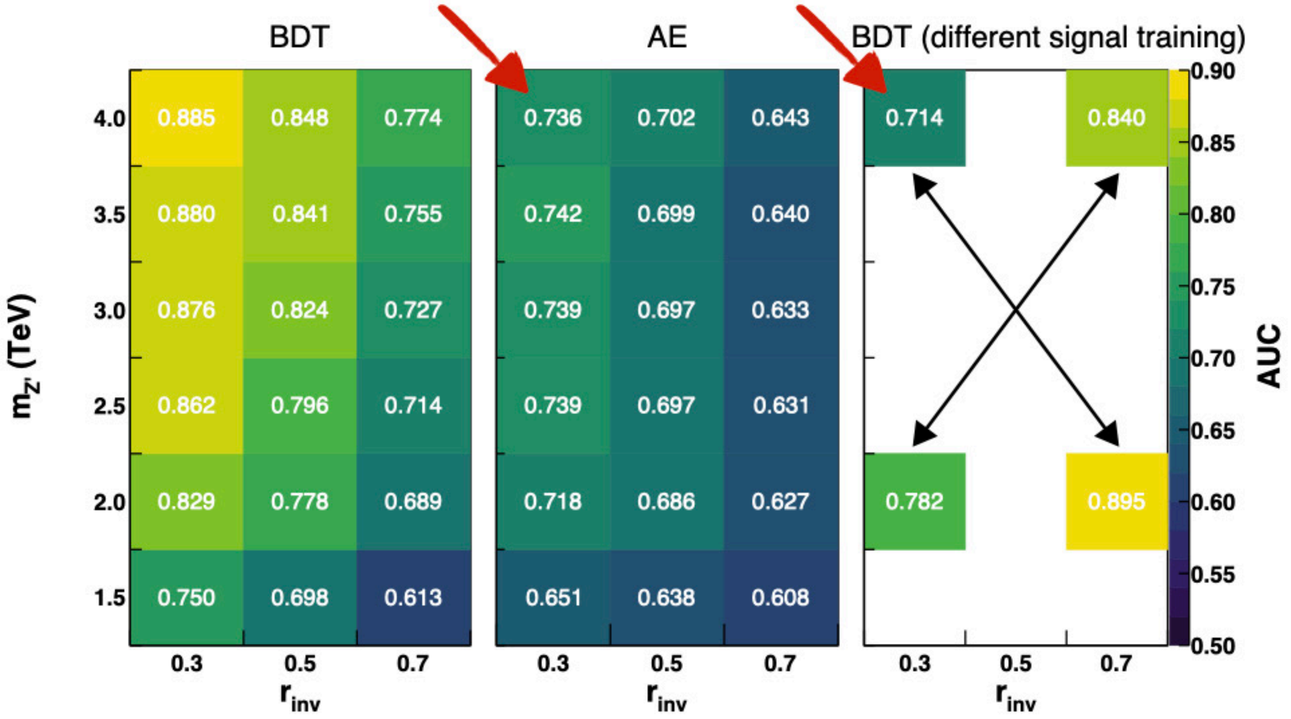



# AE vs. BDT — Performance

doi.org/10.1007/JHEP02(2022)074

## Comparison with a classifier

- obtained **good performance** for the AE,
- **AE can outperform a BDT** trained on a **wrong signal hypothesis**,
  - swapping corners in $m_{Z'}$—$r_{inv}$ plane,
  - training on $m_{Dark}$ = 20 GeV, testing on other masses,
- **results very encouraging**.

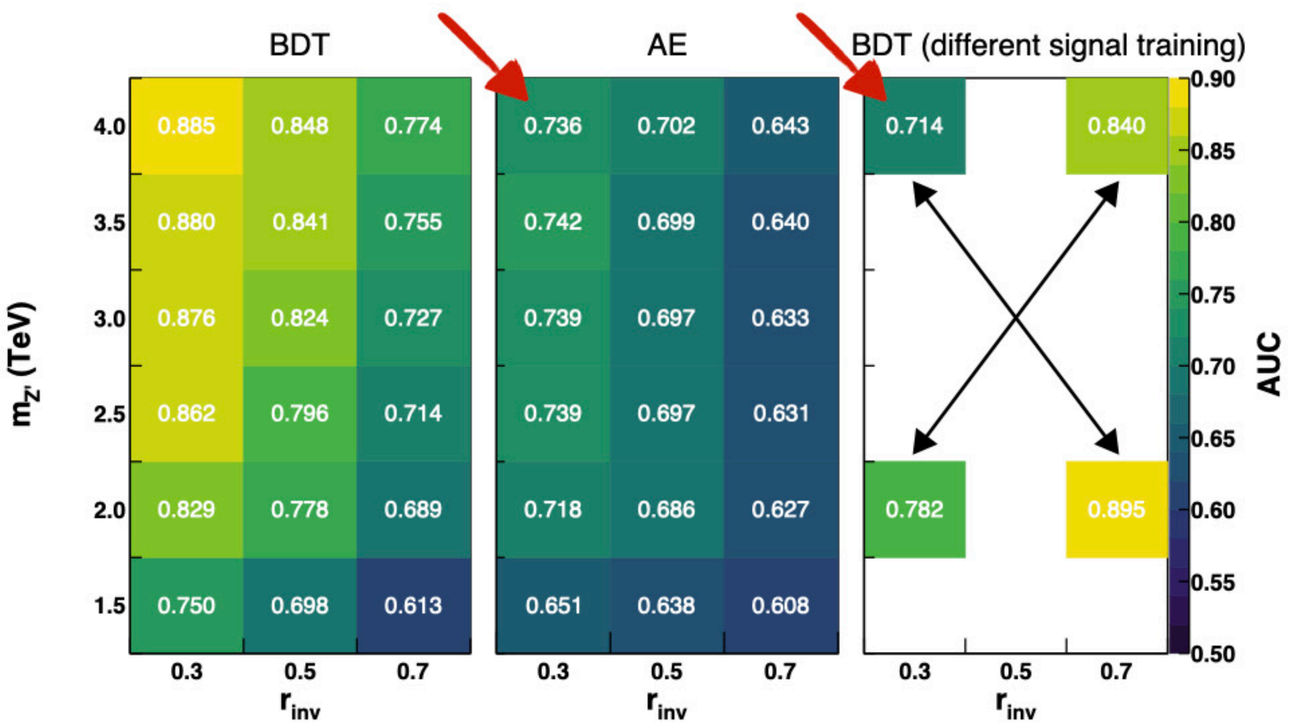


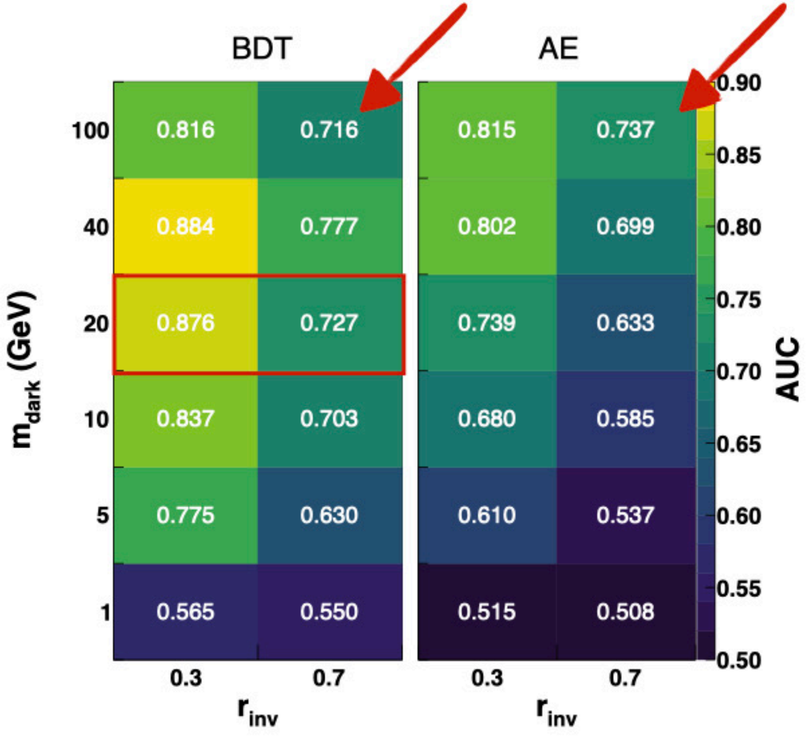
# AE vs. BDT — Performance

## Comparison with a classifier

doi.org/10.1007/JHEP02(2022)074

- obtained **good performance** for the AE,
- **AE can outperform a BDT** trained on a **wrong signal hypothesis**,
  - swapping corners in $m_{Z'}$—$r_{inv}$ plane,
  - training on $m_{Dark}$ = 20 GeV, testing on other masses,
- **results very encouraging.**

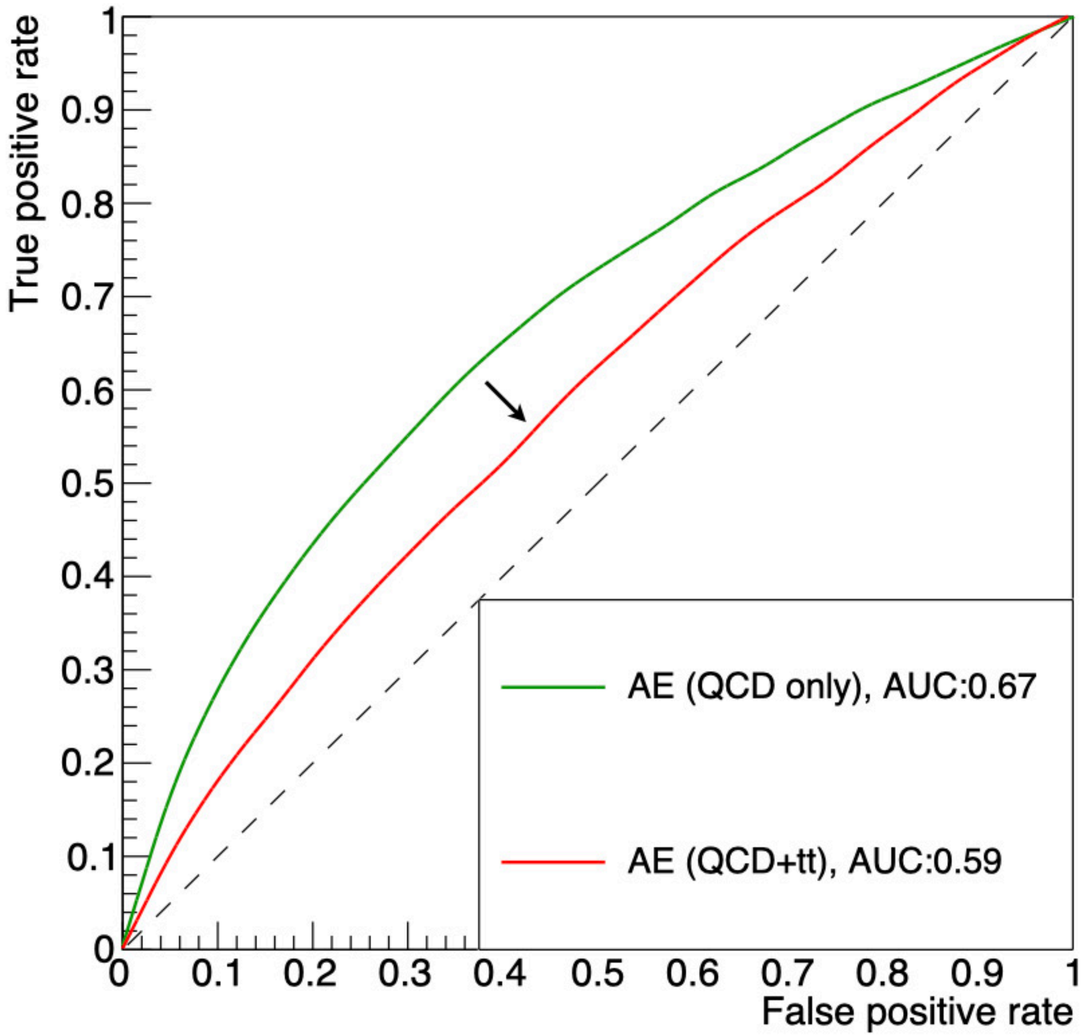


# AE — Adding $t\bar{t}$ background

## Adding $t\bar{t}$ background

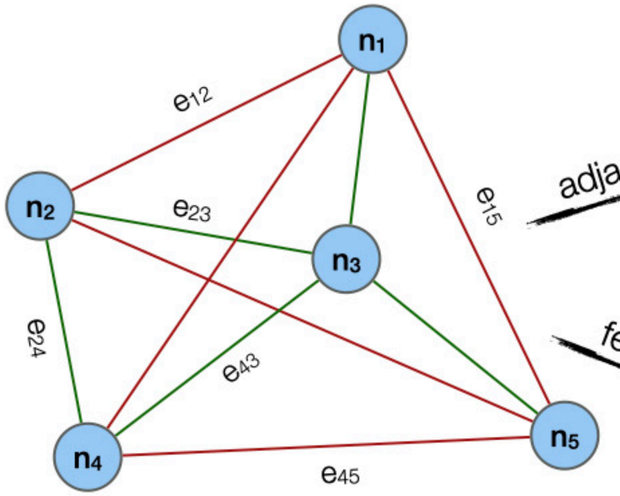
- one of the possible backgrounds for SVJ is $t\bar{t}$,
- input parameters **re-optimized** for a combination of **QCD and $t\bar{t}$**:
  - $EFP_1 \rightarrow \tau_{21}$,
  - $C_2^{\beta=0.5} \rightarrow C_2^{\beta=1.0}$,
  - $D_2^{\beta=0.5} \rightarrow D_2^{\beta=1.0}$,
- despite re-optimization, performance of the **AE dropped significantly**,
- **AE** is based on **substructure variables**:
  - sensitive to a **subset of interesting relations** between constituents,
  - designed for **SM jets discrimination**,
  - more discussion: Semivisible Jets Workshop 2022,
- further **optimization of input** features should be possible, but…
- instead, one could work with **jet constituents** directly.



# GRAPH AUTOENCODERS

## Graph Autoencoders (GAE)

- the network takes two inputs:
  - adjacency matrix (AM) → describes edges of the graph,
  - features matrix (FM) → features of the nodes,



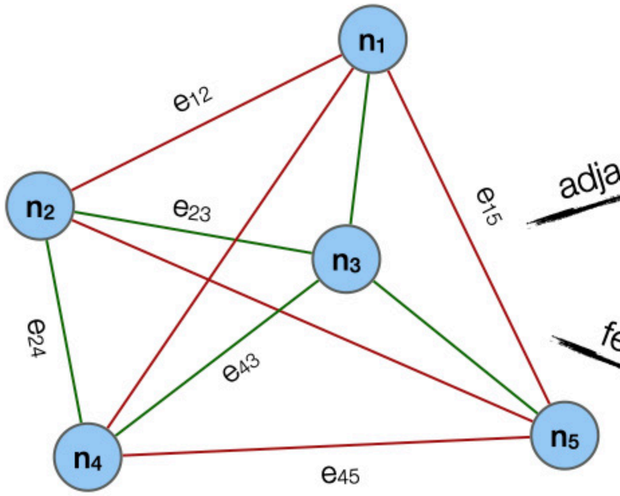

| n | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 1 | 1 | 1 | 0 |
| 3 | 1 | 1 | 1 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 0 |
| 5 | 0 | 0 | 1 | 0 | 1 |

| n | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $f_1$ | 3.2 | 0.5 | -2.3 | 6.2 | 2.1 |
| $f_2$ | -1 | 1 | 1 | 1 | -1 |
| $f_3$ | 23 | 41 | 9 | 18 | 28 |

# GRAPH AUTOENCODERS

## Graph Autoencoders (GAE)

- the network takes two inputs:
  - **adjacency matrix (AM) → describes edges of the graph,**
  - **features matrix (FM) → features of the nodes,**
- **AM** is provided to **each layer** and convoluted with the output of the previous layer (with **decreasing kernel size**),
- decoder is very simple and attempts to **reconstruct the AM**,
- **other implementations possible**, e.g. reconstruction of FM instead of AM, or both.

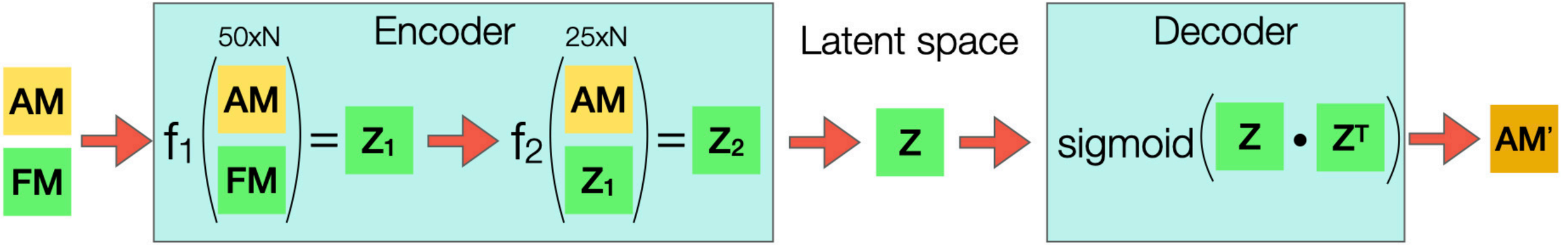



adjacency matrix (edges)

| n | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 1 | 1 | 1 | 0 |
| 3 | 1 | 1 | 1 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 0 |
| 5 | 0 | 0 | 1 | 0 | 1 |

features matrix (nodes)

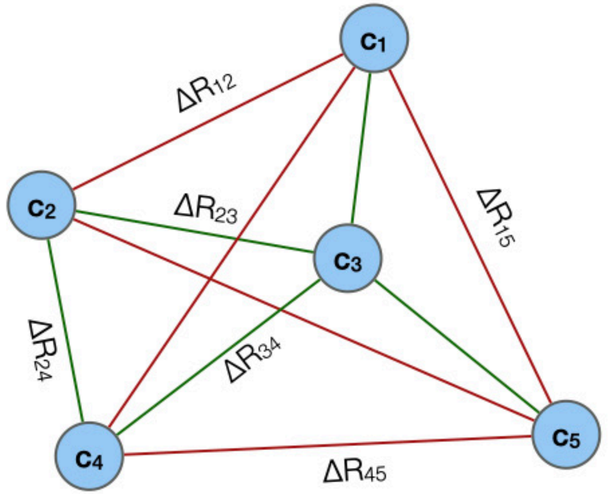
| n | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $f_1$ | 3.2 | 0.5 | -2.3 | 6.2 | 2.1 |
| $f_2$ | -1 | 1 | 1 | 1 | -1 |
| $f_3$ | 23 | 41 | 9 | 18 | 28 |



# GRAPH AUTOENCODERS FOR SVJ

## Using GAE for SVJ

- **adjacency matrix**: built out of **ΔR between constituents:**
  - if $\Delta R < \Delta R_c \rightarrow$ they are adjacent,
  - possible to use **ΔR directly**,
  - **other ways** to calculate AM also possible,
    e.g. scale ΔR by $p_T$ or $m_{inv}$ of a pair of constituents etc.,
- **features matrix: constituent 4-vectors + PDG ID + constituents impact parameter.**

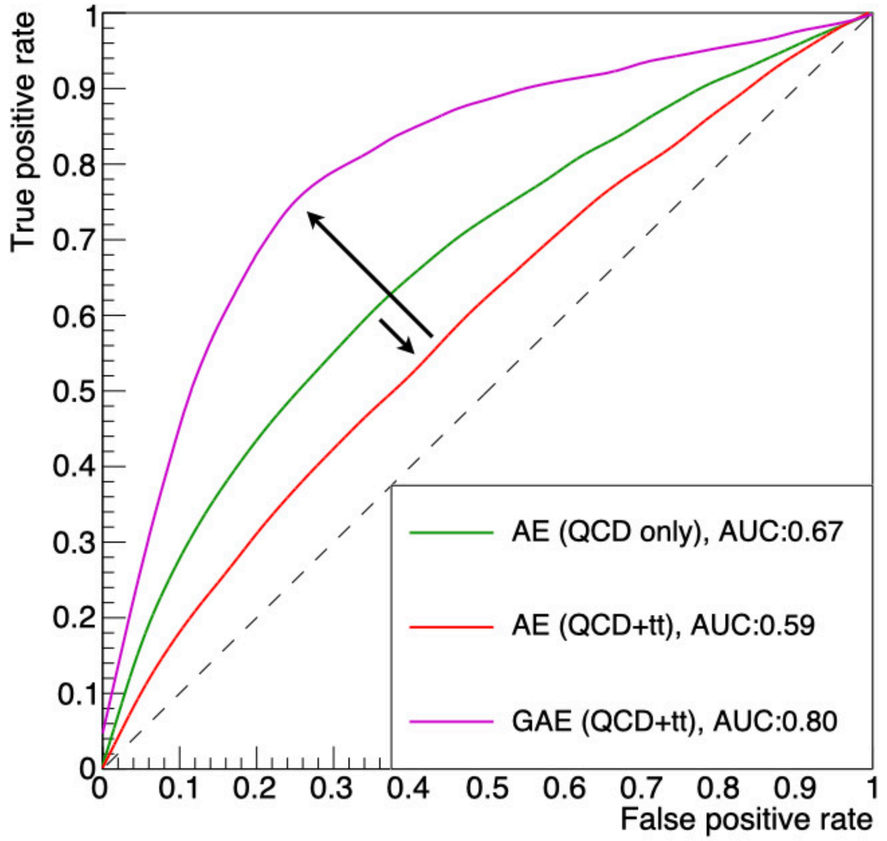



| Parameter | Value |
|---|---|
| $N_{constituents}$ | 50 |
| $\Delta R_c$ | 0.1 |
| Features | η, ϕ, m, $p_T$, ID, $d_0$, $d_z$ |
| $N_{jets}$ | 2 |
| Jet type | AK8 |
| Weighting | flat jet $p_T$ |
| Features normalization | Standard Scaler |
| Batch size | 4096 |
| Loss | huber |
| Optimizer | Nadam |
| Epochs | 10000 |
| Learning Rate | 1e-5 |
| ES patience | 100 |
| Architecture | 60, 60, 14 |
| Input activation | Softsign |
| Hidden activation | Linear |
| Dropout | 0.1 |

# GAE vs. AE — Performance

## Results comparison

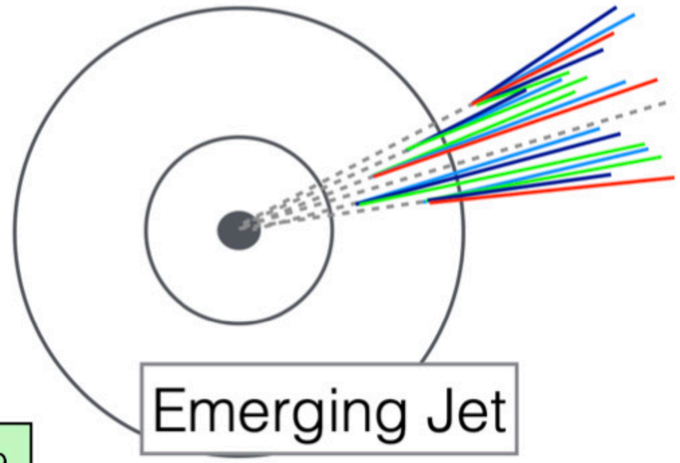
- both AE and GAE **optimized for QCD + $t\bar{t}$**,
- results for GAE very **preliminary**,
- **GAE gives great results**, allowing to mitigate performance drop caused by adding $t\bar{t}$ background!



# GAE — Emerging Jets

Can GAE spot other signatures?

- GAE based jet tagger is **largely independent from signal hypothesis** (training on background only)
  → should be sensitive to **any anomalous jets**,
- testing on **emerging jets** (EMJ),

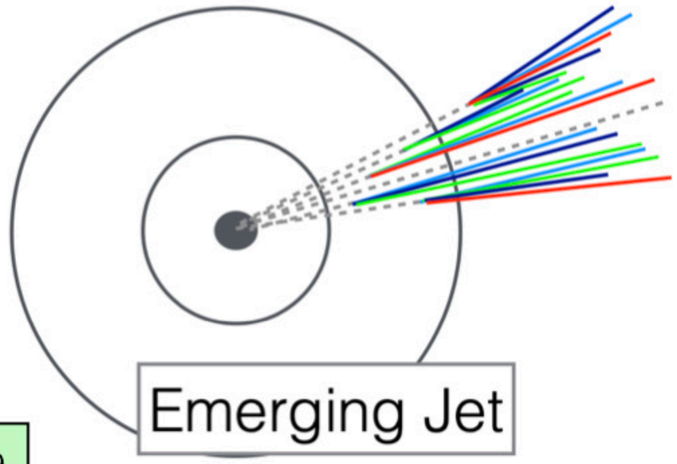



1502.05409

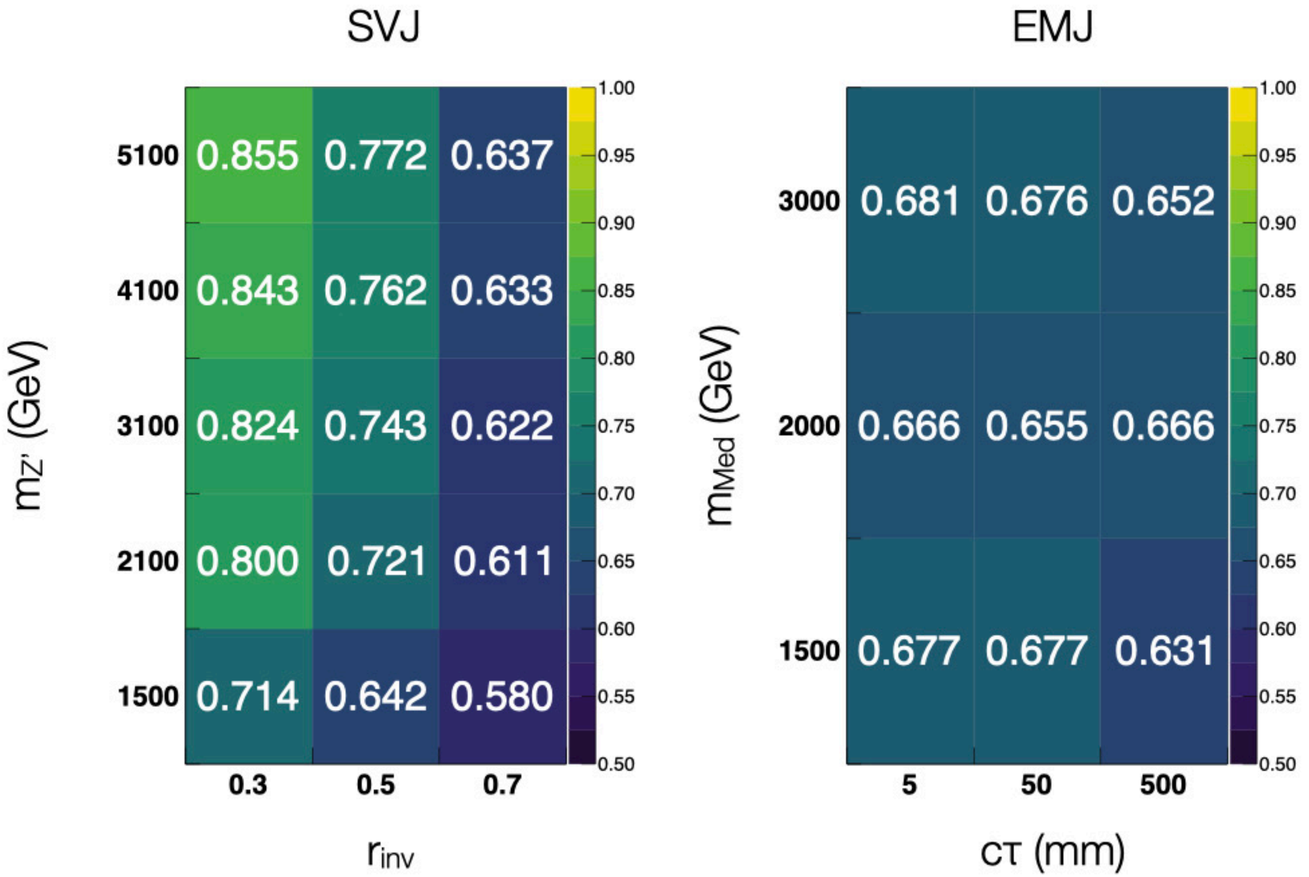
# GAE — Emerging Jets

## Can GAE spot other signatures?

- GAE based jet tagger is **largely independent from signal hypothesis** (training on background only)
  → should be sensitive to **any anomalous jets**,
- testing on **emerging jets** (EMJ),
- checking performance **out-of-the-box**,
  without any adjustments in preselection or input features,
- tagger yields **good sensitivity to EMJ**, despite this signal **not being used** for the method's development.




1502.05409



# CONCLUSIONS

## Autoencoders for anomalous jets detection

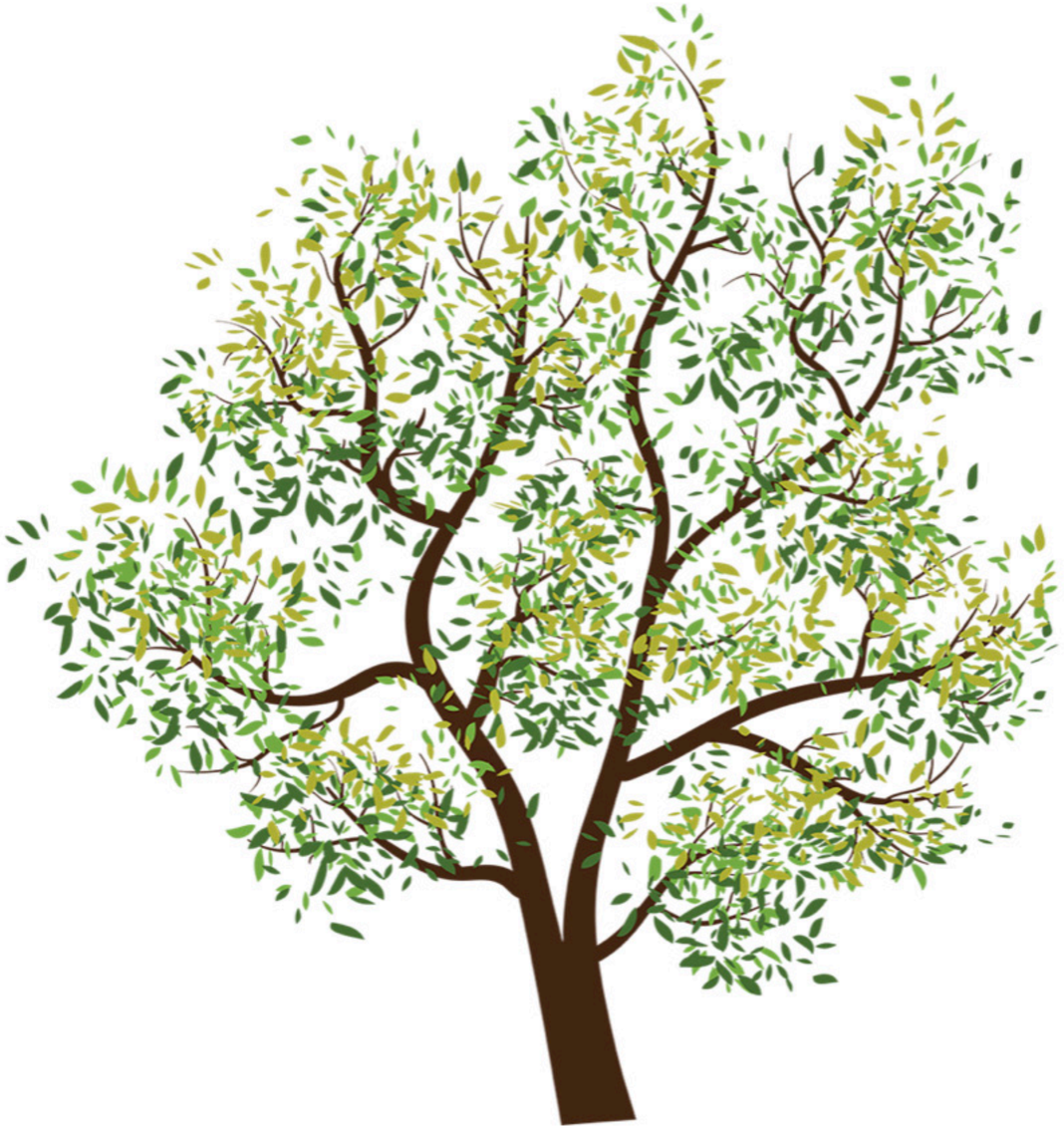
- semivisible jets introduce new **challenging signatures**,
- validated **autoencoders as a tool for semivisible jets** detection,
- **autoencoders are more robust** against different signal hypotheses than a supervised classifier,
- performance drop after adding **$t\bar{t}$ background**,
- **graph autoencoder** → excellent performance even including $t\bar{t}$ backgrounds,
- GAE also works well for **other signatures** (Emerging Jets),
- overall: **very promising results** for unsupervised learning.

# Backup

# CLASSIFIER

## Boosted Decision Tree (BDT)

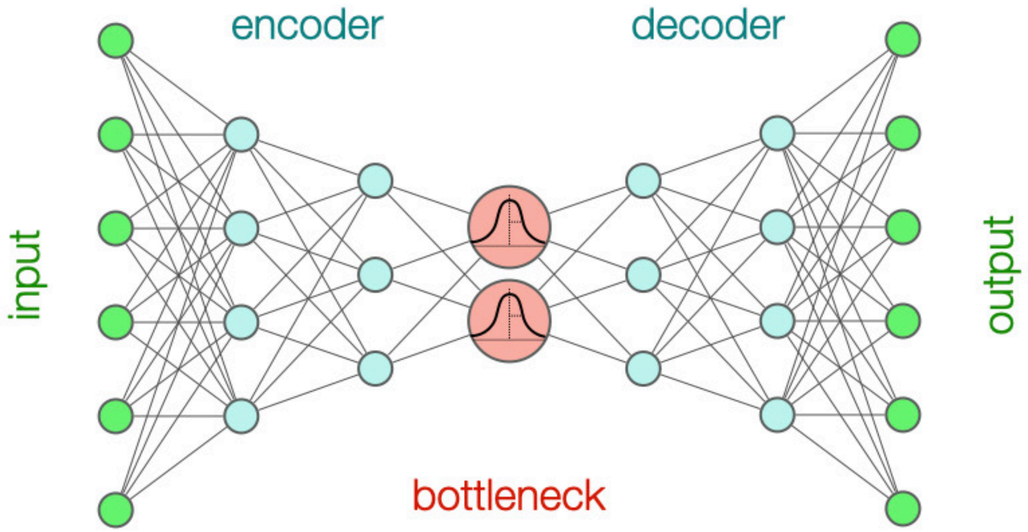
- basic classifier trained on both background and signal,
- fully **supervised learning,**
- trained on a **mixture of different signals,**
- used for comparison between supervised and unsupervised learning.



# ANOMALY DETECTION TECHNIQUES

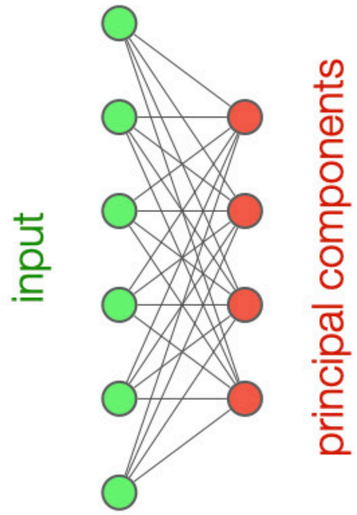
## Variational Auto-Encoder (VAE)

- same as auto-encoder, but **stores distributions** rather than single numbers in the latent space,
- can perform better than vanilla auto-encoder in certain cases,
- on the other hand, more complicated than AE,
- in our case: trained and tested only on reconstruction loss (no KL-loss included) → could be improved.




## Partial Component Analysis (PCA)

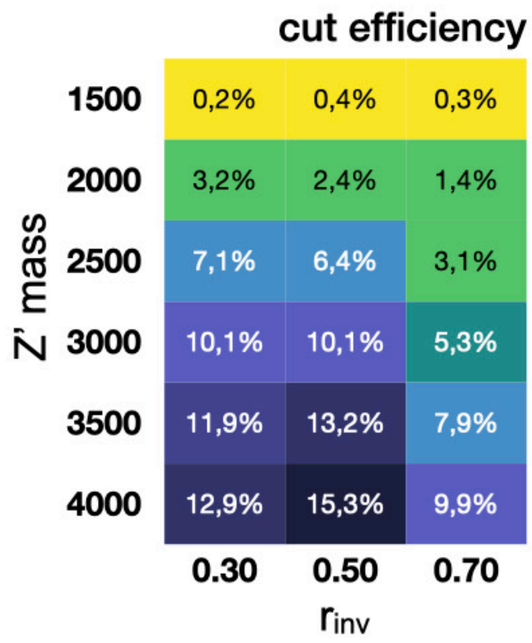
- well established **anomaly detection technique,**
- finds lower-dimension basis in which data are not correlated,
- it can be viewed as a very **simplified auto-encoder** (input → bottleneck → output),
- "weights" can be found **analytically**,
- less complicated than AE, but may yield worse performance.

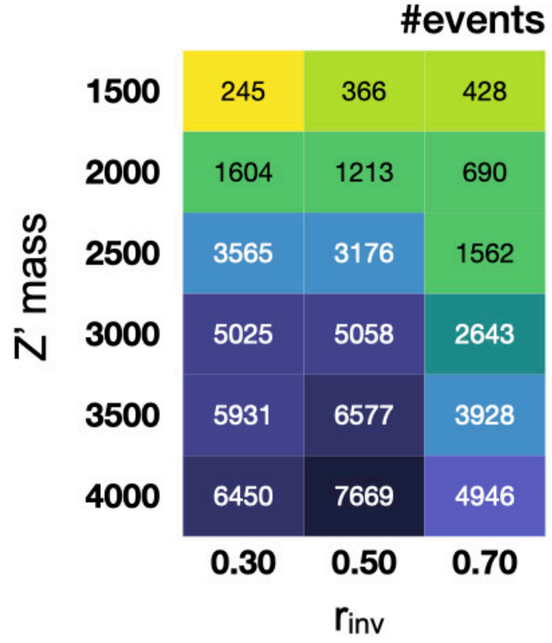


# PRESELECTION

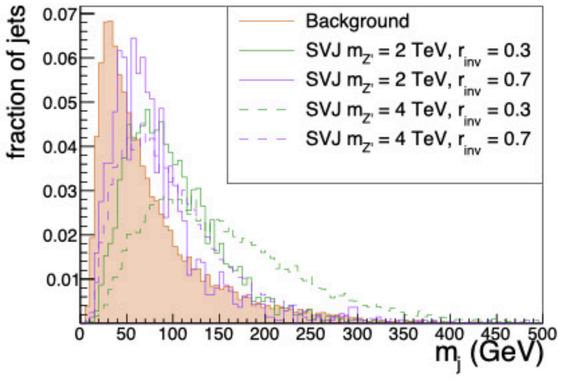
- applying simple preselection:
  - **2 healthy leading AK8 jets** (used for training/evaluation),
  - cuts on event topology and MET mimic CMS trigger and reduce QCD background,
- example **cut flow** in the table,
- SVJ cut efficiency between 0.2% and 15%, QCD efficiency 0.1%,
- ML sample: ≈100k QCD jets and 1-5k SV jets per signal point.

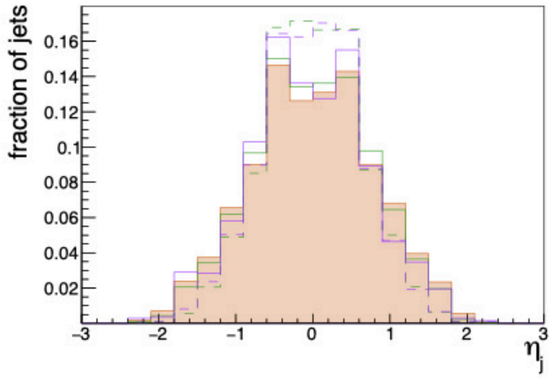
| | QCD | | | SVJ ($m_{Z'}$=3500, $r_{inv}$=0.3) | | |
|---|---|---|---|---|---|---|
| | #evt | abs ε (%) | rel ε (%) | #evt | abs ε (%) | rel ε (%) |
| Initial | 37315739 | 100 | 100 | 50000 | 100 | 100 |
| n jets ≥ 2 | 36918744 | 99 | 99 | 43260 | 87 | 87 |
| $\|\eta_{jet}\| < 2.4$ | 36334732 | 97 | 98 | 42687 | 85 | 99 |
| $\|\Delta\eta_{j1j2}\| < 1.5$ | 24625443 | 66 | 68 | 28440 | 57 | 67 |
| Jet $p_t$ > 200 GeV | 24625443 | 66 | 100 | 28440 | 57 | 100 |
| $m_t$ > 1500 GeV | 4565132 | 12 | 19 | 25009 | 50 | 88 |
| MET/$m_t$ > 0.25 | 48340 | 0,1 | 1,1 | 5931 | 11,9 | 24 |

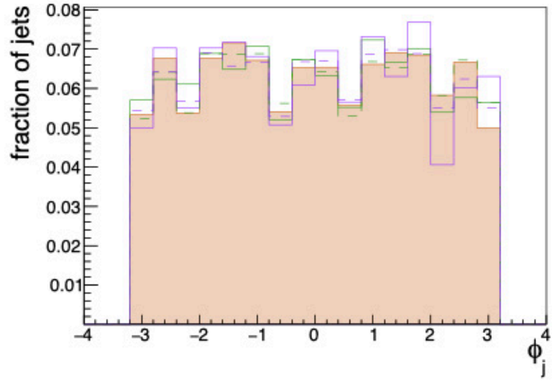


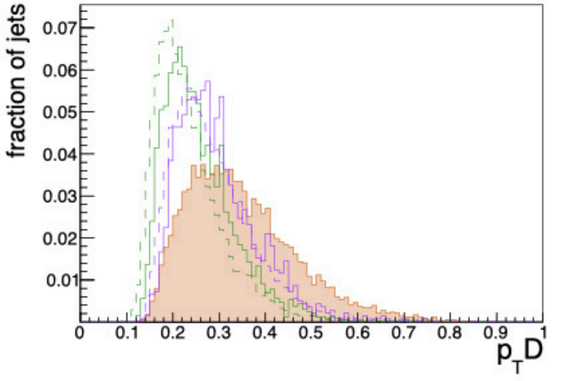



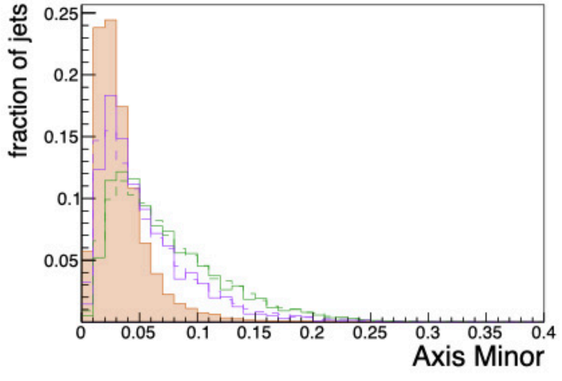
# AE — Input features

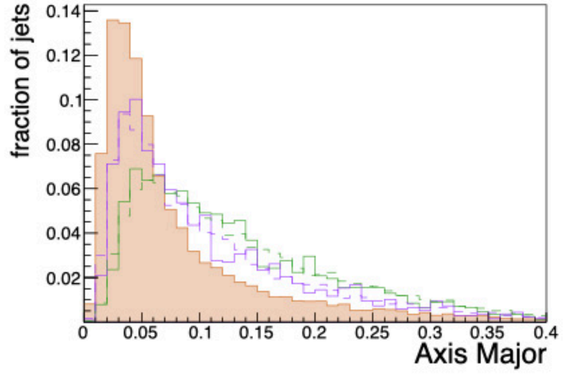


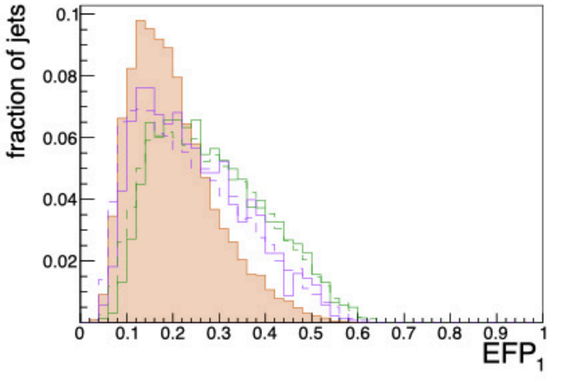



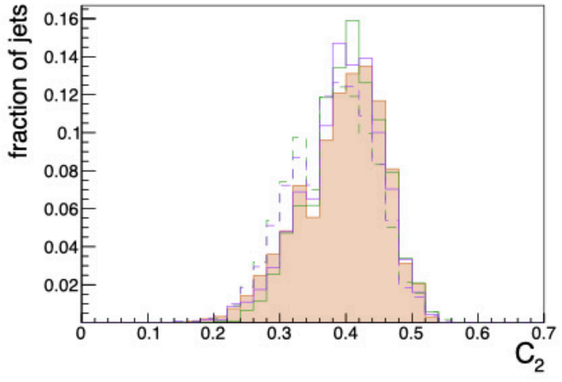



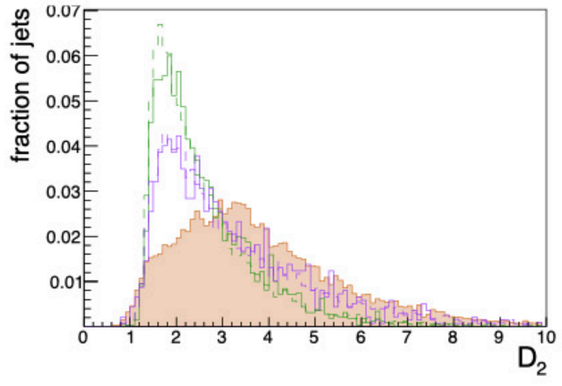



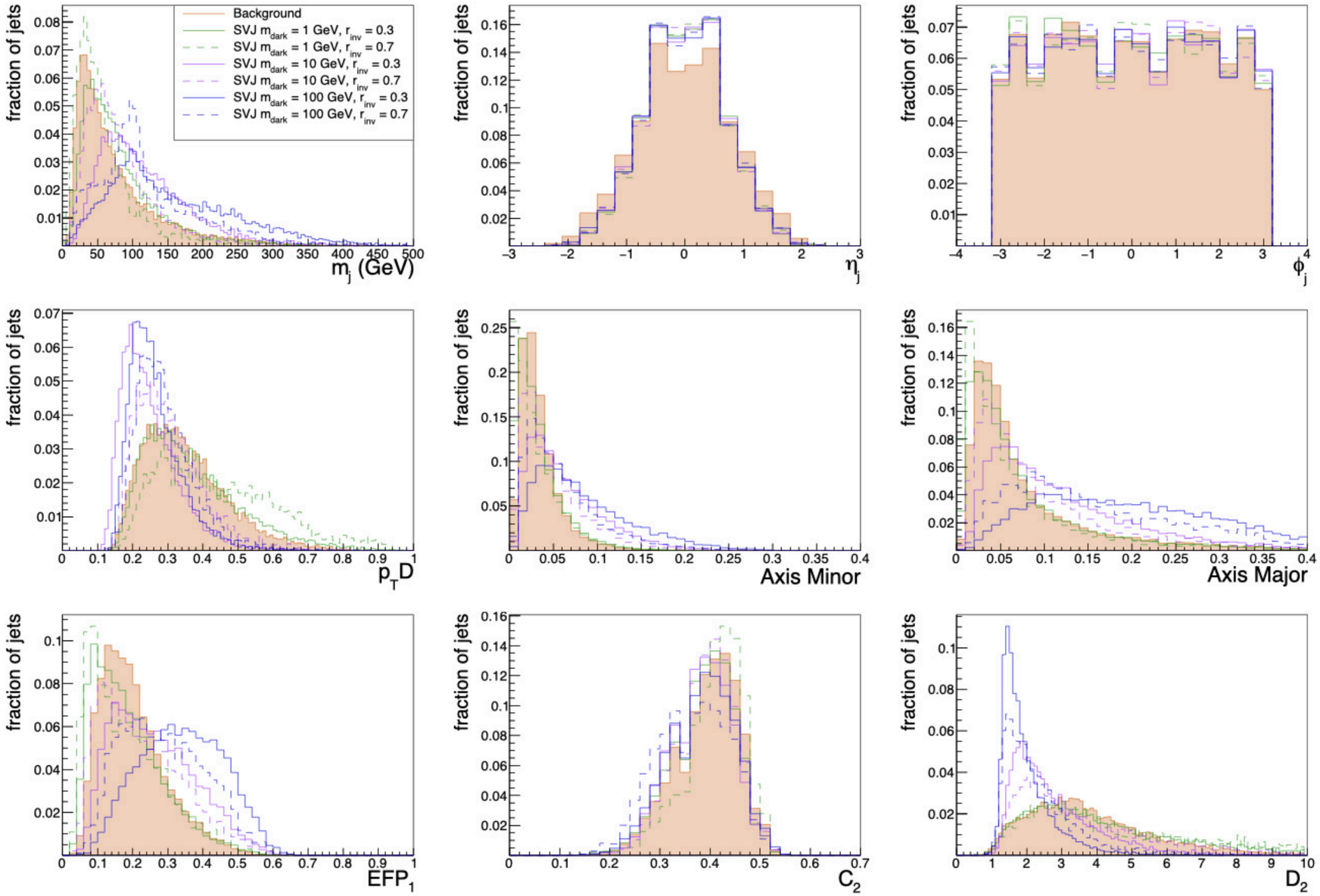



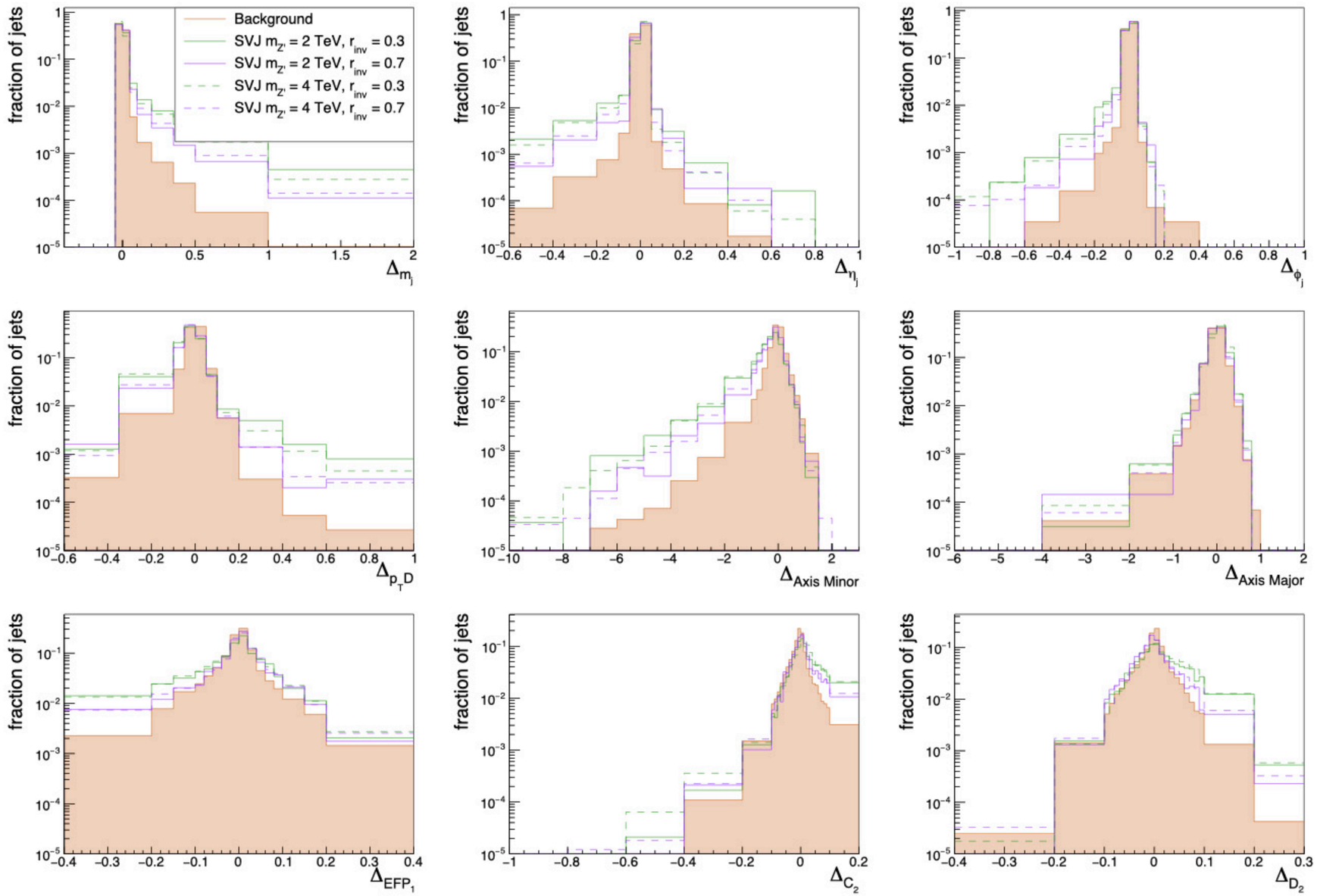

## Final set of variables

- made sure to select variables that give **good discrimination** and are **not too correlated** with each other,
- adding **four-momenta of jet constituents** didn't improve the performance, while significantly complicated network architecture,
- **selected jet features:**
  - η, ϕ → information about detector effects,
  - jet mass,
  - substructure variables ($p_TD$, jet axes, $EFP_1$, $C_2$, $D_2$).

# AE — Input Variables, $M_{DARK}$



# AE — Out-in residuals



# AE OPTIMIZATION

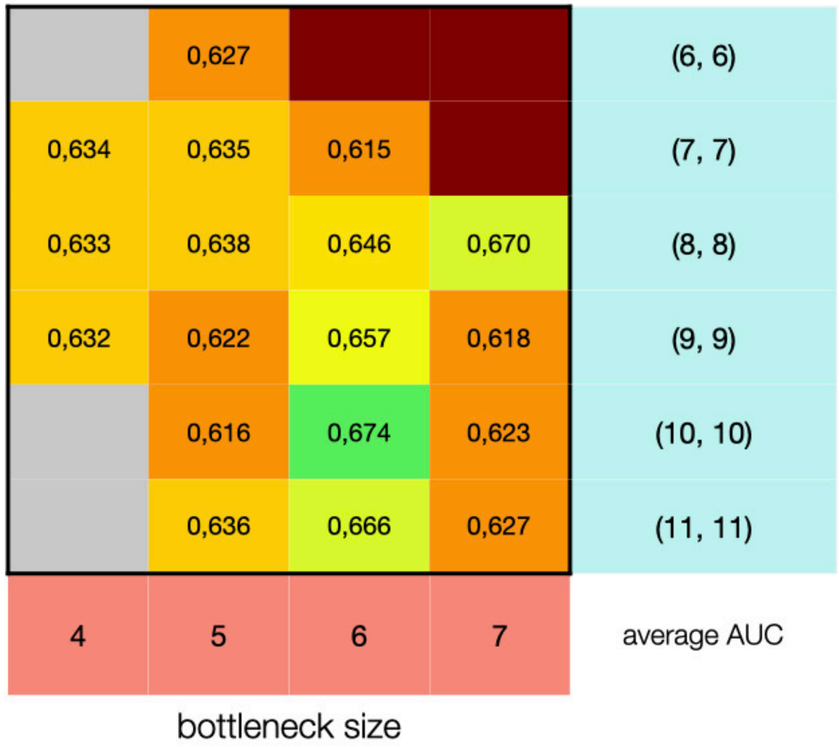
## Architecture optimization

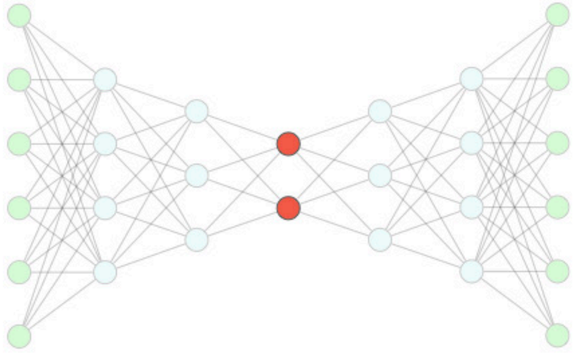
- scanned sizes of hidden layers and bottlenecks,
- **found an optimum** in Area Under ROC Curve (**AUC**) for 10 nodes in hidden layers and bottleneck of size 6.

## Final training

- a number of other **parameters and settings** was optimized iteratively,
- **150 models** trained for the final set of parameters,
- the model closest to the mean of AUC distribution presented in the following slides.

| Parameter | Value | | |
|---|---|---|---|
| architecture | (10, 10)-6 | learning rate | $10^{-6}$ |
| validation fraction | 0.15 | LR patience | 9 |
| test fraction | 0.15 | LR factor | 0.5 |
| loss function | mean abs error | ES patience | 12 |
| optimizer | Nadam | activation | elu |
| metric | accuracy | output activation | linear |
| $N_{epoch}$ | 200 | normalization | Standard Scaler |
| | | batch size | 256 |



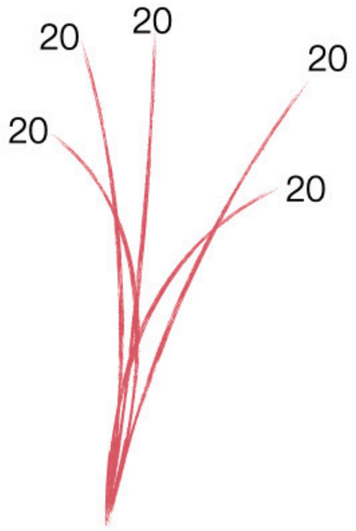
# SUBSTRUCTURE VARIABLES — $p_T D$

## Example substructure variable

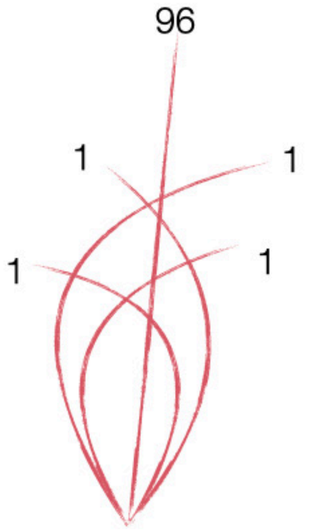
- jet transverse momentum dispersion ($p_T D$) is defined as:

$$p_T D = \frac{\sqrt{\sum p_{T,i}^2}}{\sum p_{T,i}}$$

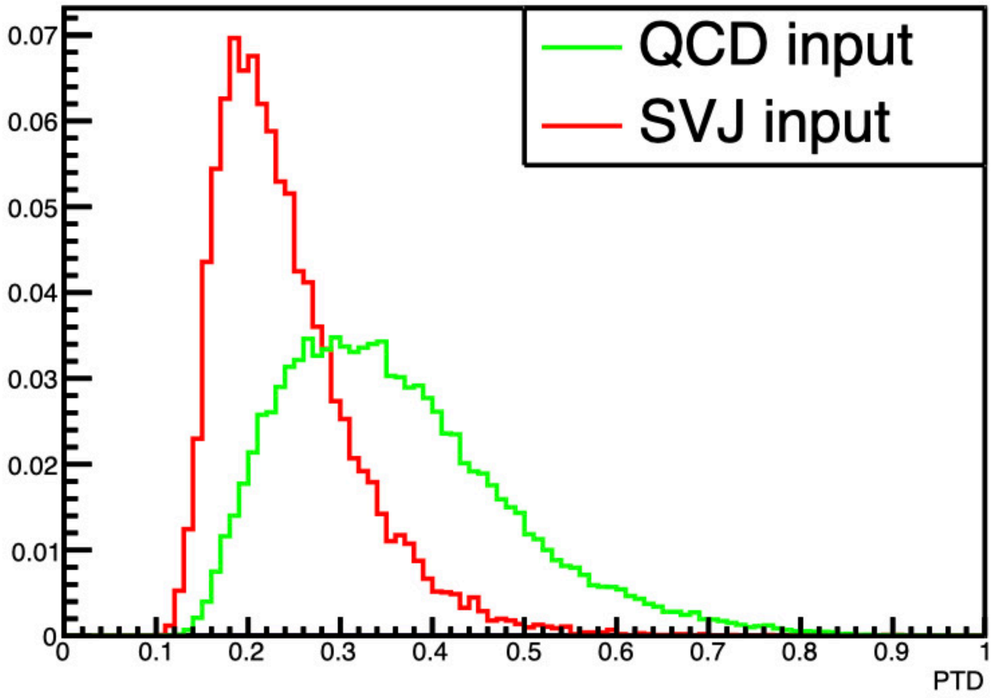
- it quantifies number of constituents that carry significant fraction of the $p_T$.




$p_T D = 0.45$




$p_T D = 0.96$



# SUBSTRUCTURE VARIABLES — JET AXES

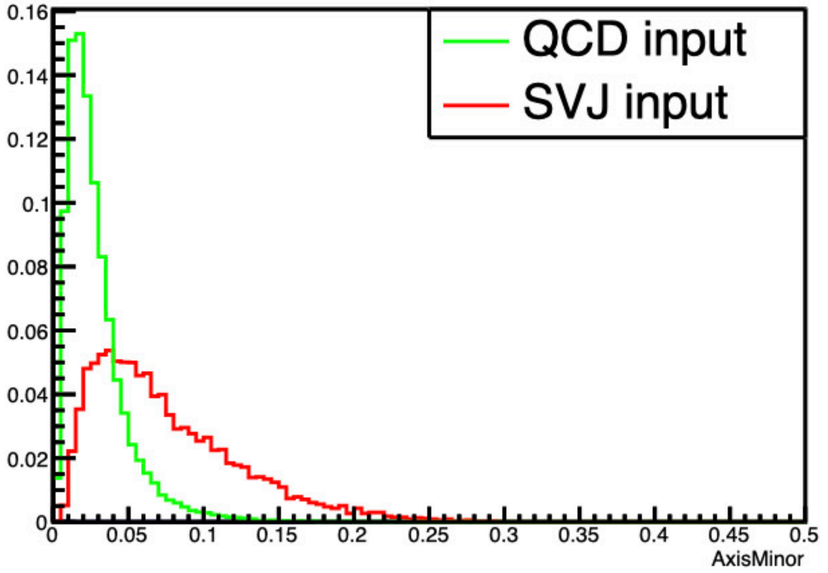
- jet axes are defined based on the $p_t$-weighted η-ϕ matrix of jet constituents:

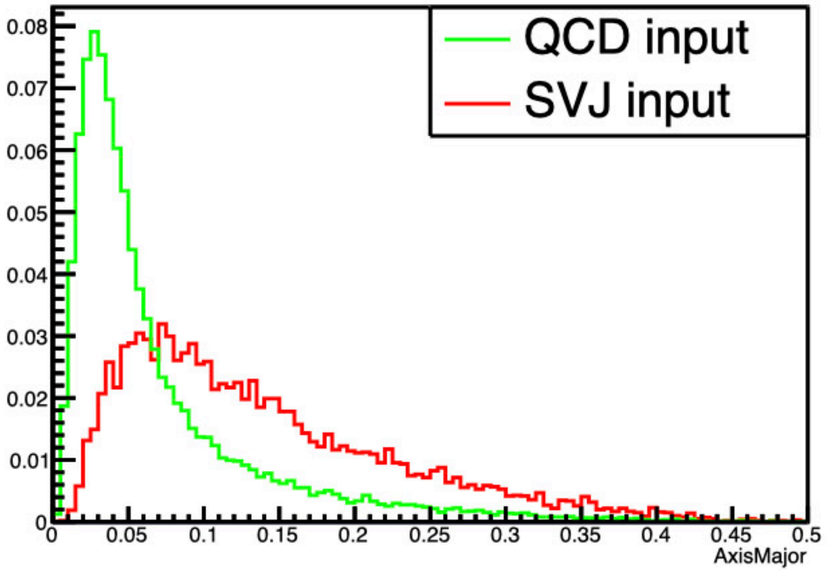
$$M = \begin{bmatrix} \sum p_{t,i}^2 \Delta\eta_i^2 & -\sum p_{t,i}^2 \Delta\eta_i \Delta\phi_i \\ -\sum p_{t,i}^2 \Delta\eta_i \Delta\phi_i & \sum p_{t,i}^2 \Delta\phi_i^2 \end{bmatrix}$$

- and are given as:

$$\sigma_{major} = \sqrt{\frac{\lambda_1}{\sum p_{t,i}^2}} \qquad \sigma_{minor} = \sqrt{\frac{\lambda_2}{\sum p_{t,i}^2}}$$

- where $\lambda_1$ and $\lambda_2$ are eigenvalues of M.



# SUBSTRUCTURE VARIABLES — ECF'S

- **pure ECF's** are calculated as:

$$ECF_1 = \sum p_{t,i}$$

$$ECF_2^\beta = \sum_{i<j} p_{t,i} p_{t,j} \Delta R_{ij}^\beta$$

$$ECF_3^\beta = \sum_{i<j<k} p_{t,i} p_{t,j} p_{t,k} (\Delta R_{ij} \Delta R_{ik} \Delta R_{jk})^\beta$$

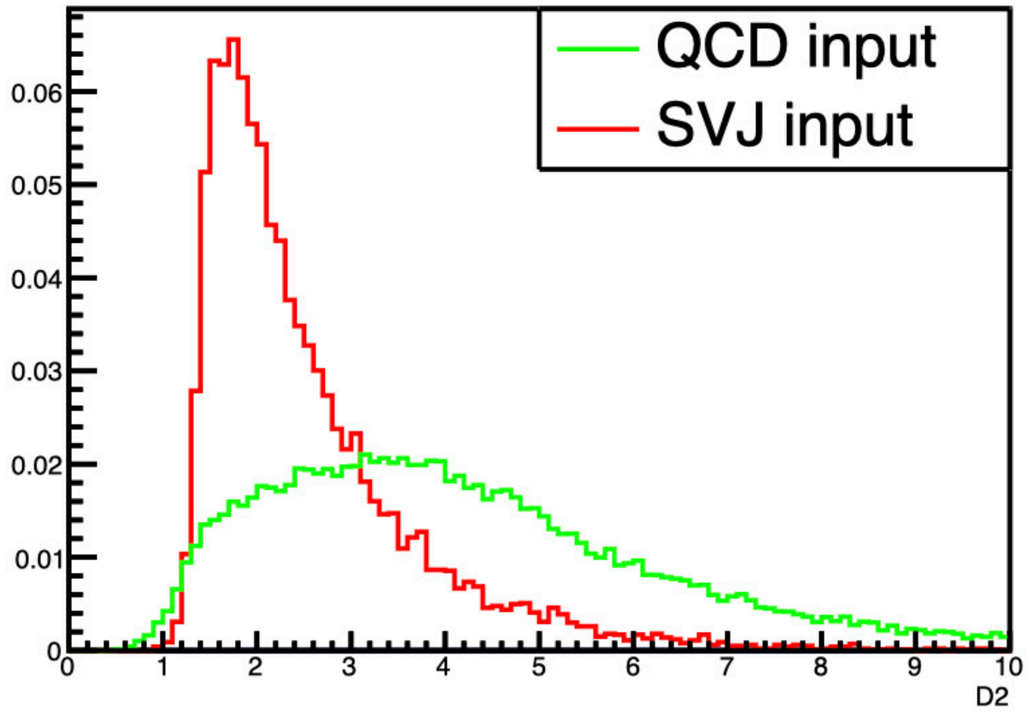
- then they are **normalized to sum of $p_t$ of constituents** and to the jet radius:

$$e_2 = \frac{ECF_2}{(ECF_1)^2 \cdot \Delta R}$$

$$e_3 = \frac{ECF_3}{(ECF_1 \cdot \Delta R)^3}$$

- finally, the following ratios are defined:

$$C_2 = \frac{e_3}{e_2^2} \qquad D_2 = \frac{e_3}{e_2^3}$$

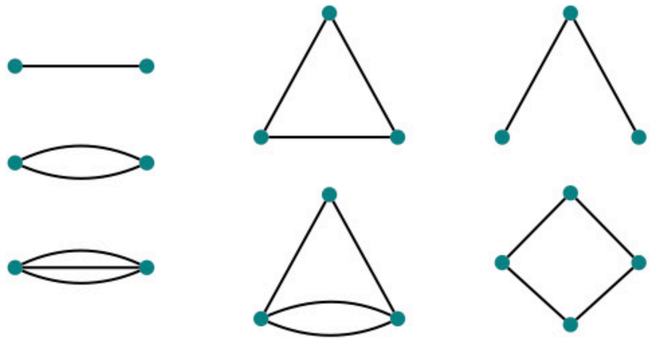


# SUBSTRUCTURE VARIABLES — EFP'S

- **EFP is defined** as:

$$EFP_G = \sum_{i_1}^{M} \cdots \sum_{i_N}^{M} z_{i_1} \cdots z_{i_N} \prod_{(k,l)\in G} \Delta R_{i_k i_l}$$

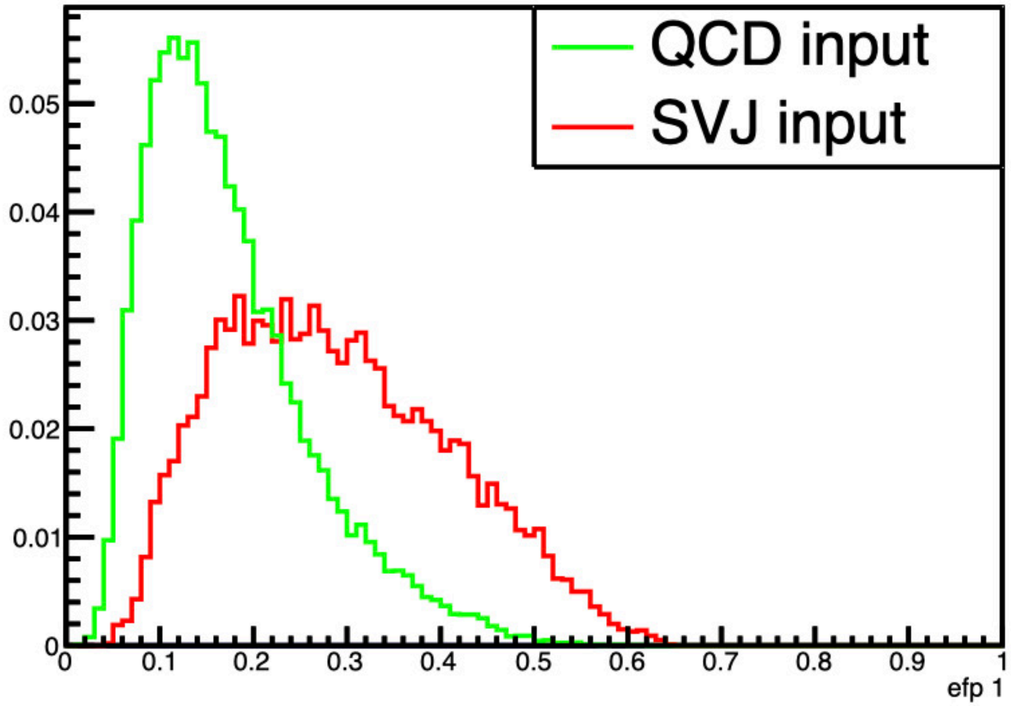
- where $z_i = p_{t,i} / p_{t,jet}$,
- those complicated indices can be viewed as **graphs** for which given EFP is calculated - G index tell us how many edges graphs should have:



- $EFP_1$ represents the **simplest graph** (two nodes, one edge), which turns out to be **equivalent to (normalized)** $ECF_2$:

$$EFP_1 = \sum z_i z_j \Delta R_{ji}$$

$$ECF_2^{\beta} = \sum_{i<j} p_{t,i} p_{t,j} \Delta R_{ij}^{\beta}$$

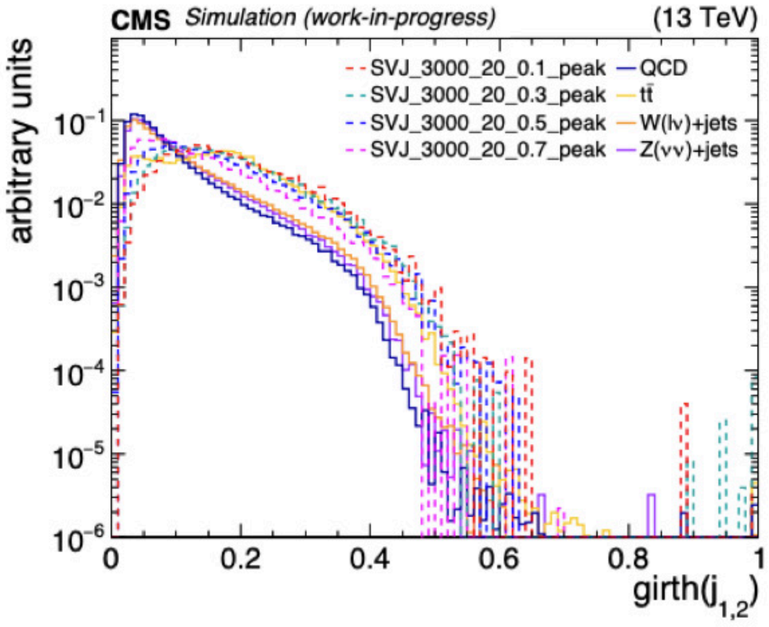


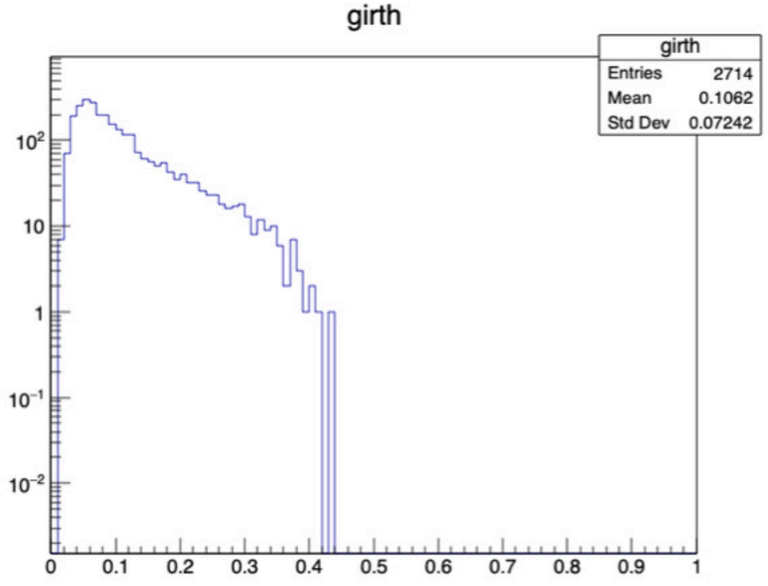
# SUBSTRUCTURE VARIABLES — GIRTH

- definition:

$$g = \sum_i \frac{p_{\mathrm{T},i}}{p_{\mathrm{T,jet}}} r_i.$$

Distances $r$ of each particle or cell from the jet center are calculated on the (rapidity,phi) cylinder. The jet center is taken as the $(y, \phi)$ of the jet's 4-vector, but the $p_T$-weighted centroid is almost identical. It is important to use rapidity rather than pseudorapidity for the jet location because the jet is massive. A radial moment sums a function of these distances, weighted by $p_T$, then normalized to the total $p_T$ of the jet.



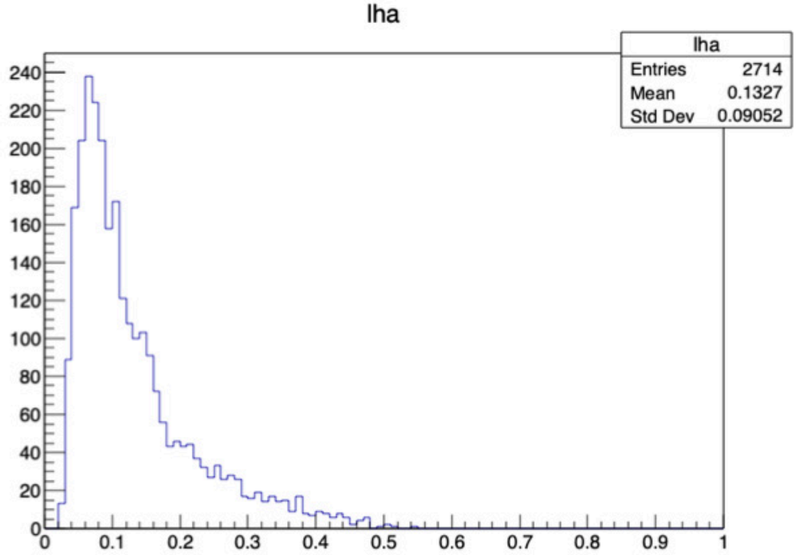
# SUBSTRUCTURE VARIABLES — LHA

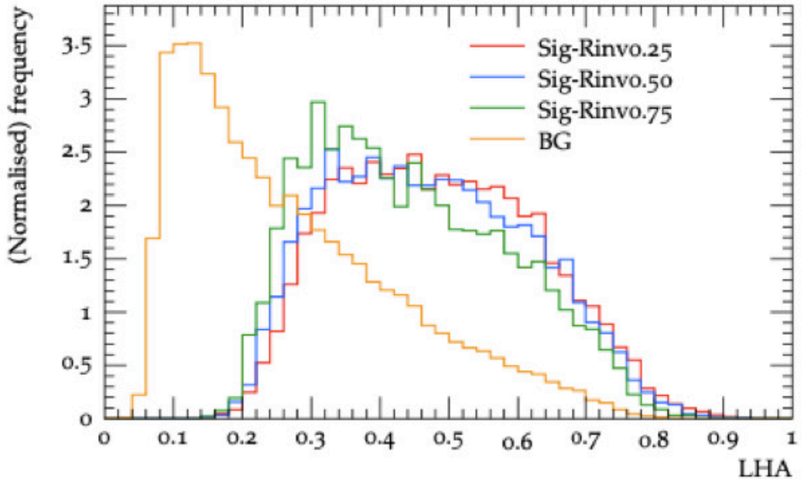
- LHA is defined as:

$$\lambda_\beta^\kappa = \sum_{i \in \text{jet}} z_i^\kappa \theta_i^\beta,$$

where $i$ runs over the jet constituents, $z_i \in [0,1]$ is a momentum fraction, and $\theta_i \in [0,1]$ is a (normalized) angle to the jet axis.

- θ is ΔR normalized to jet radius (0.8),
- $z_i$ is constituent $p_t$ normalized to total $p_t$.



# AE — Inputs correlations

## Correlations between EFP's

- for good auto-encoder performance → **input variables shouldn't be too correlated**,
- we found that for QCD (and SVJ), all **EFP's are fully correlated** with each other,
- because of that, we decided to **only consider including $EFP_1$**.

## Correlations between other inputs

- similar correlations for QCD and SVJ,
- girth, LHA, $e_2$, $e_3$ and $EFP_1$ **fully correlated**,
- $EFP_1$ correlated with axes, which are also correlated with each other.

## Final set of variables

- η, ϕ, m, $p_tD$, axis minor, axis major, $EFP_1$, $C_2$, $D_2$,
- the largest remaining correlation (≈0.8) between jet axes and $EFP_1$.

EFP's correlations (QCD)

| efp | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1,0 | 1,0 | 0,9 | 1,0 | 0,9 | 0,9 | 0,9 | 0,9 | 1,0 | 0,9 | 0,9 | 0,9 |
| 2 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 |
| 3 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 |
| 4 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 |
| 5 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 |
| 6 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 |
| 7 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 | 1,0 | 1,0 | 0,9 |
| 8 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 |
| 9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 |
| 10 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 |
| 11 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 |
| 12 | 0,9 | 0,9 | 0,9 | 0,9 | 0,9 | 0,9 | 0,9 | 1,0 | 1,0 | 1,0 | 1,0 | 1,0 |

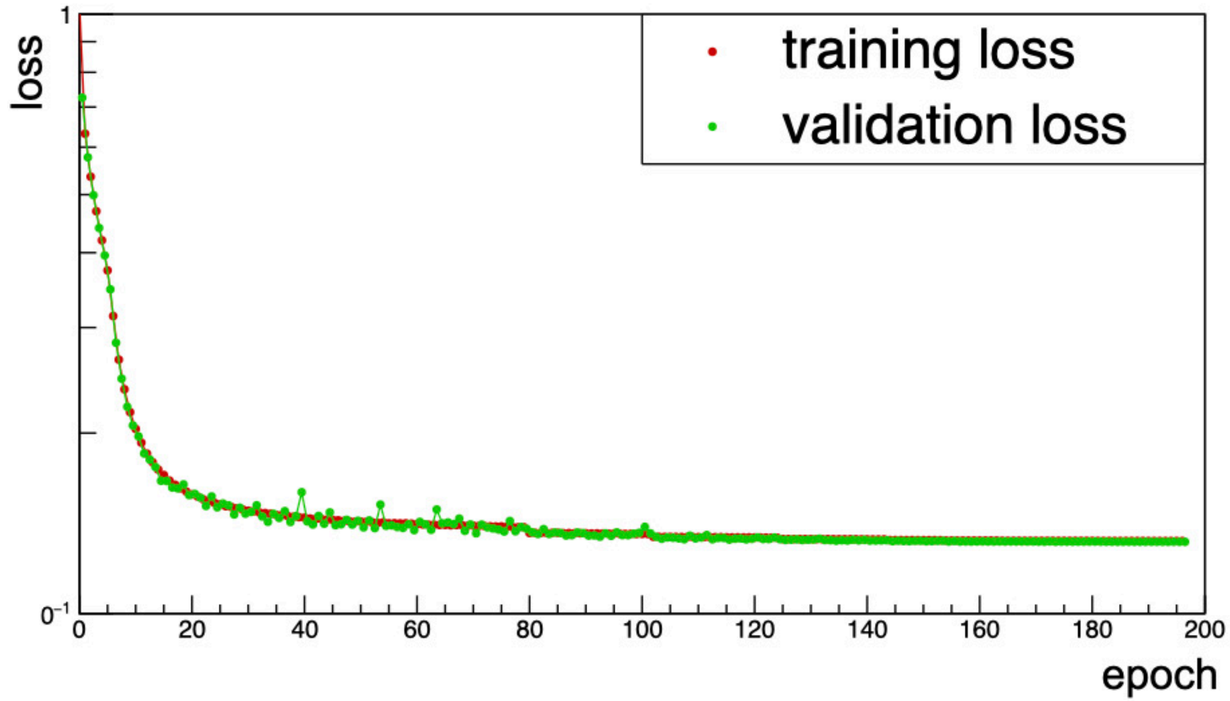
input correlations (QCD)

| | efp 1 | Eta | Phi | M | ptD | AxisMinor | AxisMajor | Girth | LHA | e2 | e3 | C2 | D2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| efp 1 | 1,0 | -0,0 | 0,0 | 0,5 | -0,5 | 0,8 | 0,8 | 1,0 | 1,0 | 1,0 | 0,9 | -0,0 | 0,0 |
| Eta | -0,0 | 1,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | -0,0 | -0,0 | 0,0 | -0,0 |
| Phi | 0,0 | 0,0 | 1,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 | 0,0 |
| M | 0,5 | 0,0 | 0,0 | 1,0 | -0,4 | 0,4 | 0,6 | 0,6 | 0,6 | 0,5 | 0,3 | -0,5 | -0,1 |
| PTD | -0,5 | 0,0 | 0,0 | -0,4 | 1,0 | -0,6 | -0,5 | -0,4 | -0,4 | -0,5 | -0,4 | 0,3 | -0,1 |
| AxisMinor | 0,8 | 0,0 | 0,0 | 0,4 | -0,6 | 1,0 | 0,7 | 0,7 | 0,7 | 0,8 | 0,8 | 0,0 | 0,3 |
| AxisMajor | 0,8 | 0,0 | 0,0 | 0,6 | -0,5 | 0,7 | 1,0 | 0,9 | 0,9 | 0,8 | 0,7 | -0,3 | -0,1 |
| Girth | 1,0 | 0,0 | 0,0 | 0,6 | -0,4 | 0,7 | 0,9 | 1,0 | 1,0 | 1,0 | 0,9 | -0,2 | -0,1 |
| LHA | 1,0 | 0,0 | 0,0 | 0,6 | -0,4 | 0,7 | 0,9 | 1,0 | 1,0 | 1,0 | 0,9 | -0,2 | -0,1 |
| e2 | 1,0 | -0,0 | 0,0 | 0,5 | -0,5 | 0,8 | 0,8 | 1,0 | 1,0 | 1,0 | 0,9 | -0,0 | 0,0 |
| e3 | 0,9 | -0,0 | 0,0 | 0,3 | -0,4 | 0,8 | 0,7 | 0,9 | 0,9 | 0,9 | 1,0 | 0,2 | 0,3 |
| C2 | -0,0 | 0,0 | 0,0 | -0,5 | 0,3 | 0,0 | -0,3 | -0,2 | -0,2 | -0,0 | 0,2 | 1,0 | 0,6 |
| D2 | 0,0 | -0,0 | 0,0 | -0,1 | -0,1 | 0,3 | -0,1 | -0,1 | -0,1 | 0,0 | 0,3 | 0,6 | 1,0 |

# AE — Loss evolution

## Training quality

- **loss evolution** for all trained models looks good (no signs of overtraining/undertraining).




loss

# AE — ROC CURVES

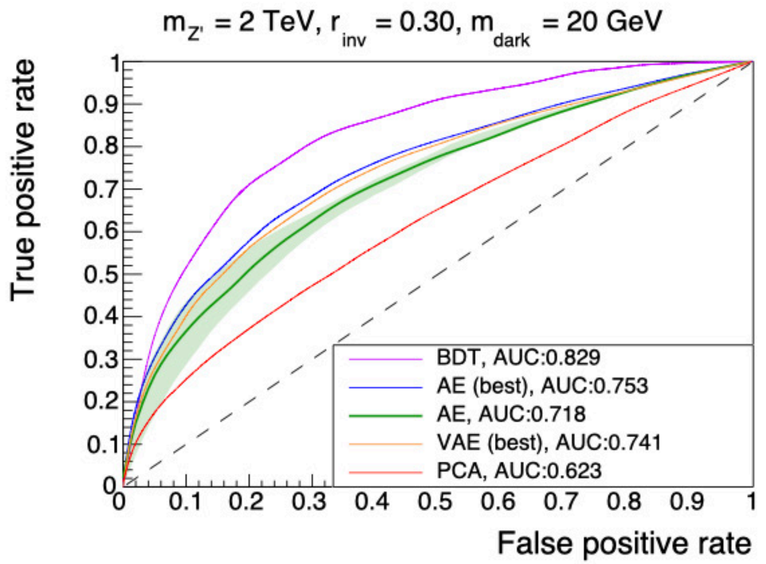
## ROC's comparison

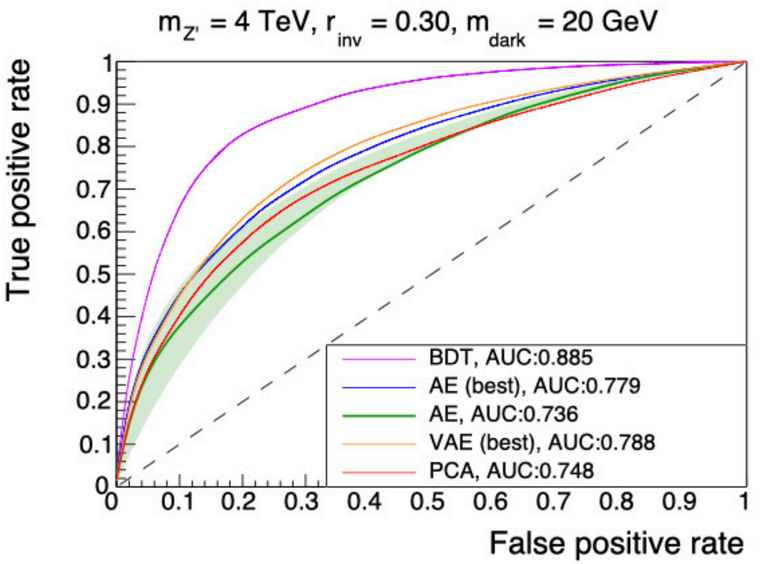
- as expected, BDT yields the best performance,
- **AE is not that far behind BDT!**

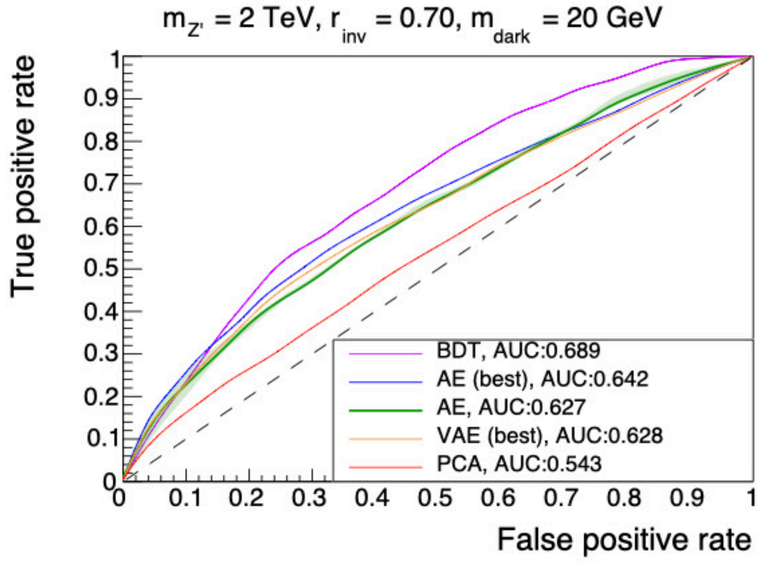
## Alternative approaches

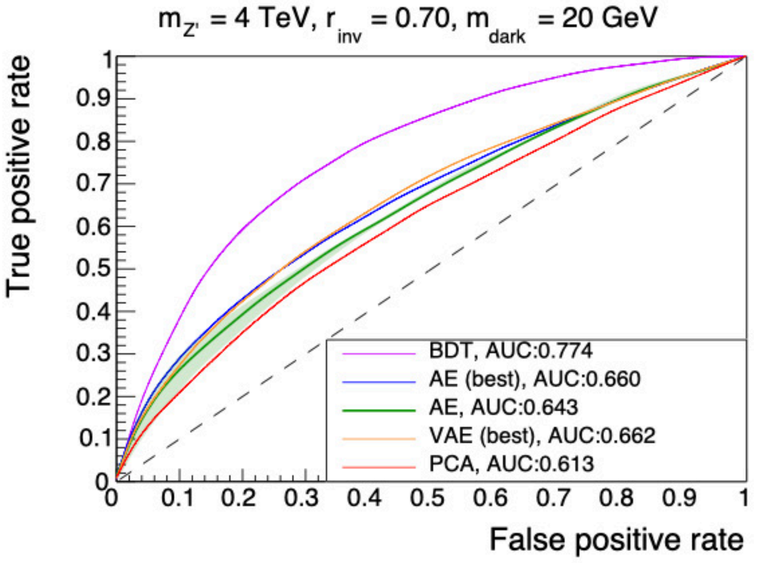
**Other anomaly detection techniques** were also investigated:

- Variational Autoencoder (VAE): similar to plain AE (but we trained and tested on reconstruction loss only — could be improved),
- Principal Component Analysis (PCA): **significantly worse** than other architectures.



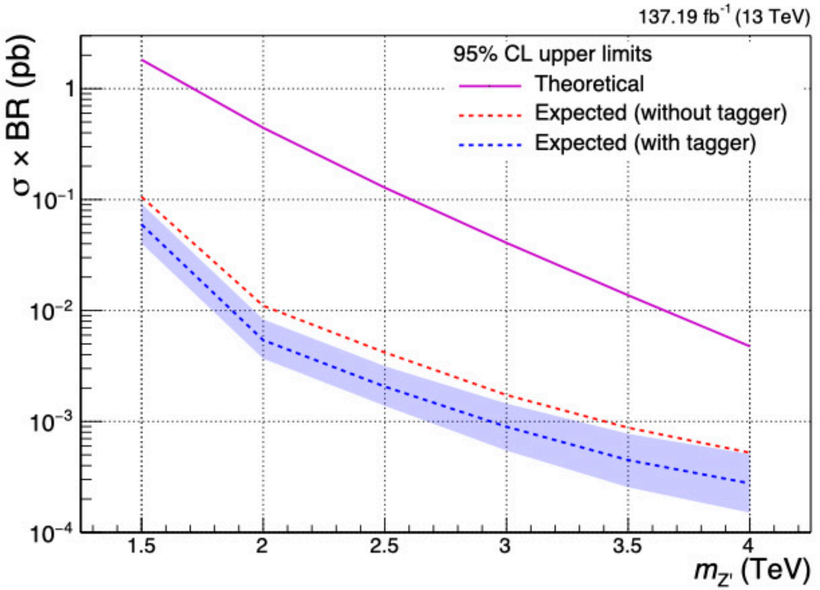



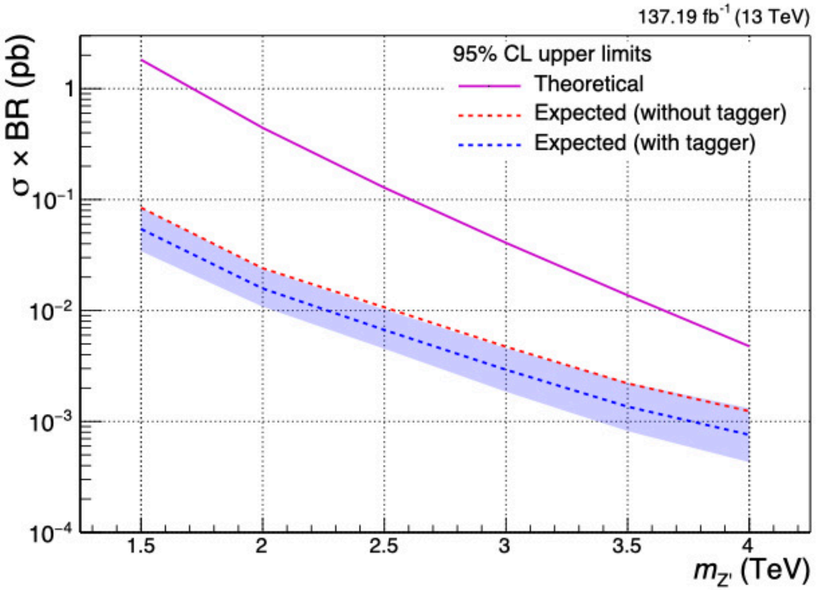



# AE — LIMITS

Run 2, $r_{inv}$=0.3, $m_{Dark}$ = 20 GeV




Run 2, $r_{inv}$=0.7, $m_{Dark}$ = 20 GeV



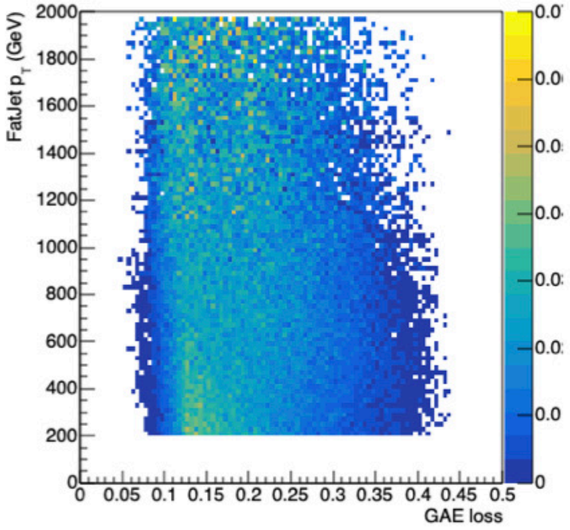

- events classified into 3 categories: 0 SVJ, 1 SVJ and 2 SVJ,
- transverse mass $m_T$ histograms used to set limits,
- only **basic uncertainties** taken into account (luminosity, trigger),
- limits are stronger for lower values of $r_{inv}$ and improvement from using the tagger is more pronounced in this region,
- autoencoder based approach is able to cover a large area of the phase-space.

# GAE TAGGER VS. JET $p_T$

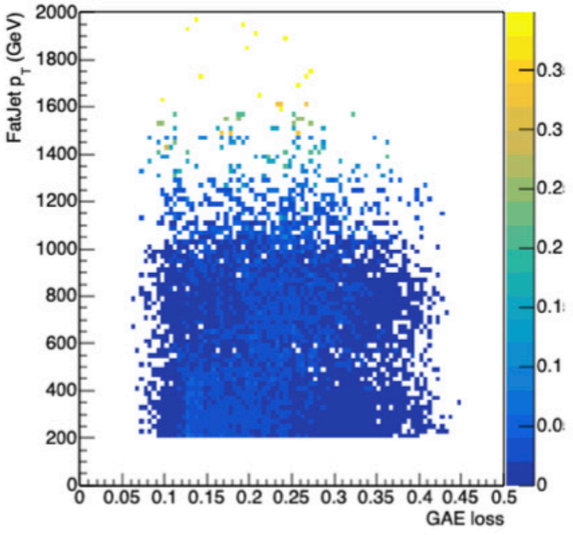
## Is jetGAE correlated with $p_T$?

- we don't want the network to learn **differences in jet $p_T$** distribution → **flattening** is applied to the training sample,
- verified that **correlation is small** for backgrounds (a little bit larger for the signal),
- more detailed tests ongoing,
- if needed — DisCo (distance correlation) minimization already implemented in our framework
  → can be reused to **decorrelate jetGAE loss from jet $p_T$**.

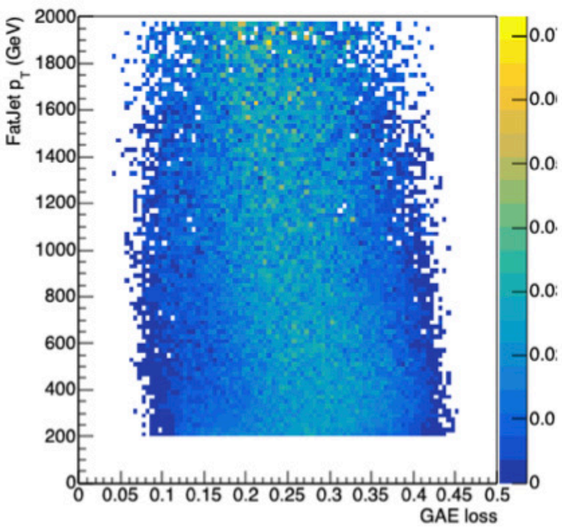
QCD
flattened jet $p_T$




$t\bar{t}$
flattened jet $p_T$




SVJ
flattened jet $p_T$

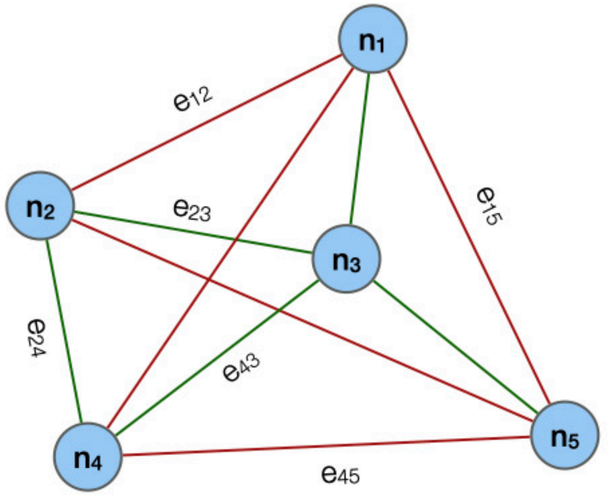


# GRAPH NN — EXISTING SOLUTIONS

## Existing papers/solutions

Some solutions exist, but **none can be directly applied** to our problem (they are classifiers, reconstruct one huge graph etc.):

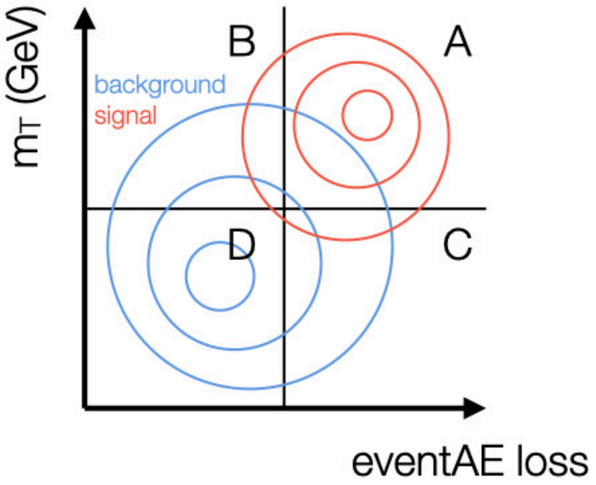
- https://github.com/tkipf/gae (based on this **paper**: https://arxiv.org/abs/1611.07308) → **GVAE** (Python 2, TensorFlow 1),
- https://github.com/zfjsail/gae-pytorch (**reimplementation** of the above with **PyTorch and Python 3**),
- https://vermamachinelearning.github.io/keras-deep-graph-learning/Layers/Convolution/graph_conv_layer/ (Keras extension with **convolutional graph layers**),
- nice **GAE description**: https://par.nsf.gov/servlets/purl/10209598
- summary of **Graph NN in HEP**: https://arxiv.org/abs/2203.12852
- **GAE for anomaly detection**: https://link.springer.com/article/10.1007/JHEP08(2021)080



# ANALYSIS STRATEGY

## Background estimation / signal extraction

- **eventAE loss vs. $m_T$** for the **ABCD-style** background estimation,
- if needed, use DisCo to **decorrelate loss and $m_T$**,
- **fit $m_T$ distribution** in the signal region.




## Other checks and improvements

- find an **orthogonal control region** to verify the procedure in data
  - expected no excess (or low significance),
- verify **robustness against detector effects**
  - e.g. inject events which don't pass ϕ spike or HEM filters into the training dataset and use them as signal while testing → should yield no discrimination,
- **relax preselections** to make it even more generic
  - e.g. $r_{inv}$ = 0.0, emerging jets etc.,
- could be **extended to HLT and scouting**
  - first studies done on jetAE → the **same performance** as with fully-reconstructed events.