Boosted W Tagging using Lund Jet Plane

Rafael Andrei Vinasco Soler - Unal

Supervisors: Reina Camacho Toro, Carlos Sandoval.

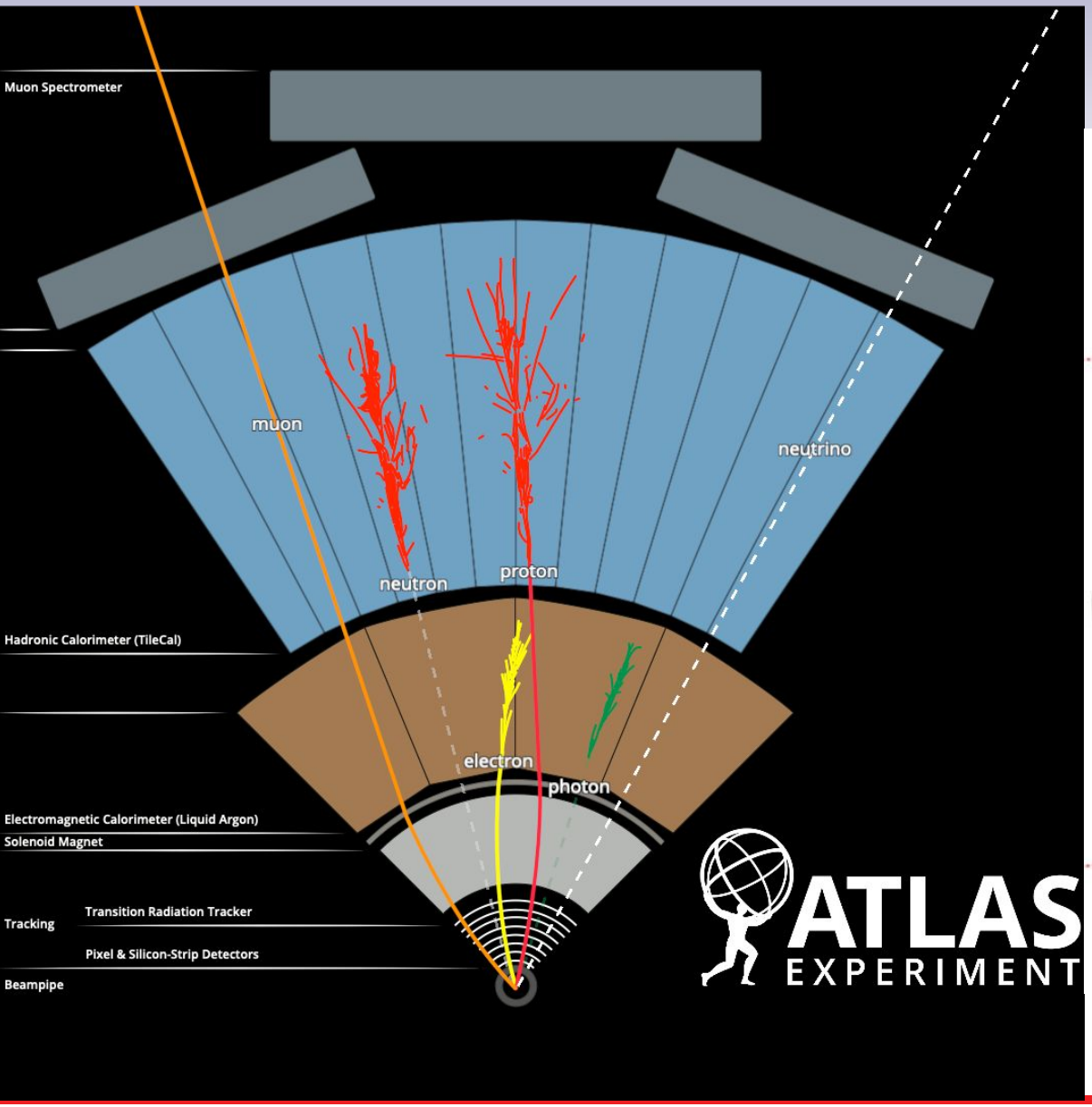
And thanks to Mykola Khandoga and Jad Mathieu Sardain for their full support

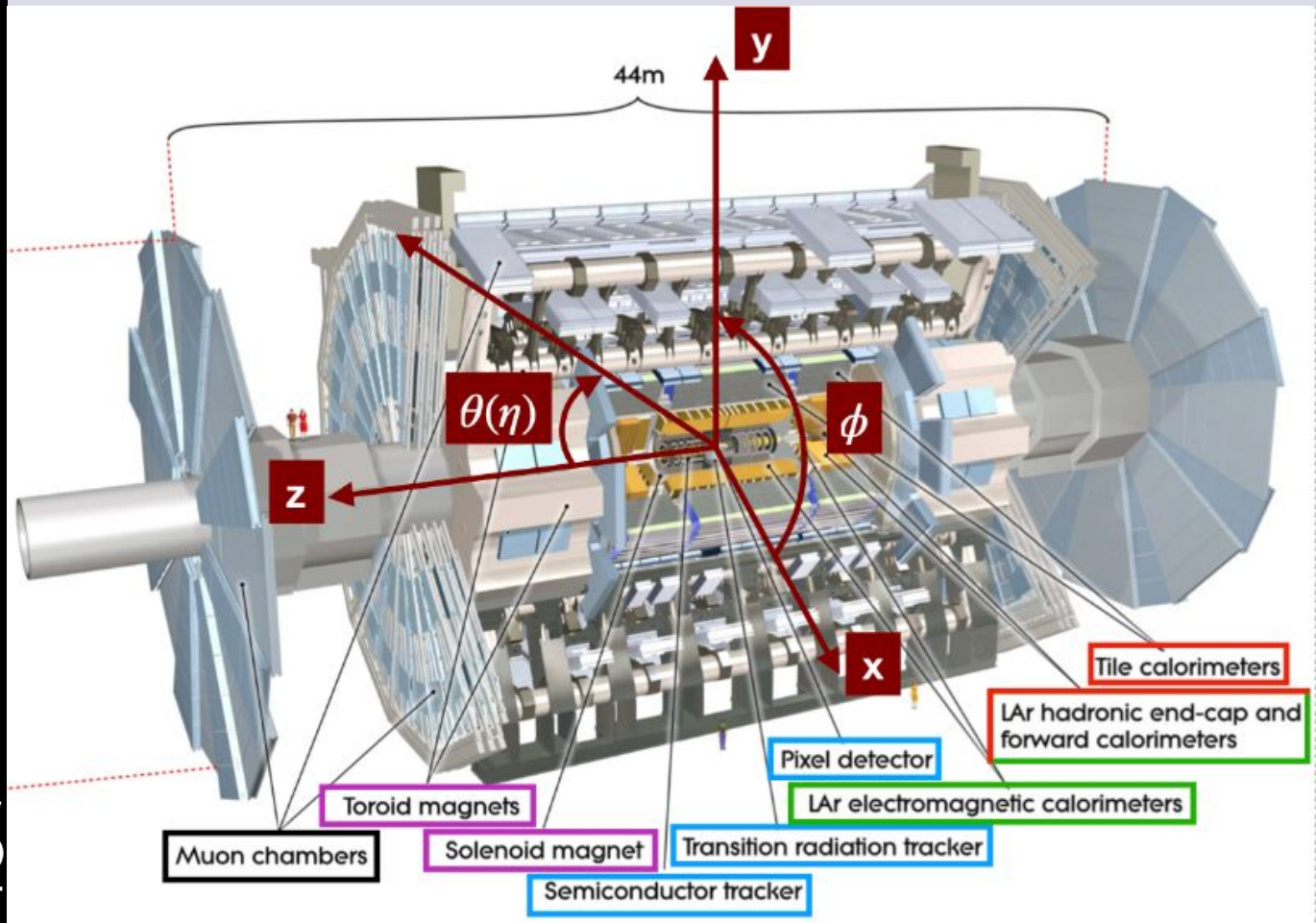
Latin American alliance for Capacity buildiNG in Advanced physics

LA-CoNGA physics

Cofinanciado por el programa Erasmus+ de la Unión Europea

ATLAS DETECTOR



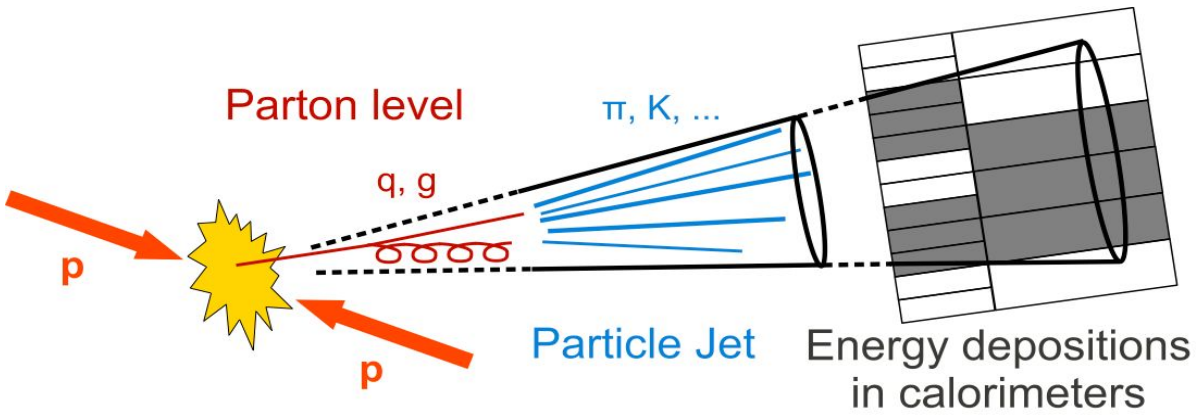


What is our goal?

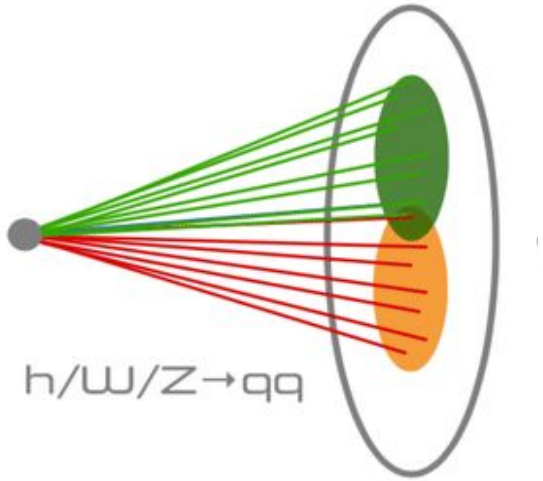
Use Lund plane variables as input for machine learning methods to develop a new **tagging methods for boosted W boson**.

Jet: A set of collimated particles produced in the hadronization of a quark or gluon.

Background

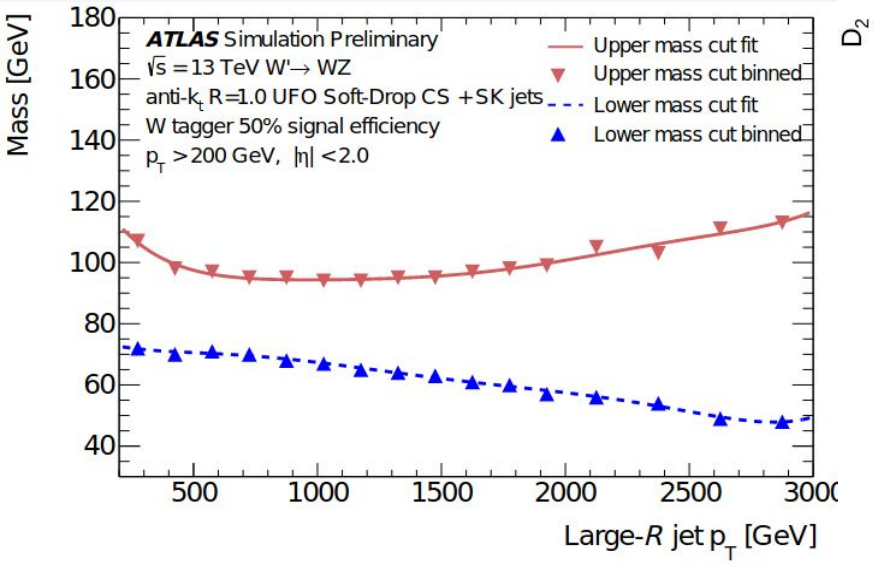


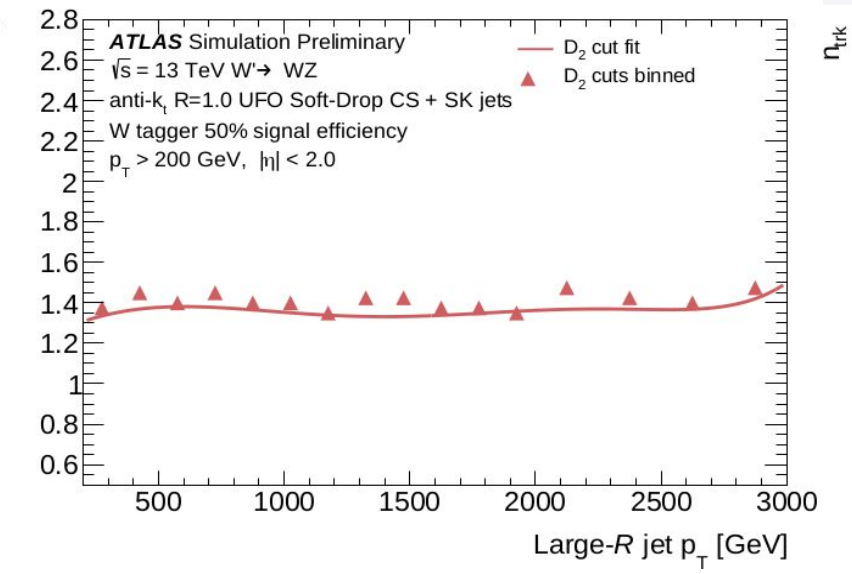
Signal

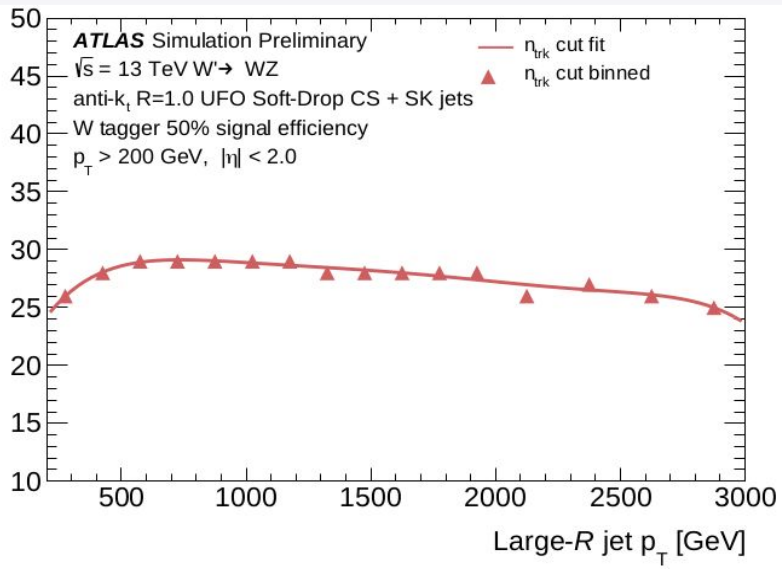


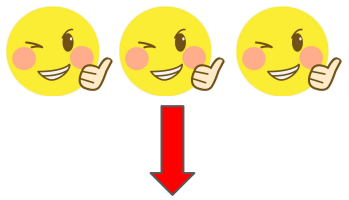
How we identify Boosted boson now?

Currently is used a tagger that perform cuts on **3 Jet substructure variables.** These cuts are made according to the Jet transverse momentum (**pT**).









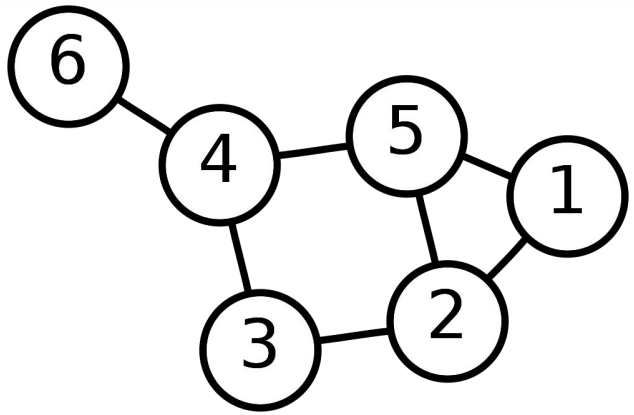
Jet Tagged!!

Plots taken from: **Performance of W/Z taggers using UFO jets in ATLAS**
https://cds.cern.ch/record/2777009/files/ATL-PHYS-PUB-2021-029.pdf

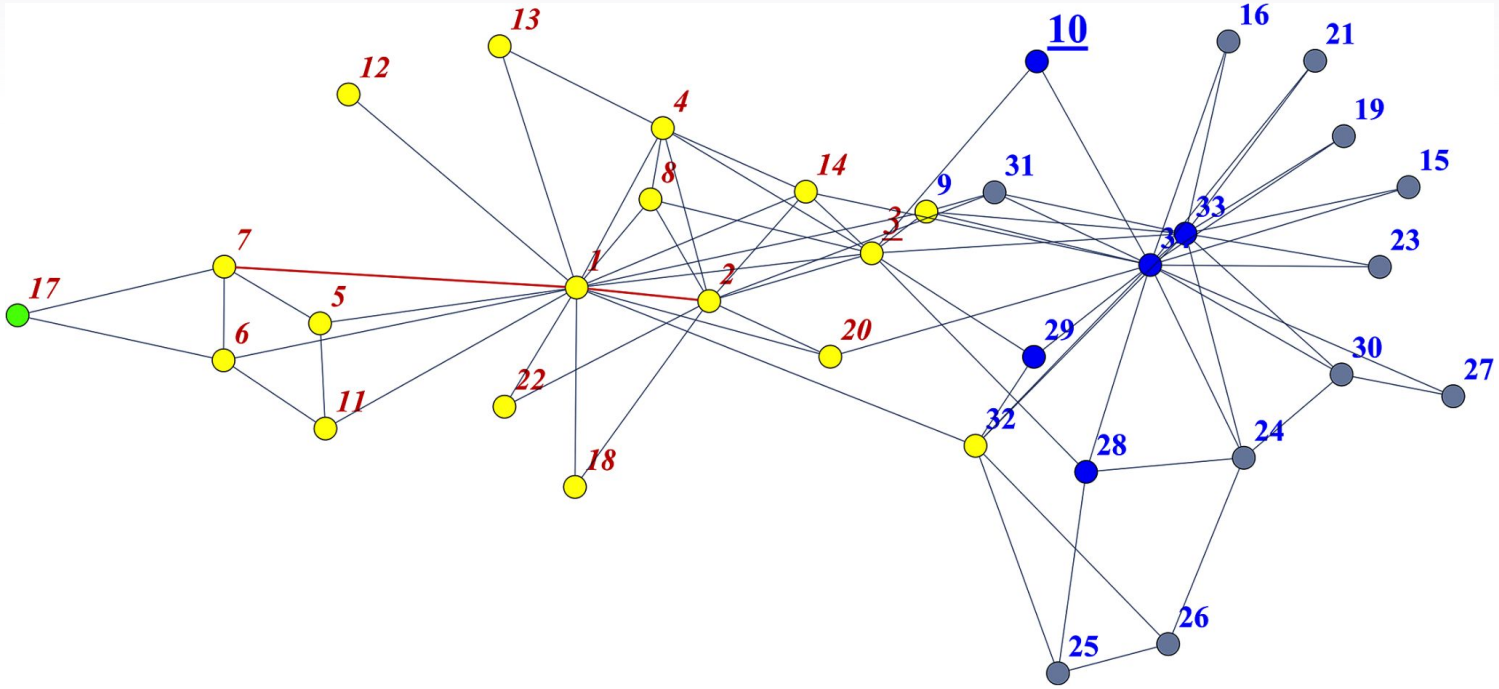
Graph Neural Networks

Garphs: “mathematical structures used to model pairwise relations between objects. A graph in this context is made up of **vertices** (also called *nodes* or *points*) which are connected by **edges** (also called *links* or *lines*)”

Wikipedia

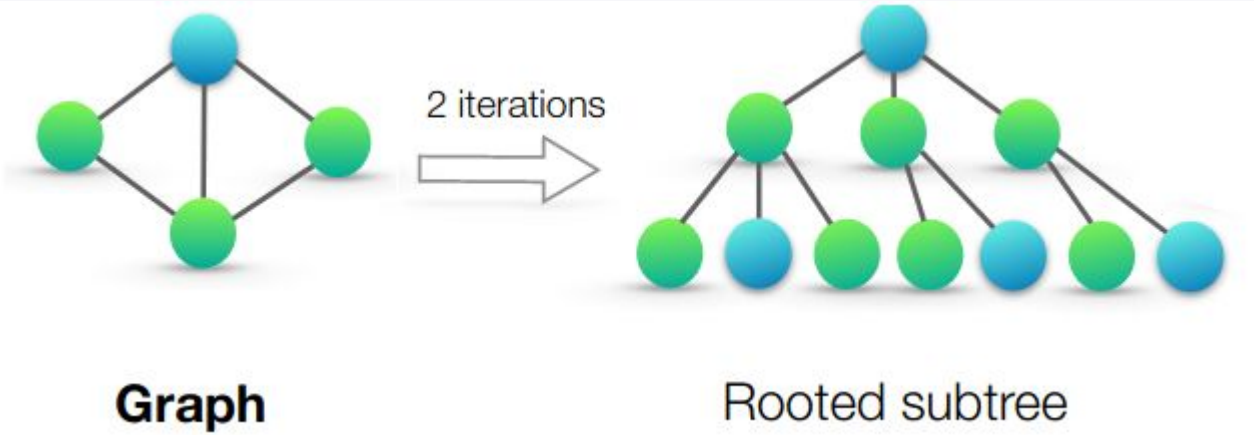


A graph with six vertices and seven edges.



Graph Neural Networks

We begin by summarizing some of the most common GNN models and, along the way, introduce our notation. Let $G = (V, E)$ denote a graph with node feature vectors X_v for $v \in V$. There are two tasks of interest: (1) *Node classification*, where each node $v \in V$ has an associated label y_v and the goal is to learn a representation vector h_v of v such that v's label can be predicted as $y_v = f(h_v)$; (2) *Graph classification*, where, given a set of graphs $\{G_1, ..., G_N\} \subseteq \mathcal{G}$ and their labels $\{y_1, ..., y_N\} \subseteq \mathcal{Y}$, we aim to learn a representation vector h_G that helps predict the label of an entire graph, $y_G = g(h_G)$.



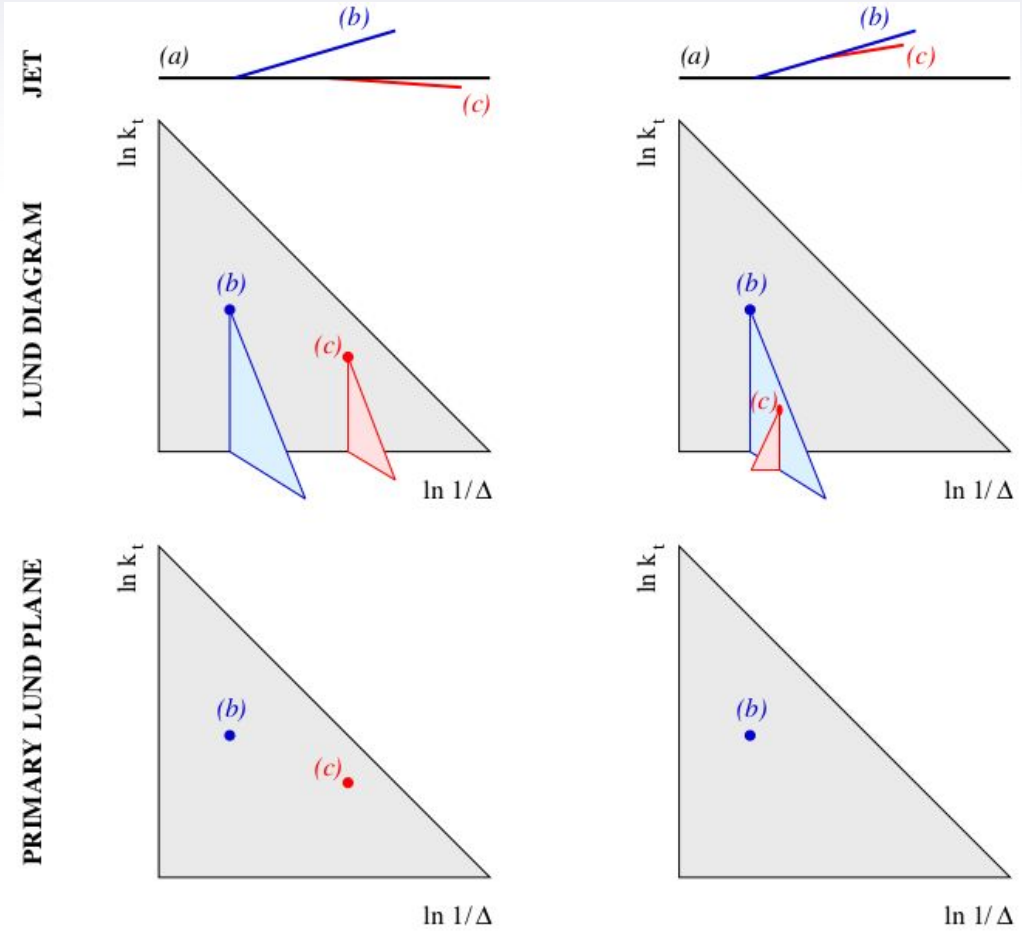
After k iterations of aggregation, a node is represented by its transformed feature vector, which captures the structural information

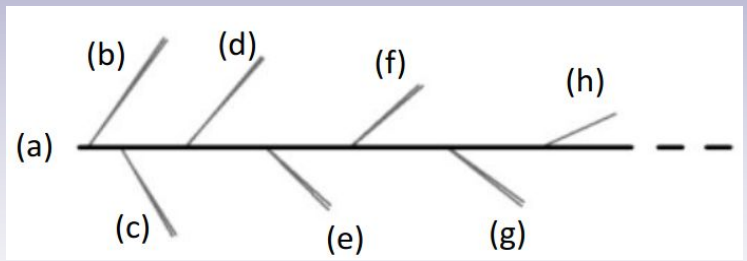
$$h_G = \mathrm{READOUT}\left(\left\{h_v^{(K)} \mid v \in G\right\}\right).$$

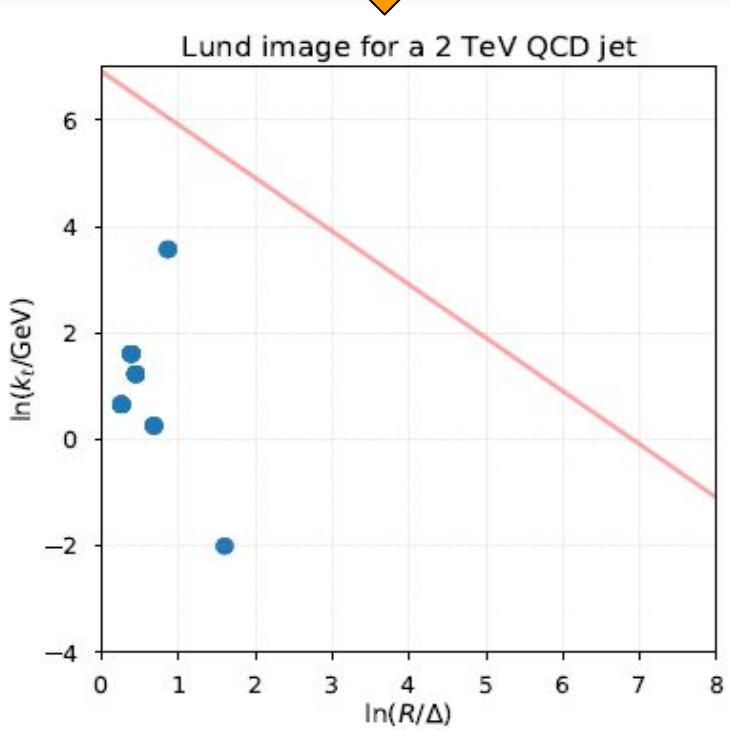
Plots and definitions taken from: **HOW POWERFUL ARE GRAPH NEURAL NETWORKS?** https://arxiv.org/pdf/1810.00826.pdf

Lund Plane

Lund plane: Is a way to represent the phase space of jet constituents reconstructed by reversing jet clustering sequence.



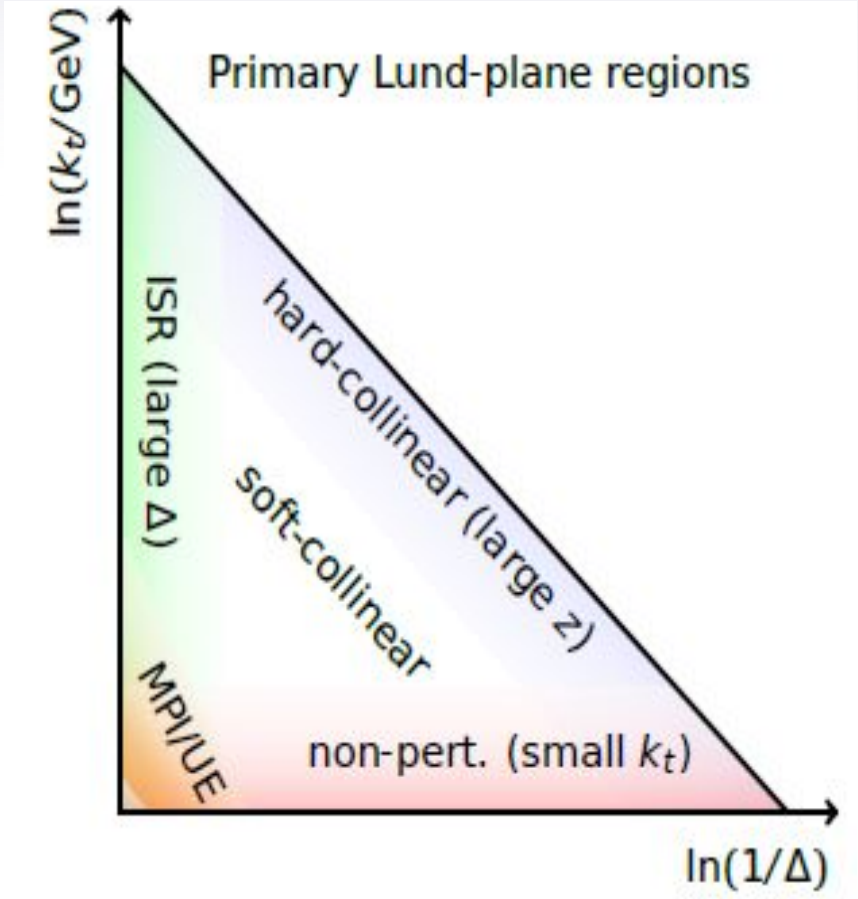




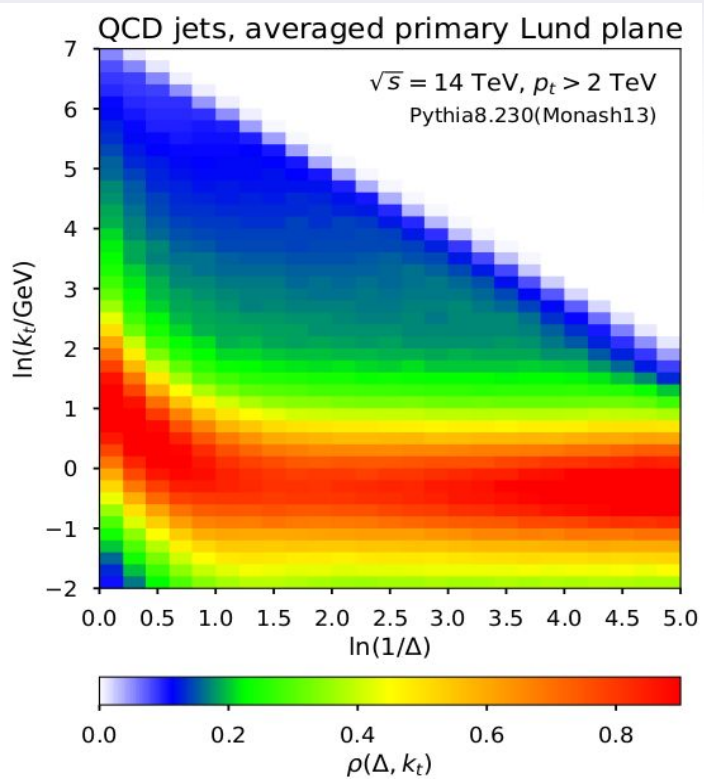
Plots taken from: **Dreyer, F.A., Salam, G.P. and Soyez, G. (2018). The Lund jet plane**. https://arxiv.org/pdf/1807.04758.pdf

Lund Plane regions

Using **ln(Kt)** and **ln(1/Δ)** we can identify differents regions.



Lund Plane

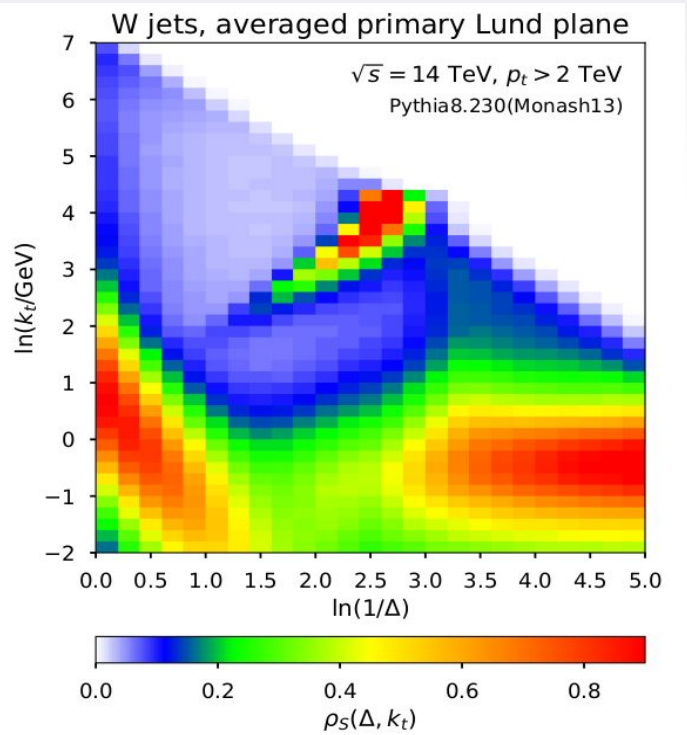


- **Great to separate QCD and W-jets**

- **Lund plane variables:**
  - **kT** : Transverse momentum of the emission.
  - **Δ** : Emission angle
  - **Z** : Momentum fraction of branching

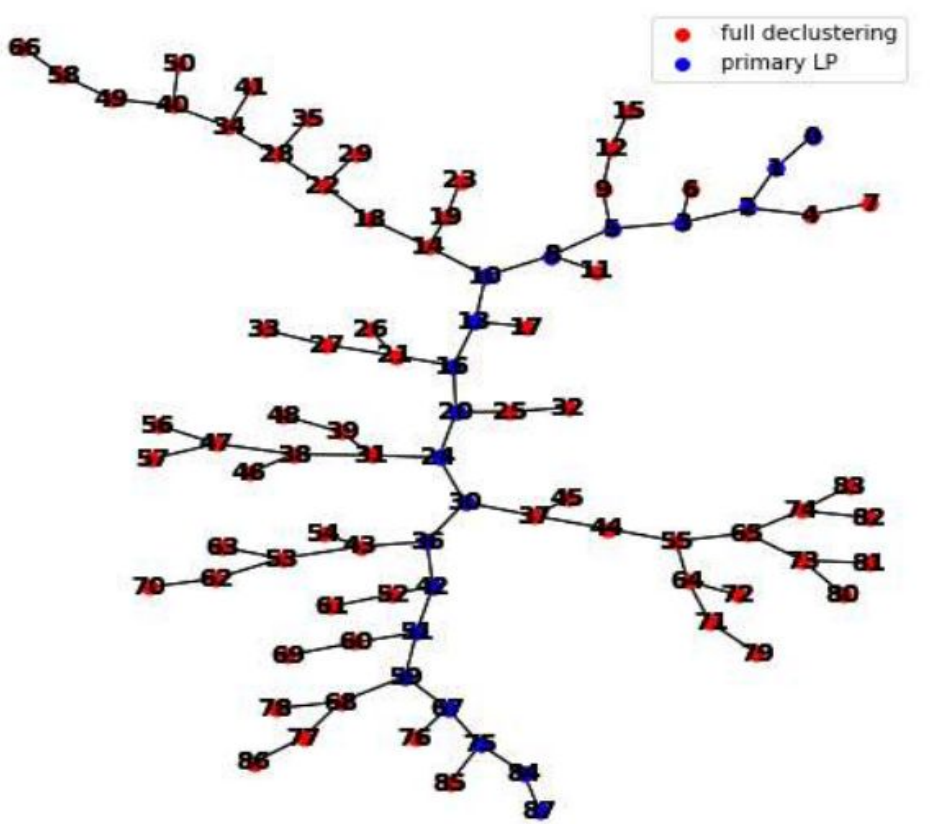
$$\Delta \equiv \Delta_{ab}, \quad k_t \equiv p_{tb}\Delta_{ab}, \quad m^2 \equiv (p_a + p_b)^2,$$

$$z \equiv \frac{p_{tb}}{p_{ta}+p_{tb}}, \quad \kappa \equiv z\Delta, \quad \psi \equiv \tan^{-1}\frac{y_b - y_a}{\phi_b - \phi_a},$$



Plots taken from: **Dreyer, F.A., Salam, G.P. and Soyez, G. (2018). The Lund jet plane**. https://arxiv.org/pdf/1807.04758.pdf

Full Lund plane



- Using the Lund Plane we are going inside the hadronization history. Every single emission is represented!
- If is used the information of each emision instead of using jet global variables we can do a better background discrimitation

More information used → **Better performance**

- Our inspiration comes from Lund Net https://arxiv.org/pdf/2012.08526.pdf
- Lund planes is made up as a set of vertices and their connection edge, so this is an ideal input for Graph Neural Networks!

Models (GNN architectures)

Traditional Neural Networks require input to be of fixed length whilst Graph Neural Networks do not have this limitation, whether the input graph has 2 nodes or 20, the GNN model can handle it!

GNN architectures

- LundNet (https://arxiv.org/pdf/2012.08526.pdf) ⟹ **our inspiration**
- Graph Isomorphism Network (GINConv)
- Graph Attention Network (GATConv)
- Gated Graph Sequence Neural Network (GatedGraphConv)
- Principal Neighbourhood Aggregation Network (PNAConv)

All documented as GINConv, GATConv, GatedGraphConv, and PNAConv, respectively, at :
https://pytorch-geometric.readthedocs.io/en/latest/modules/nn.html#convolutional-layers

Dataset

Events were generated using Monte Carlo simulations in Powheg and Pythia 8 and the detector is simulated using Geant4. Precisely, this is the data used:

- Dijets: 8,277,229 total events.
  mc16_13TeV.3647[03,09].Pythia8EvtGen_A14NNPDF23LO_jetjet_JZ[03,09]WithSW.deriv.DAOD_JETM8.e7142_s3126_r10201_p4355
- W prime (W boson): 3,343,338 total events.
  mc16_13TeV.426347.Pythia8EvtGen_A14NNPDF23LO_WprimeWZ_flatpT.deriv.DAOD_JETM8.e6880_s3126_r10201_p4355

Train size:

10% of the Dataset used for training - 90% of the Dataset used for testing.

Cut-based tagger

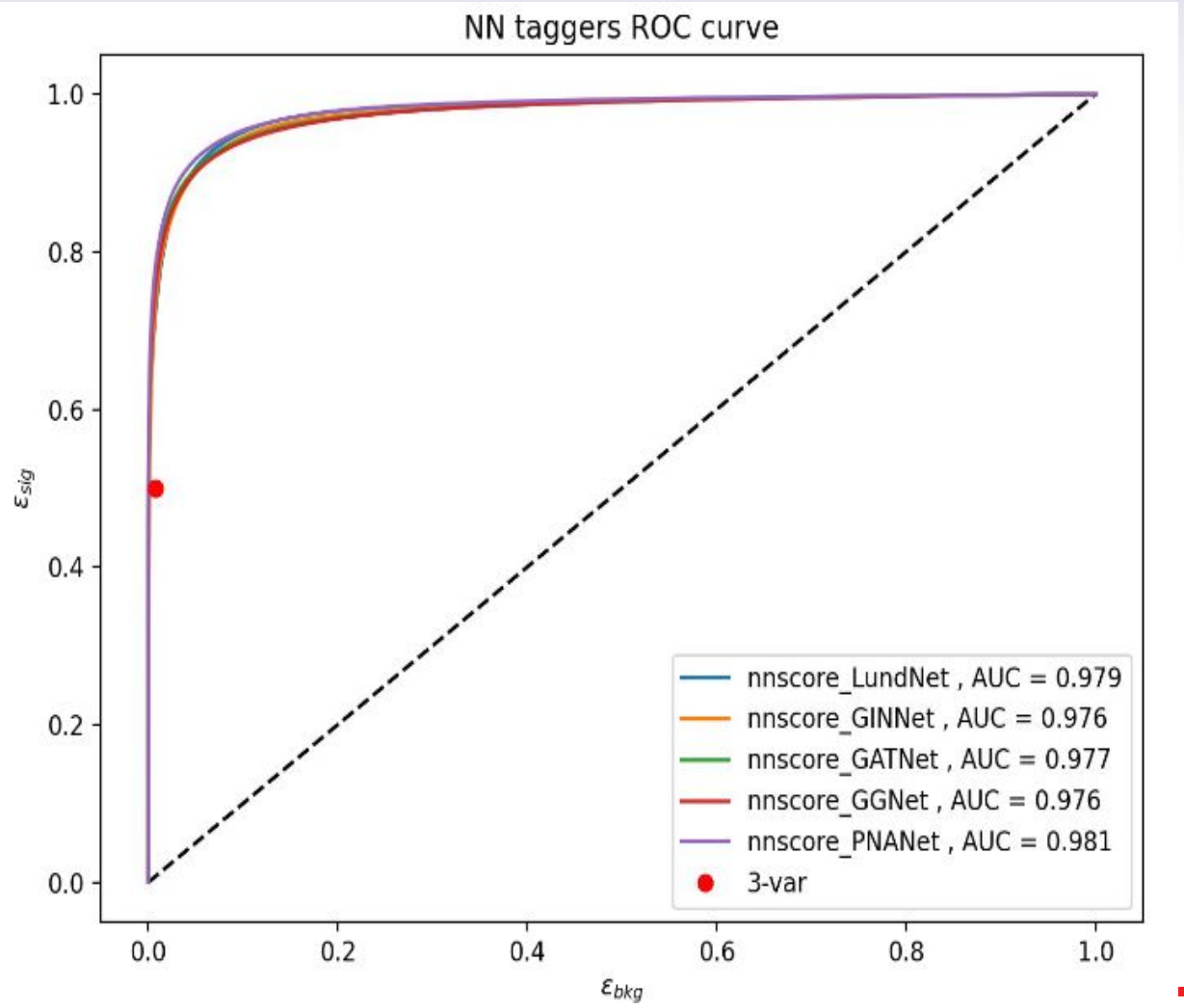
- Jet pT > 200 GeV
- 50 GeV < Jet mass < 100 GeV
- | Jet eta | < 2
- Jet truth match with W boson
- $|\Delta R| < 0.75$
- Number of b Hadrons = 0

pT > 200 GeV

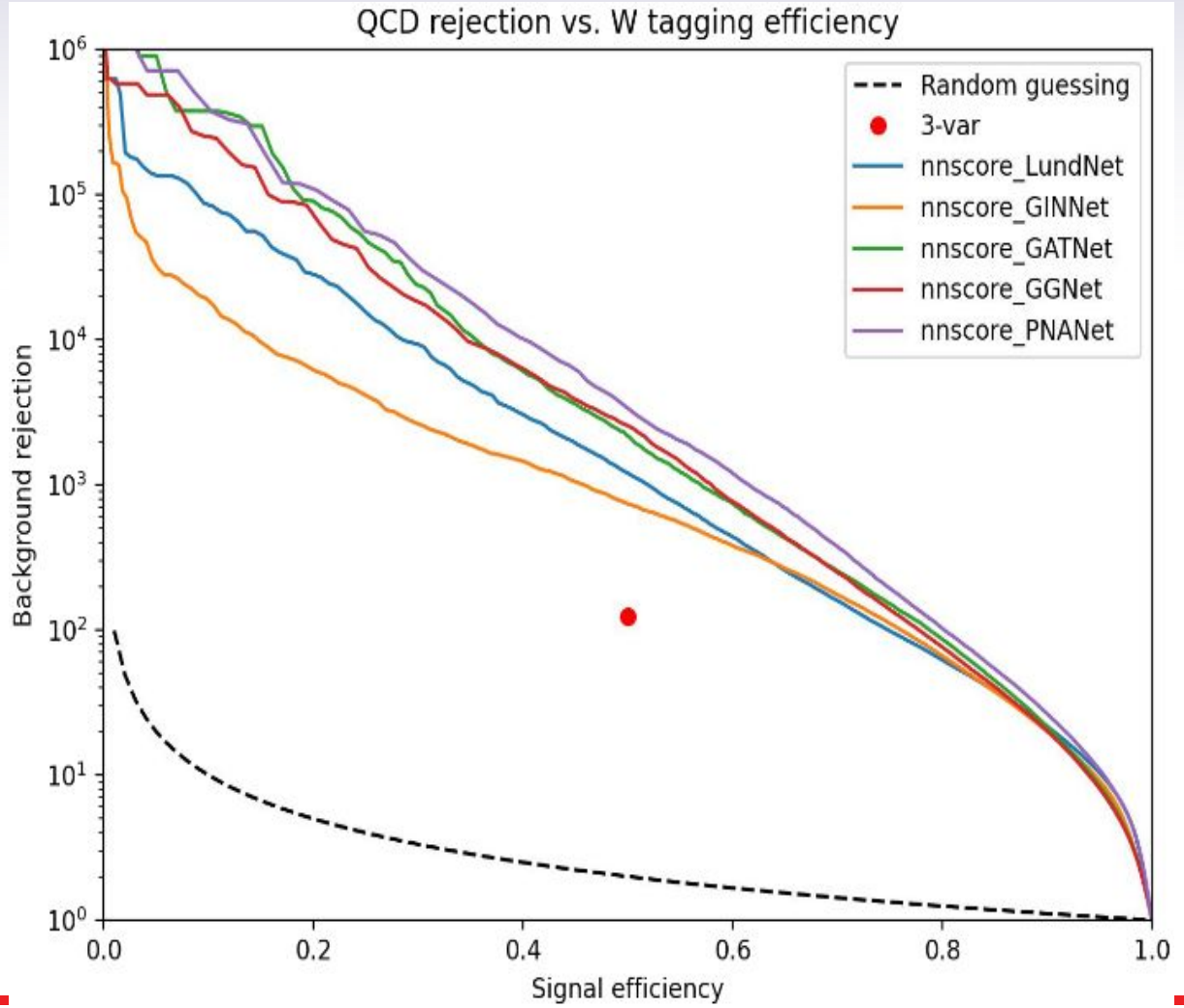
$$\epsilon_{\text{signal}} = \frac{N^{\text{identified}}_{\text{signal}}}{N^{\text{total}}_{\text{signal}}} \longrightarrow \text{“Signal efficiency”}$$

$$\frac{1}{\epsilon_{\text{background}}} = \frac{N^{\text{identified}}_{\text{background}}}{N^{\text{total}}_{\text{background}}} \longrightarrow \text{“Background rejection”}$$

NN taggers ROC curve

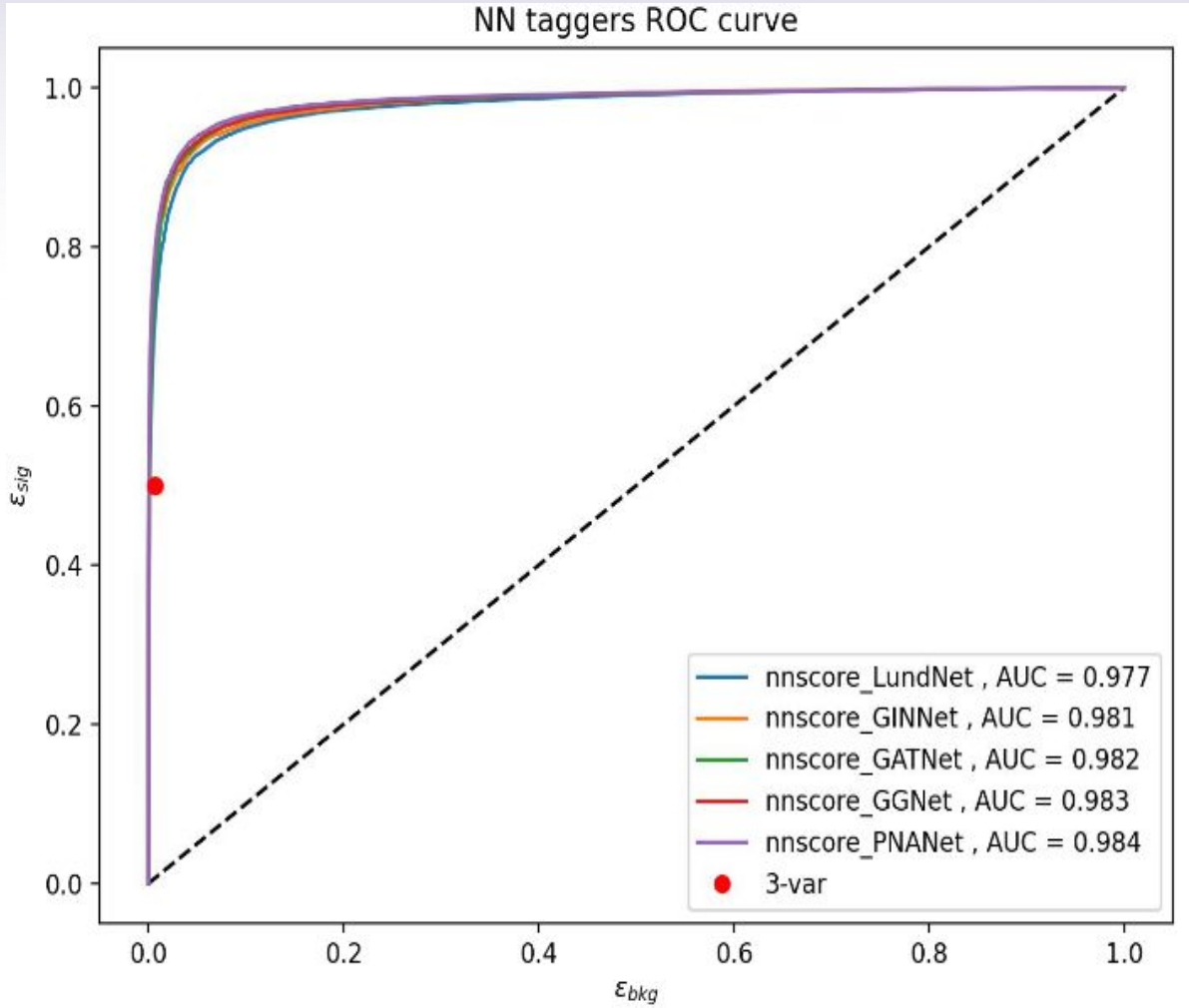


QCD rejection vs W tagging efficiency

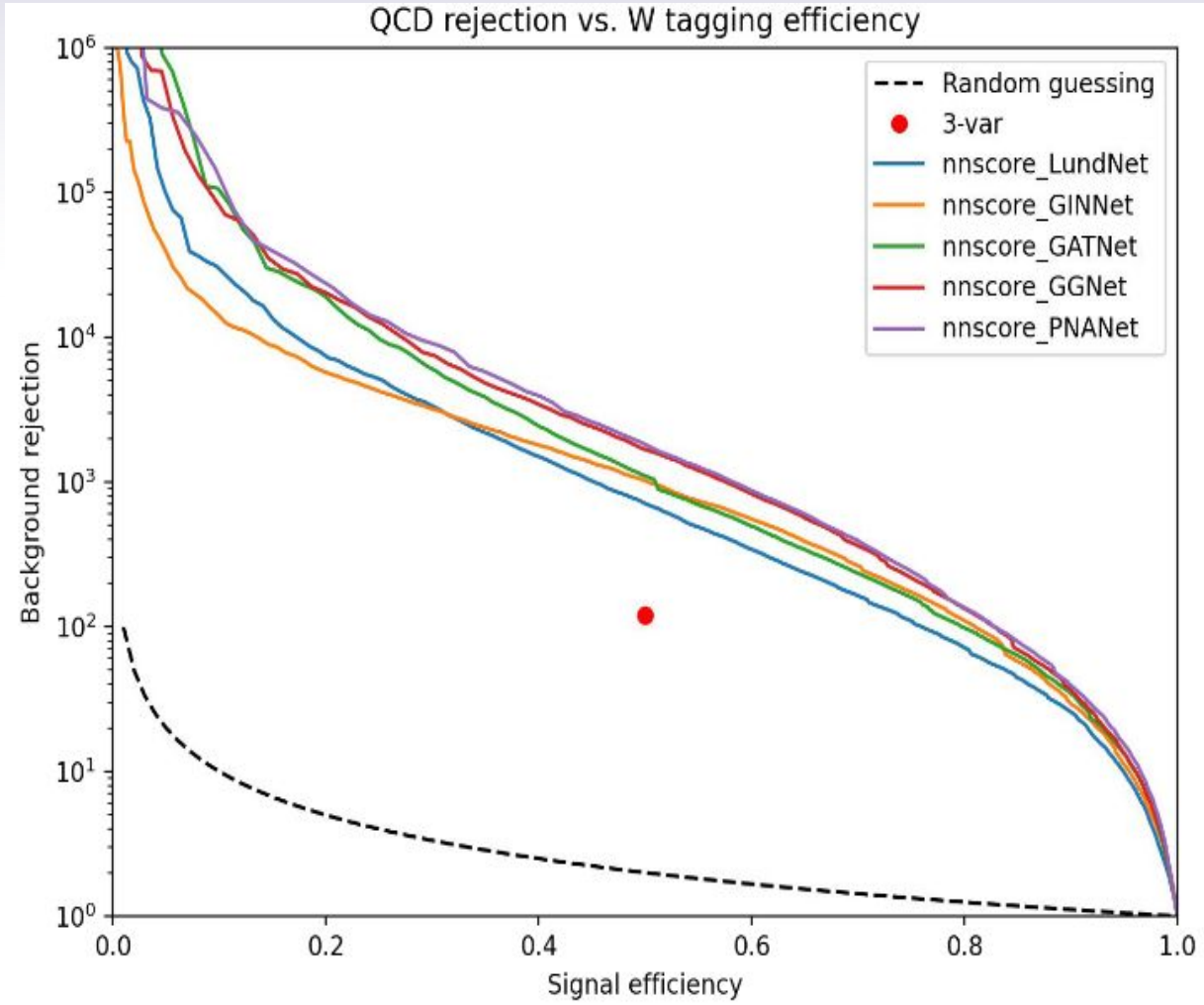


pT > 500 GeV

NN taggers ROC curve

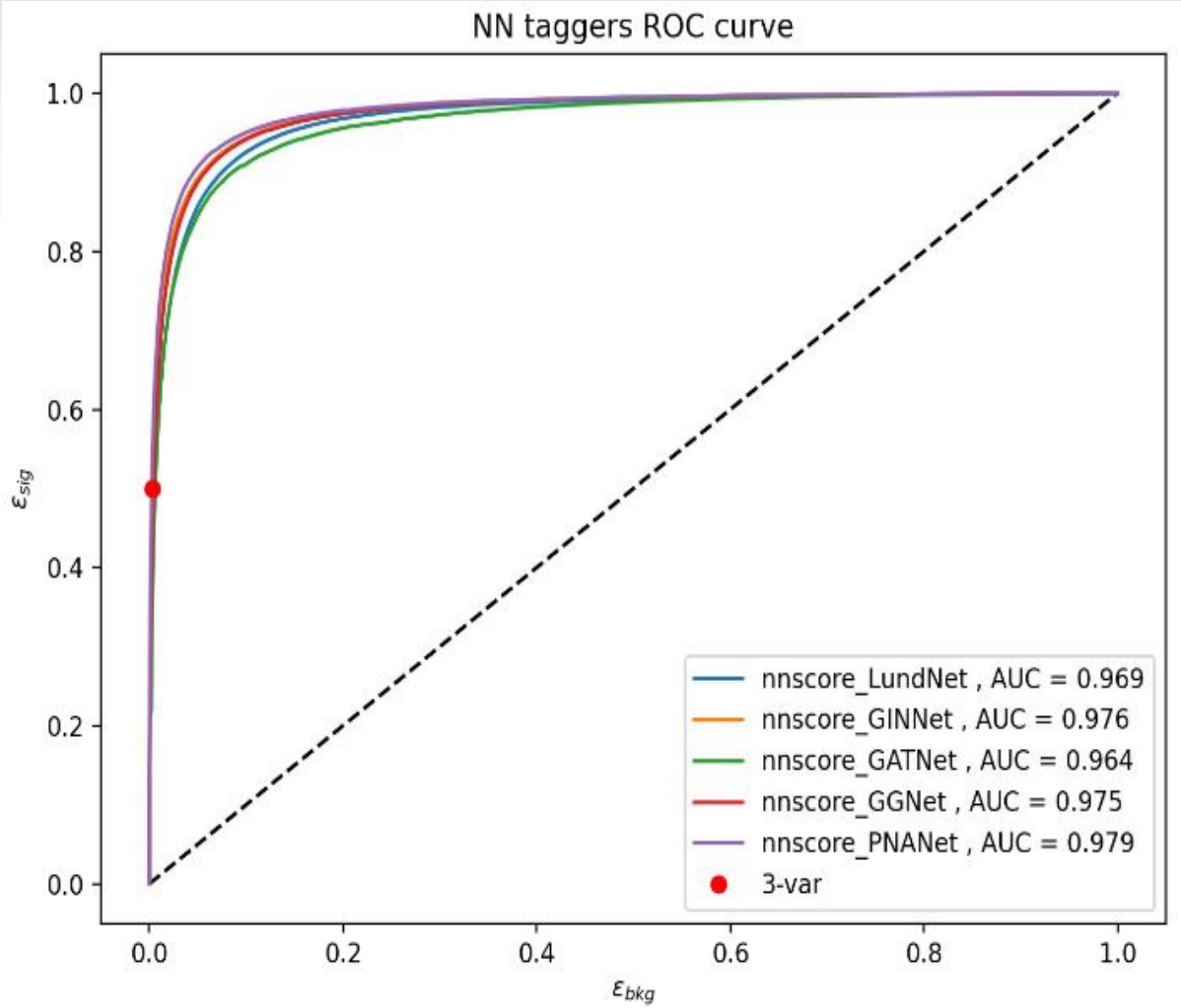


QCD rejection vs W tagging efficiency

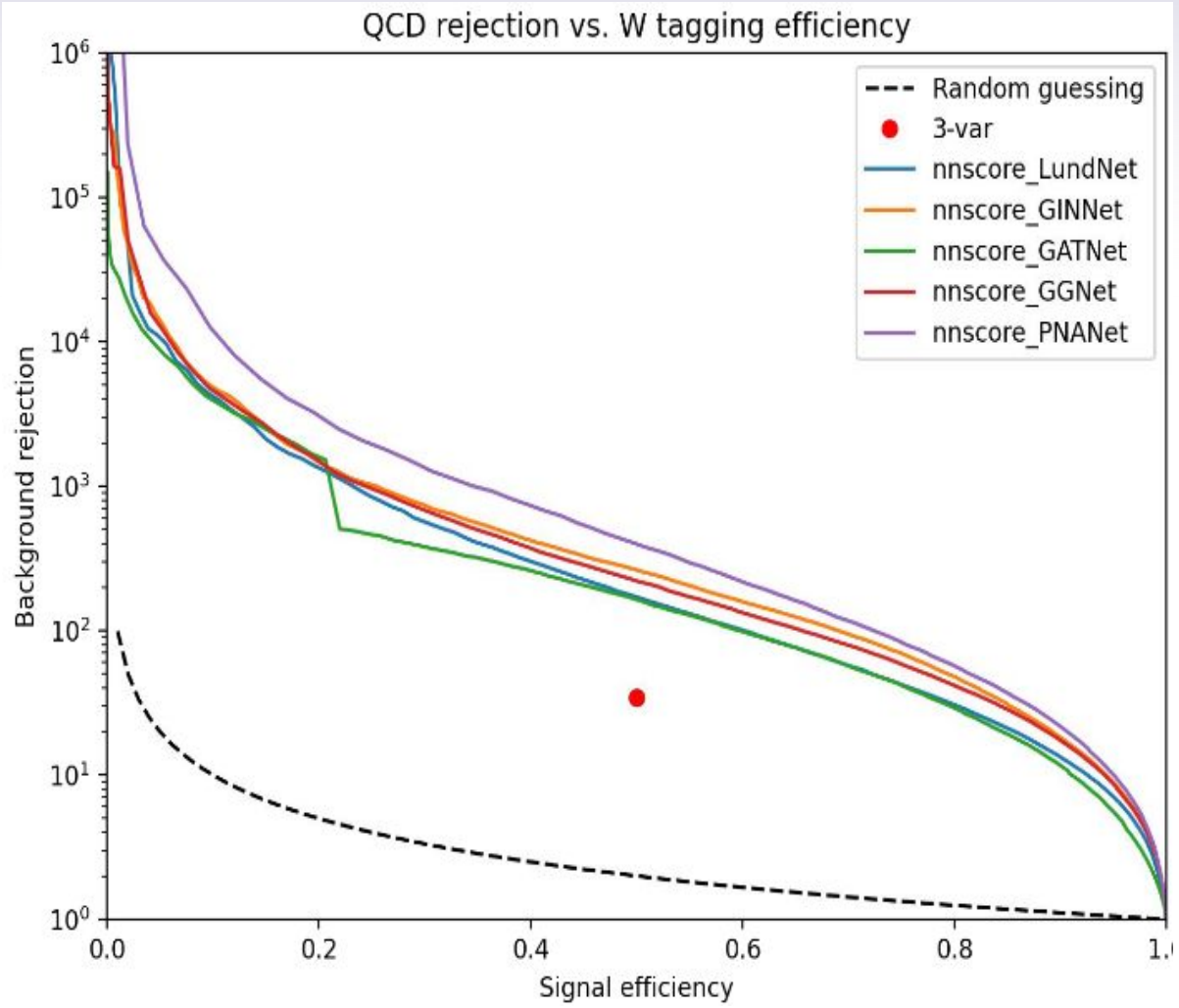


pT > 2000 GeV

NN taggers ROC curve



QCD rejection vs W tagging efficiency



Conclusions

- Presented 5 GNN architectures with improved performance over the currently boosted W boson taggers.

Work in progress

- Optimization in the implementation of the differents models.

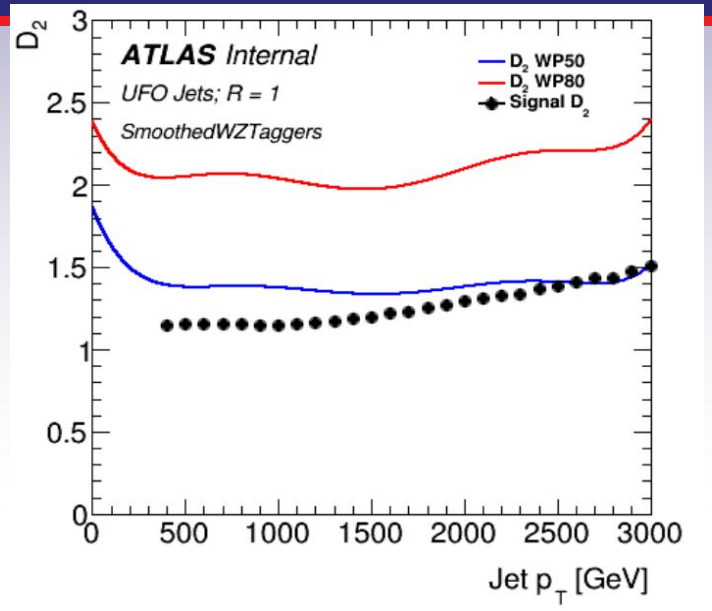
A gentle introduction ;)

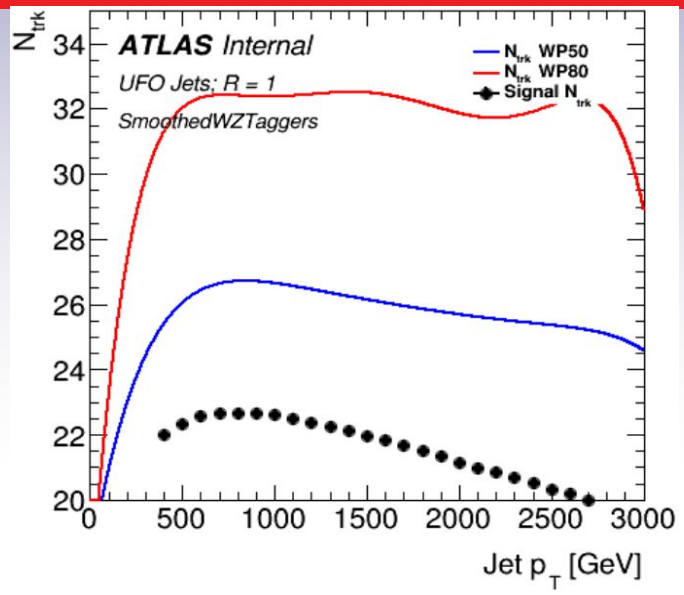
https://www.youtube.com/channel/UCxw9_WYmLqlj5PyXu2AWU_g/featured

Thanks for your attention :)

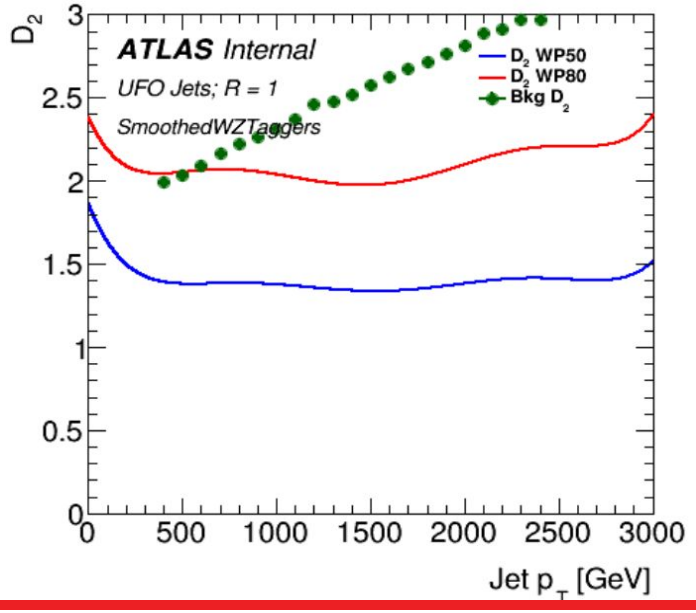
Standard Tagger performance

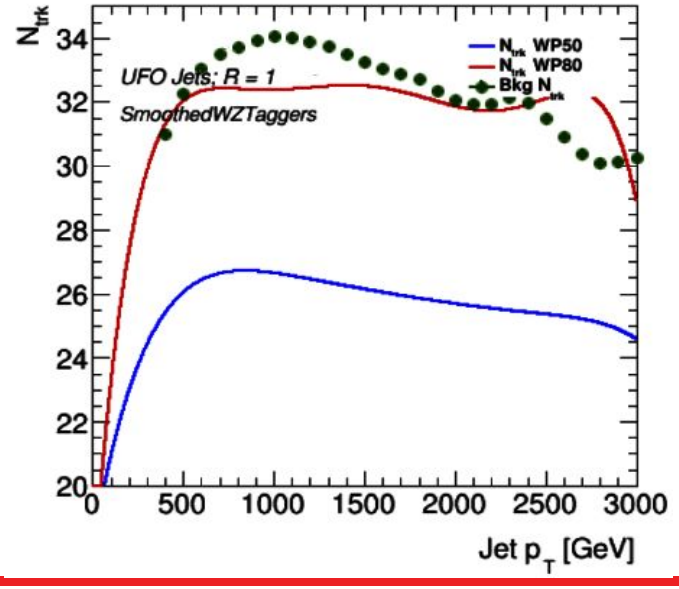
Signal performance





Background performance





New Selection

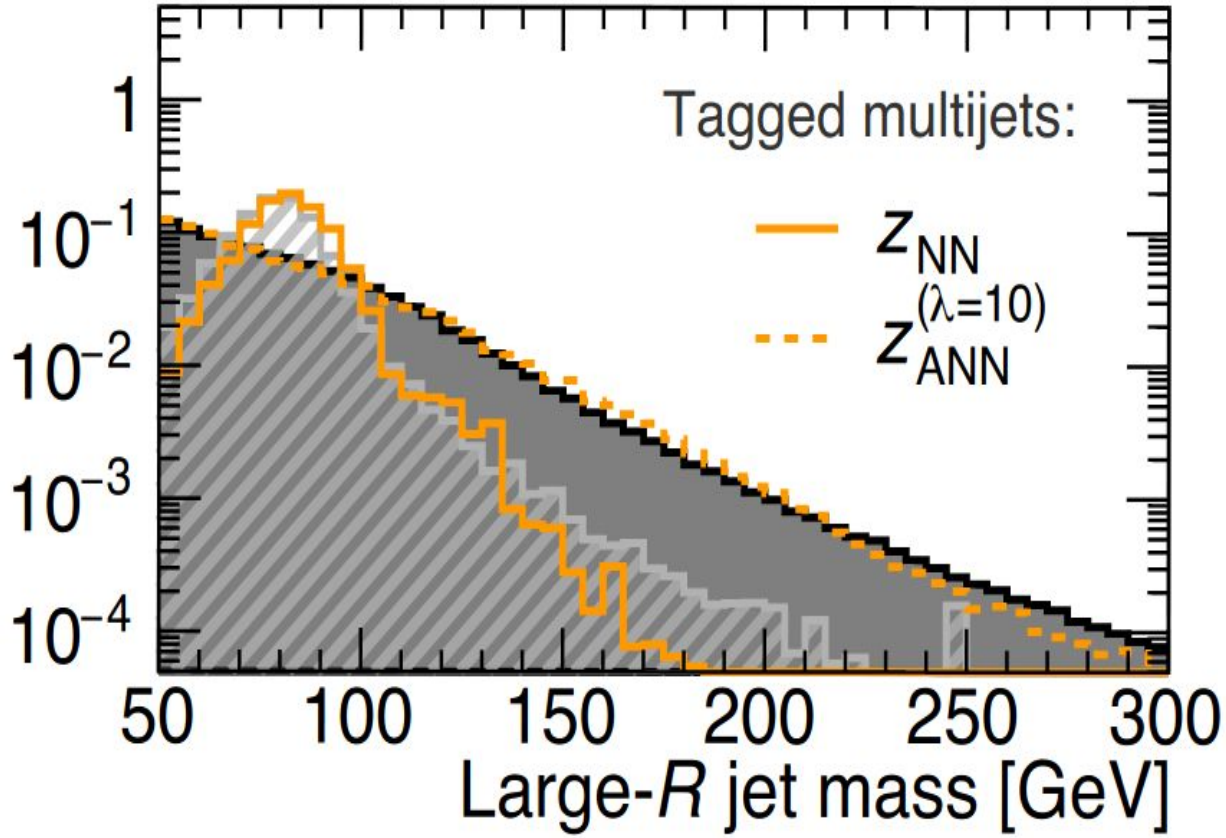
Signal and Background cuts:

Ungroomed Jet_pt > 200 GeV,
Jet_pt > 200 GeV,
Jet_pt < 3000 GeV,
Jet_mass > 40 GeV,
Jet_mass < 300 GeV,
Jet_D2 > 0,

Signal definition

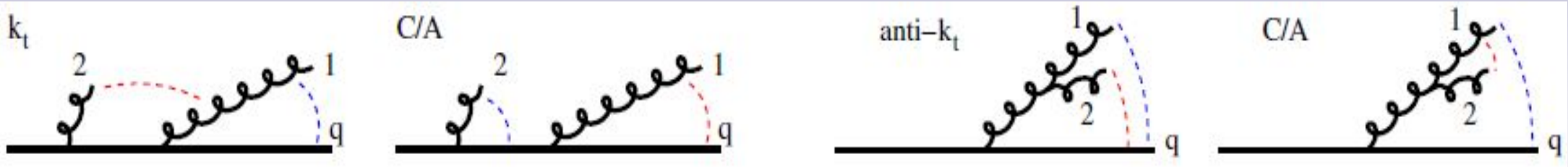
Jet truth match with W boson
Ungroomed Jet_mass > 50 GeV
Number of b Hadrons = 0

Mass sculpting



After the selection the mass profile of the background signal changed! To avoid that we could use an Adversarial Neural Network!

Declustering algorithms



- The Declustering algorithms tries to go inside the hadronization history in order to determine where each emission is coming from.

Contribuciones a NLO:

$$\bar{\rho}_2^{(k_t)}(\Delta,\kappa) \simeq -4C_F^2 \ln^2 \frac{\Delta}{\kappa} + \mathcal{O}(L) \; .$$ → Kt algorithm

$$\bar{\rho}_2^{(\text{anti-}k_t)}(\Delta,\kappa) \simeq +8C_F C_A \ln^2 \frac{\Delta}{\kappa} + \mathcal{O}(L) \; .$$ → Anti-Kt algorithm

$$\bar{\rho}_{2,\text{rc}}^{(\text{C/A})}(\Delta,\kappa) = \bar{\rho}_1(\Delta,\kappa)\, 4\pi b_0 \ln \frac{1}{\kappa} + \mathcal{O}(1) \; .$$ → C/A algorithm

http://laconga.redclara.net

contacto@laconga.redclara.net

laconga physics

Latin American alliance for
Capacity buildiNG in Advanced physics

LA-CoNGA physics

Cofinanciado por el
programa Erasmus+
de la Unión Europea

El apoyo de la Comisión Europea para la producción de esta publicación no constituye una aprobación del contenido, el cual refleja únicamente las opiniones de los autores, y la Comisión no se hace responsable del uso que pueda hacerse de la información contenida en la misma.