DEPARTMENT OF COMPUTER SCIENCE

UNIVERSITY OF OXFORD

03/08/2020

Intermediairy Presentation
ML for Quark- and gluon-jet discrimination

Maxence DRAGUET

Supervisors

Ani CALINESCU Alan BARR

Aaron O'NEILL

Objectives

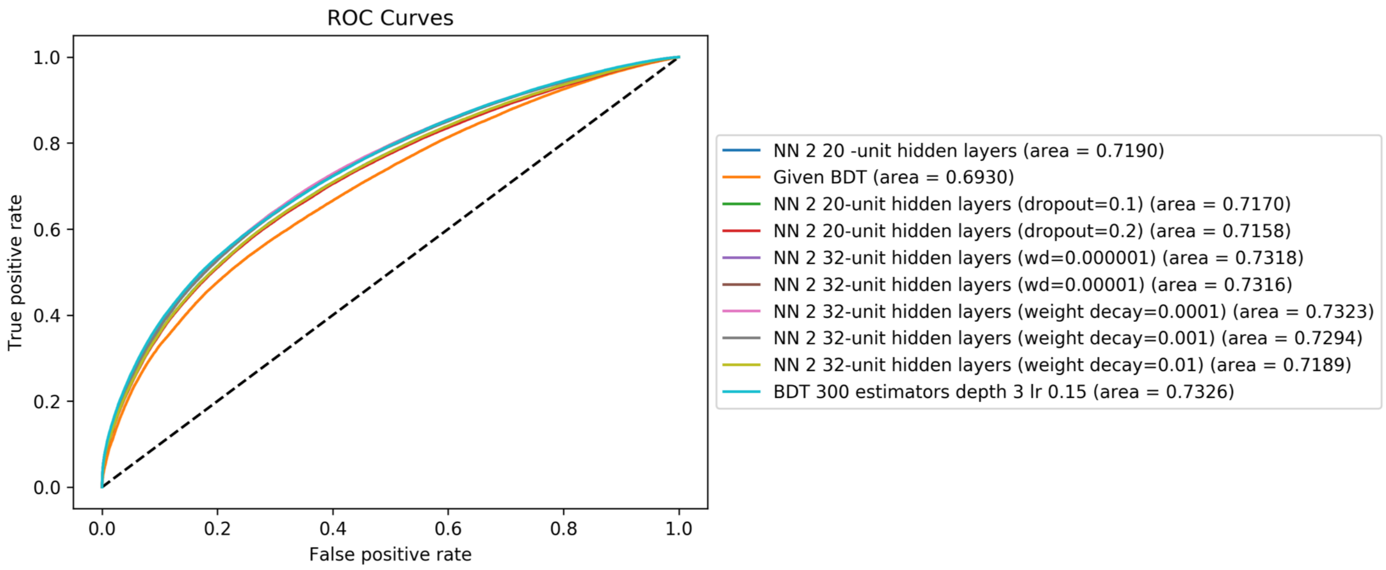
- Improve the efficiency of the quark vs gluon jet [Q-G] discrimination task.
- Restrain to fast processing and low-level inputs close to the events: idea of implementing this for complex online trigger.
- Explore machine learning solutions with a focus on performance and interpretability.

Context

- First step: develop a set of tools commonly employed in this task for benchmarking:
  1. Boosted Decision Tree using jet variables [BDT].
  2. Multi-layered perceptron (Connected network with dropout) [NN].

  Area Under the Curve of ROC curve typically achieved (precision is in the order of 68% to 74%):
  1. BDT of 300 estimators of depth 3: AUC = 0.73
  2. NN of 2 32-unit hidden layers: AUC = 0.73

- **Problems** with both methods:
  1. performance plateau too soon,
  2. 'highly engineered' input variables,
  3. difficult to interpret,
  4. domain knowledge limited to variable,
  5. limited to discrimination.

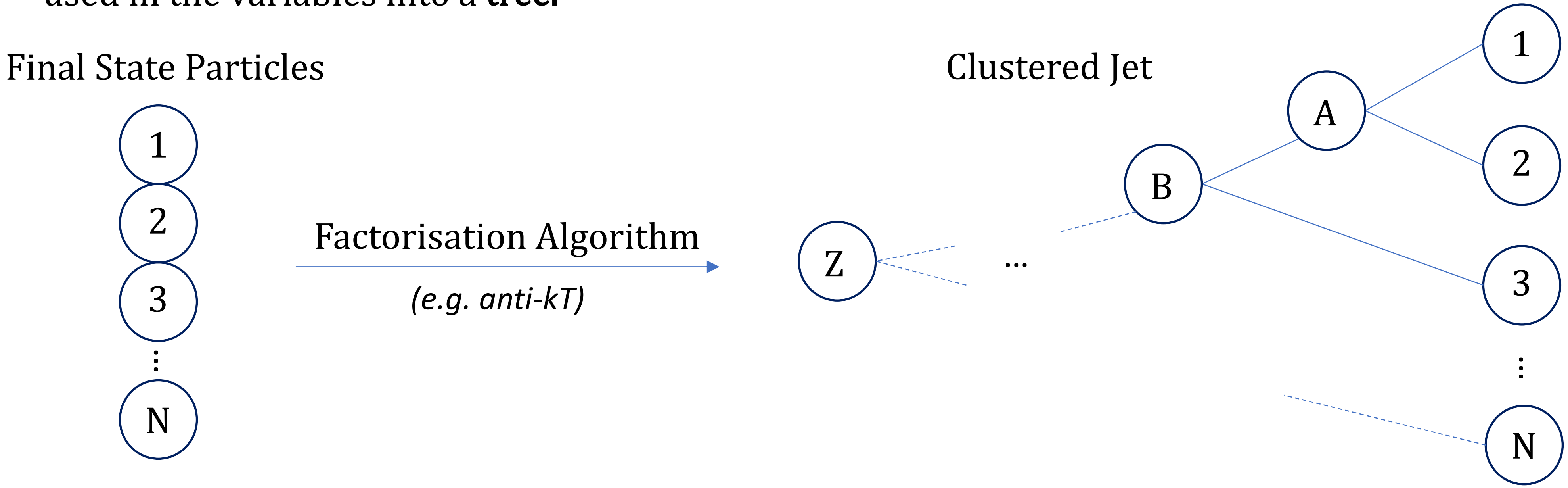


(*) Full list = jetPt, jetEta, jetPhi, jetMass, jetEnergy, jetEMFrac, jetHECFrac, jetChFrac, jetNumTrkPt500, jetNumTrkPt1000, jetTrackWidthPt500, jetTrackWidthPt1000, jetSumTrkPt500, jetSumTrkPt1000

Idea

- The 'highly-engineered' variables are derived following the factorisation of constituents observed in the detector into a jet: N final state particles are clustered into a jet (typically using the anti-kT algorithm)

- But ... this factorisation algorithm assigns a recurrent structure to the event* encoding the rich information used in the variables into a **tree.**



- Why not **hijack the clustering step to tag** in the same time **using this recurrently-encoded information**?

  This is the **idea behind JUNIPR [1].** Was **shown to be state-of-the-art for Q-G tagging at generated level [2].**

(*) Here, event is synonymous with jet. Events with x jets are considered as x events of single jet. Note that nothing forces this factorisation.
The objective is however to analyse quark and gluon jets, not jet structures emanating from specific processes hence the factorisation.

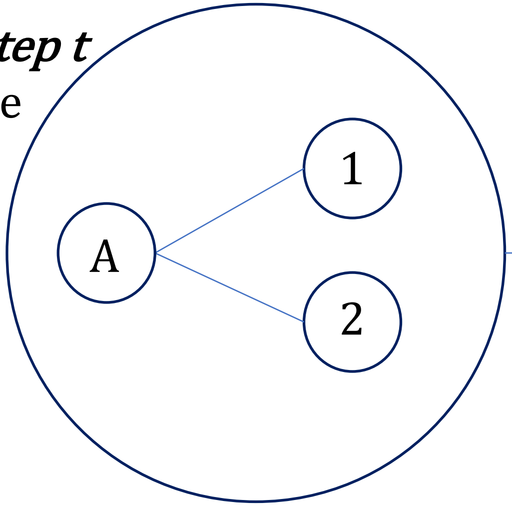
JUNIPR I

- JUNIPR: *Jets from UNsupervised Interpretable PRobabilistic models.*

- Is **in fact weakly-supervised**: train a model on a dataset that does not require label, but that is enriched in a specific kind of events (that differs for each training dataset).

- In training, JUNIPR learns the likelihood for a jet to be produced by the model (the training dataset) by maximising it. To ease training and analyse, the **likelihood is factorised** on the recurrence structure (the **tree)**.

Branching *step t*
in the tree



Probability Factorised

1. P(There is a branching at step *t*)
2. P(A is mother at *t* | possible mothers at *t*)
3. P(p_1, p_2 daughter momenta at *t* | A is mother at *t*)

*

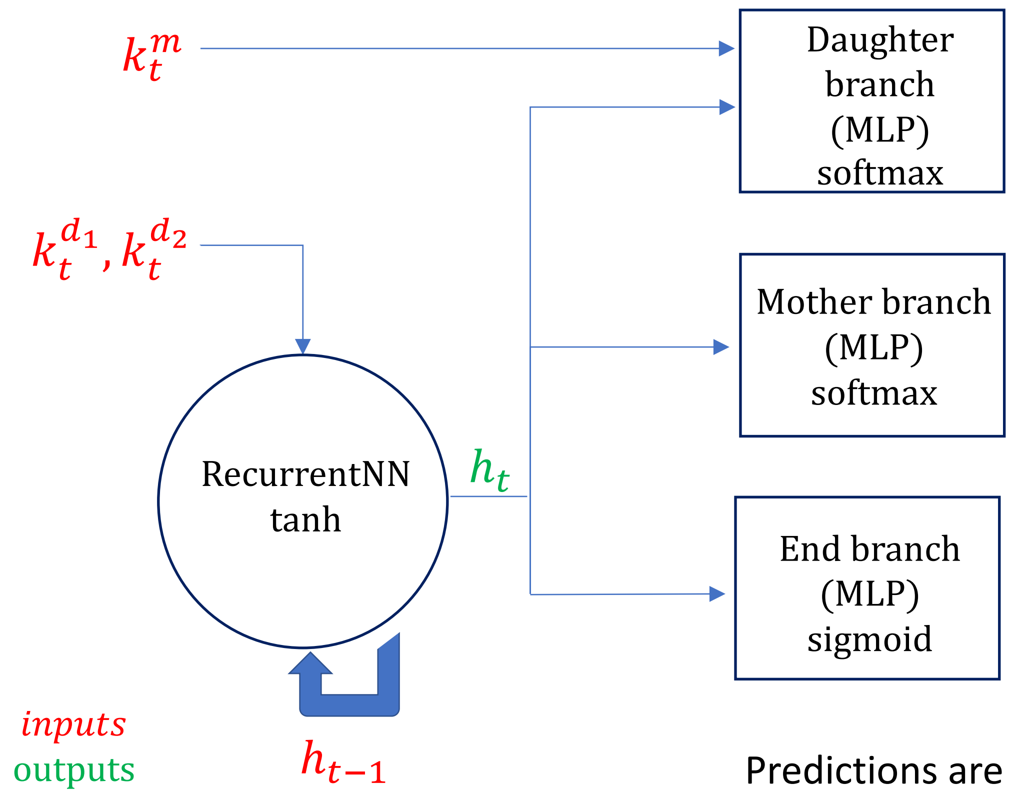
Product of these
=
Probability of Node

Product of all Node Probabilities
=
Probability of Tree

(*) Each of these probability is conditioned on the history before *step t* of the tree

JUNIPR II

- In practice, the evolving internal representation of the tree is obtained by a Recurrent Neural Network.
- At time step* *t*, the inputs of the model are the two daughter momenta $k_t^{d_1}, k_t^{d_2}$, the mother momentum k_t^m, and the hiddent state of the previous step h_{t-1}.



Represented by 4 numbers (z, t, p, d):

1. Energy fraction of the soft daughter,
2. angle of soft daughter to mother,
3. angle of plane of decay,
4. angle of the hard daughter.

Each is scaled to [0, 1] (based on average jet properties) and discretised to 10 bins. The daughter branch predicts the bins

In an energy ordered list of available particles, mother branch predicts the index of the next mother.

End branch predicts where the tree ends (1) or not (0) at this node.

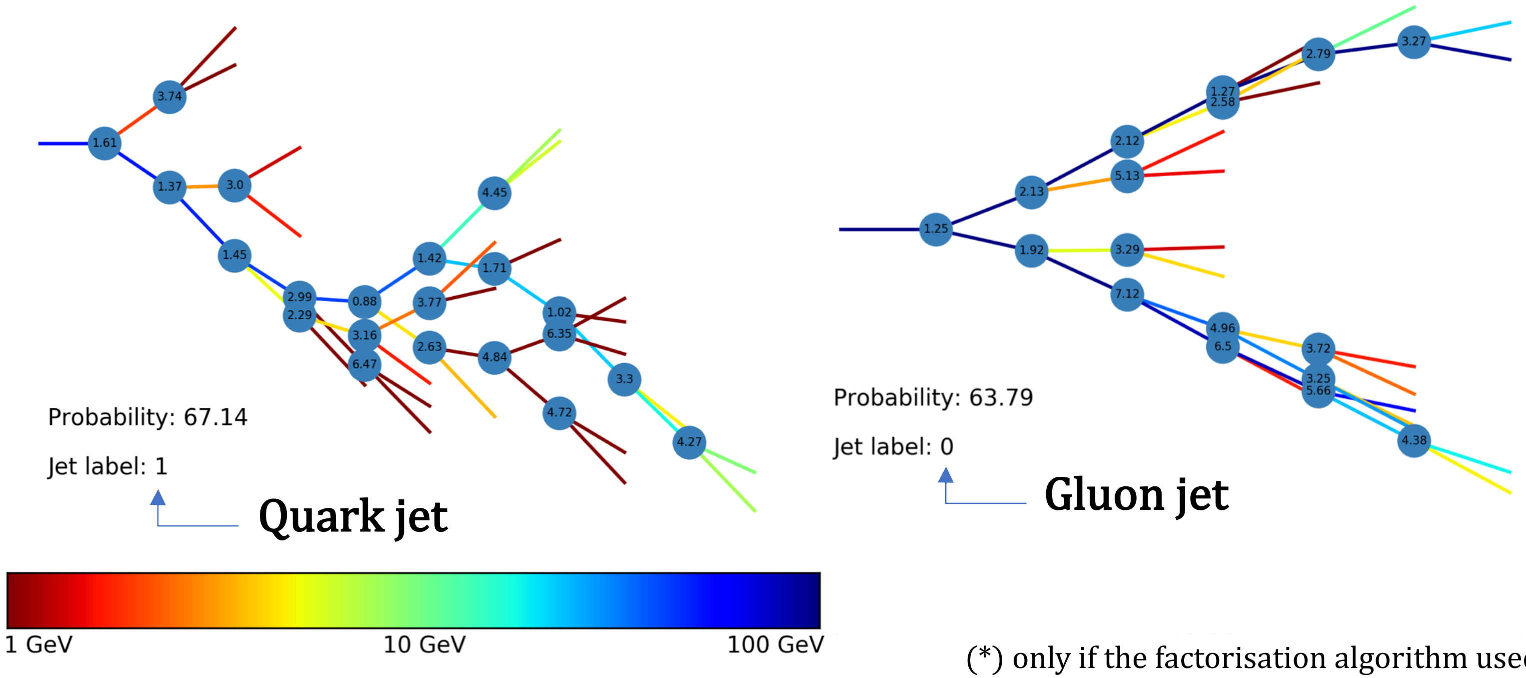
Predictions are turned into probability by computing the cross entropy with the input data.

(*) Several steps are particular. For step 1, the hidden state is an MLP transformation of the seed momentum (the very first mother). There is a final step when the tree ends: then only the probability that the tree indeed ends is computed (the mother and daughter branches are not sollicited).

JUNIPR III

- JUNIPR has many advantages over classical methods:
  1. It is a **probabilistic model.** It offers a likelihood ratio test that is the **most statistically powerful discriminant** (Neyman-Pearson Lemma) and can be used for **sampling jet trees and reweighting MC.**

  2. It is **structured around a leading order model of physics** and is therefore fully **interpretable***.
     Example: quark JUNIPR model trained for 50 epochs (on ttbar):



1. Colour shows the energy in log scale.

2. Angles are 2D visualisation of the real 3D ones.

3. Values in nodes are probabilities of each branching from JUNIPR (in –log).

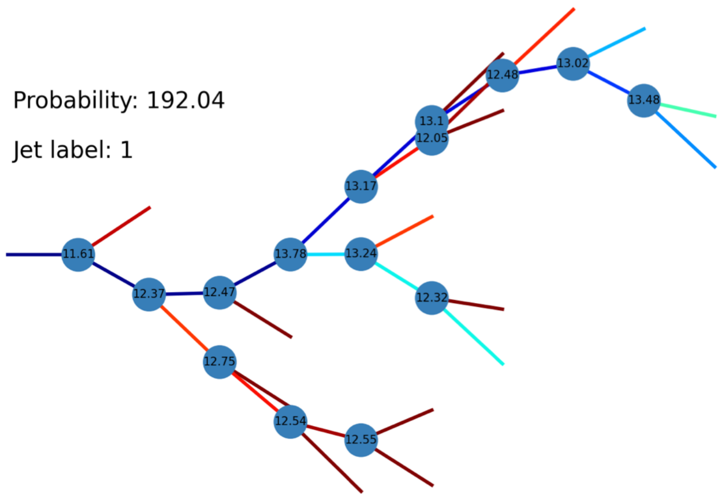
After binary JUNIPR, **node value** will be **likelihood ratio tests** and a simple **thresholding** on the global value should **offer the tag**.

(*) only if the factorisation algorithm used represent an interpretable model.

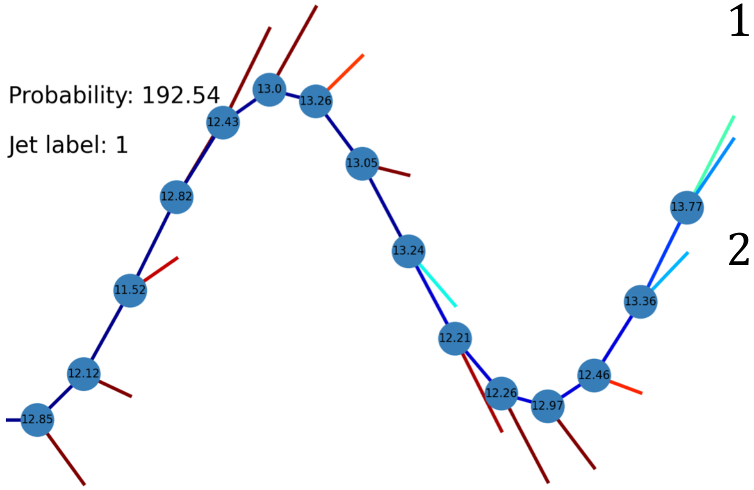
JUNIPR IV

- Note that the factorisation models is not forced to be "physical". However, performance will lightly depend on the model (interpretability is fully dependent on it).
- This is the **same** jet but it is re-clustered differently in JUNIPR* :

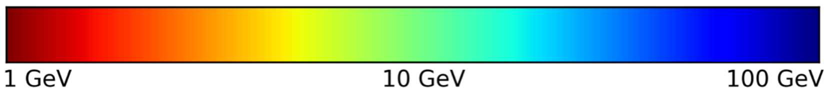
With the Cambridge-Aachen algorithm



With the anti-kT algorithm



1. The **hidden state of JUNIPR** will **have to internalise** far **more** information in the case of **anti-kT.**
2. This arises **because anti-kT does not approximate the natural collinear structure of QCD.**



(*) the models were not trained so the values are meaningless.

Experiment

- Use Atlas software to gather a dataset of final state particles (their momenta) and a dataset of the associated reconstructed jets (using anti-kT) with engineered variables.

Targeting DAOD_JETM6:

1. Contain EMTopoJet, PFLow, and Truth jets. Comparing performance will offer insight.
2. Collecting info on ttbar process (the quark jet-rich set) and dijet (the gluon jet-rich one).
3. Implements some quality cuts*.

- Process the constituents into *json* files containing data (10^6 jets each) for JUNIPR.

- Train two JUNIPR models: one on quark-rich data (ttbar, giving P(Jet|Quark)) and one on gluon-rich data (dijet, giving P(Jet|Gluon)).

- For discrimination, train these two JUNIPR models further with a **discriminative objective****:

For **quark jet** maximise $\frac{\mathrm{P(Jet|Quark)}}{\mathrm{P(Jet|Gluon)}}$ and for **gluon jets** maximise $\frac{\mathrm{P(Jet|Gluon)}}{\mathrm{P(Jet|Quark)}}$

Discriminate based on the value of $\frac{\mathrm{P(Jet|Quark)}}{\mathrm{P(Jet|Gluon)}}$

This step is vital as ML is very bad at predictions on data out of its model (for a jet from dijet, P(Jet|Quark) will likely not be precise) and to improve the discrimination efficiency.

- Compare performance with the benchmarking models (BDT and MLP) trained on jet variables.

(*) remove jets with: pT < 20 GeV, <5 constituents, and if they are not from the primary vertex.
(**) it is in fact sigmoid [-log(Probability(Jet|Quark) + log(Probability(Jet|Gluon)] that is minimised for quarks, and similarly for gluons.

Conclusion

Objectives stated at the beginning:

<<<
- Improve the efficiency of the Q-G discrimination task.

- Restrain to fast processing and low-level inputs close to the events: idea of implementing this for complex online trigger.

- Explore machine learning solutions with a focus on performance and interpretability.

>>>

JUNIPR seems like a promising option to meet this challenge.
It can also easily generalise outside of the Q-G discrimination context.

References

[1] A. Andreassen, I. Feige, C. Frye, and M.D. Schwartz. JUNIPR: a Framework for Unsupervised Machine Learning in Particle Physics. Eur. Phys. J., C79, 2019.

[2] A. Andreassen, I. Feige, C. Frye, M.D. Schwartz. Binary JUNIPR: an interpretable probabilistic model for discrimination, Phys. Rev. Lett., 123, 2019.