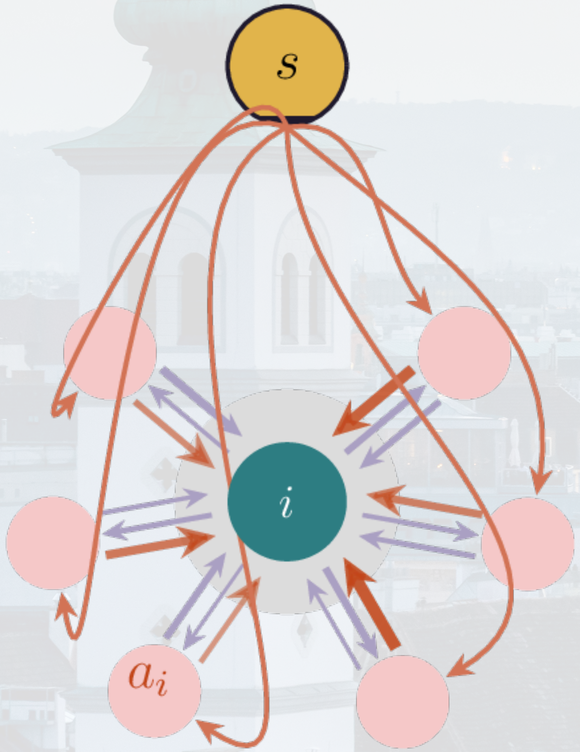
Where Does the Particle Transformer Look Inside a Jet? A Jacobian-Lens Interpretation



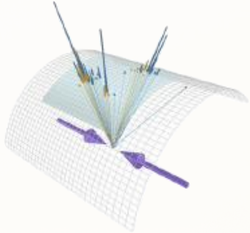
Sitian Qian (NU & FNAL)

With Huilin (SJTU & TDLI), Akshat (TTU) and Kevin (FNAL)

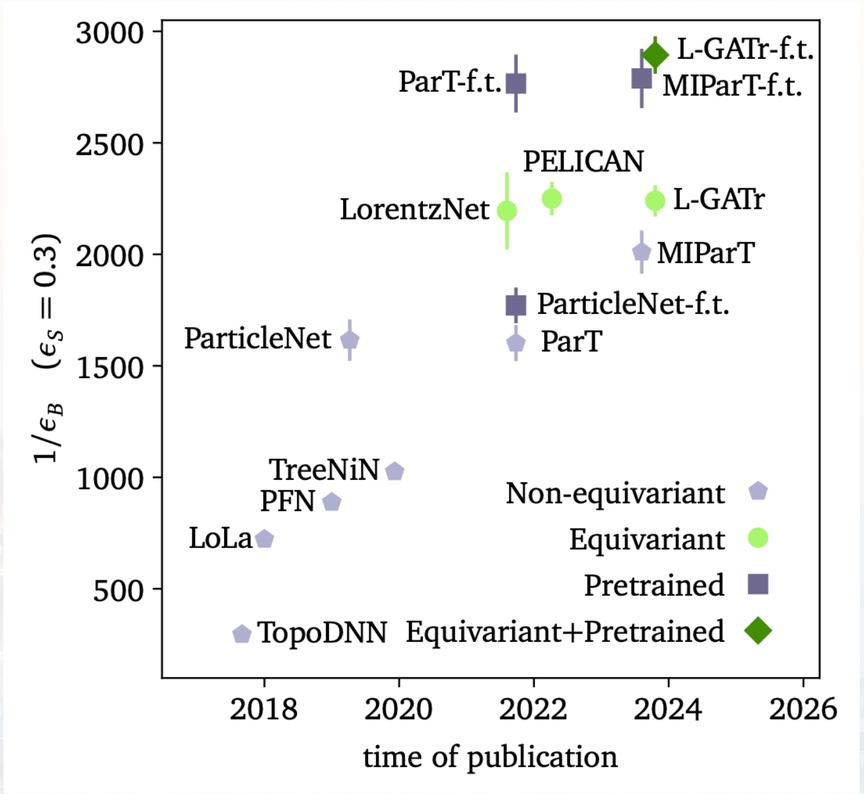
ML4Jets 2026, Vienna

2026-Sep-16th

Why reading a jet (tagger)?



- Constituent based ML tools have emerged in jet physics
  - As of today, transformer based methods have demonstrated great capabilities in various jet studies
  - But constituent based ML methods are not by definition IRC safe, therefore:
    - Where does the performance gain rest on ***physically***?
    - Whether these models are sensitive to ***generator details*** and thus generator dependent?
    - With less supervision imposed (such as pretraining with self-supervision), **which physics will the model pick up?**



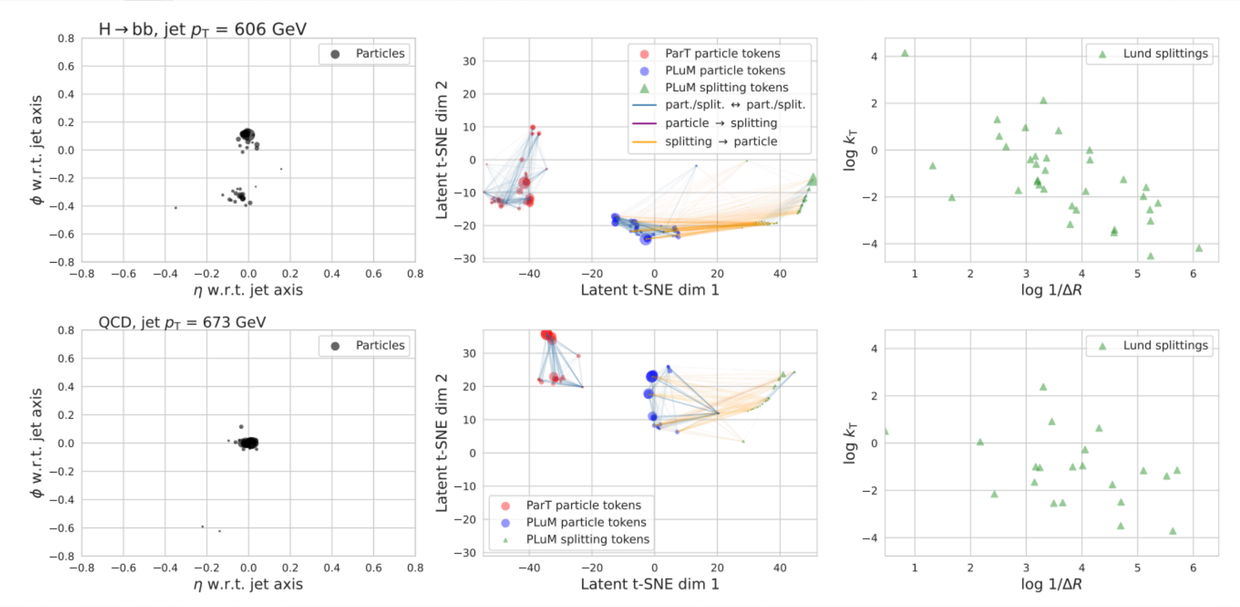
taken from 2411.00446

I don't have an answer for the questions today, rather offering an perspective for ParT

How to read a jet (tagger)?

- Interpretation methods exist, with great motivations from **physics** and ML point of views

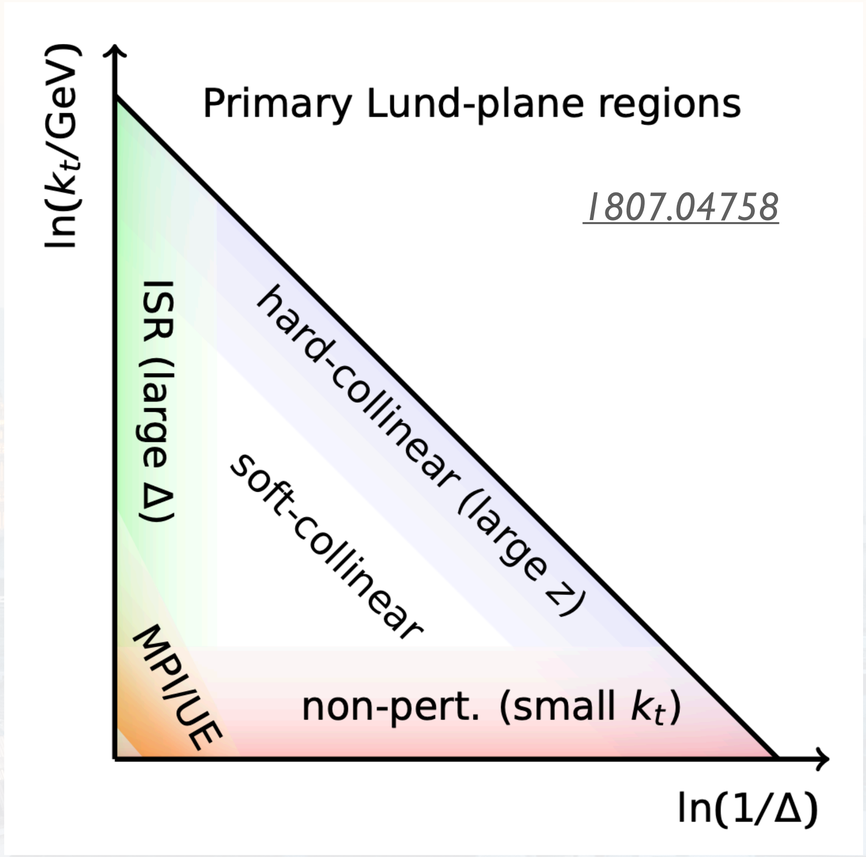
A crucial ingredient of ParT is the pair-wise feature inspired by Lund splittings [2605.26821](2605.26821)



Particle-Lund Multimodality in Jet Taggers

Loukas Gouskos*
Brown University, Providence, USA

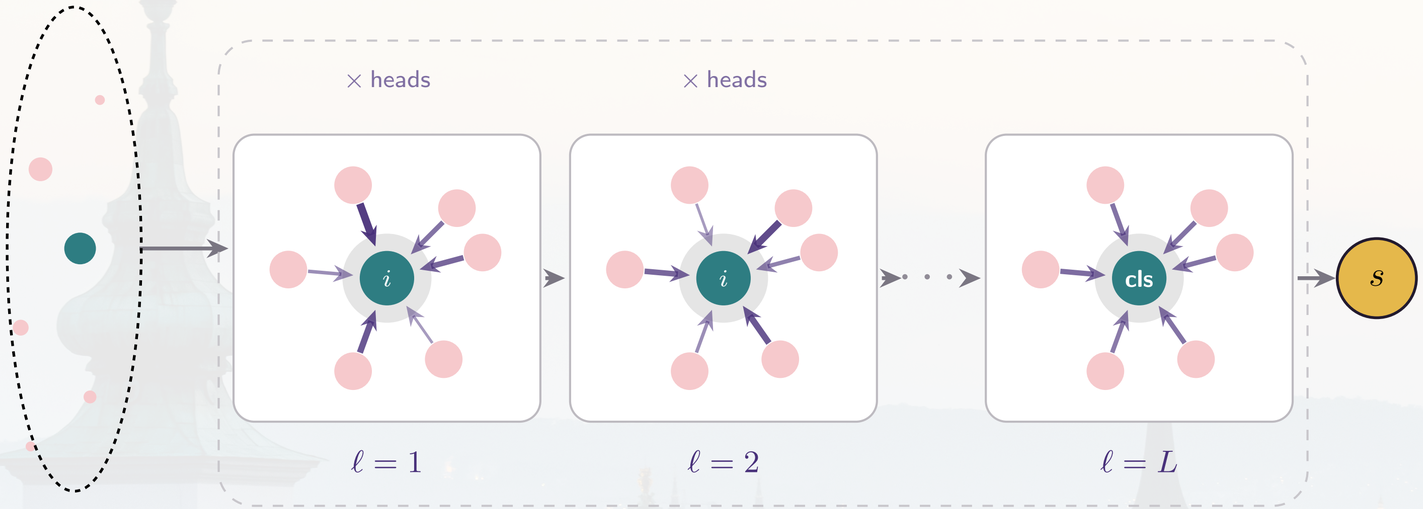
Benedikt Maier†
Imperial College of Science, Technology and Medicine, London, United Kingdom



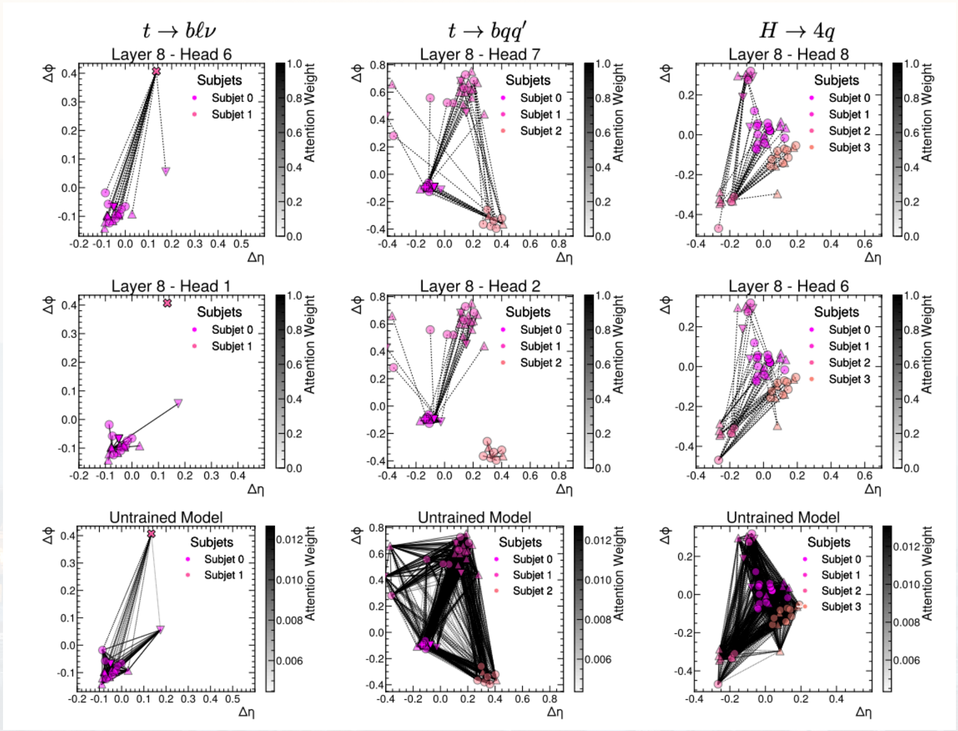
Therefore the Lund modality should help ParT's interpretation

How to read a jet (tagger)?

- Interpretation methods exist, with great motivations from physics and **ML** point of views



2412.03673



Interpreting Transformers for Jet Tagging

Aaron Wang
University of Illinois Chicago
Chicago, IL 60607
aaronw5@uic.edu

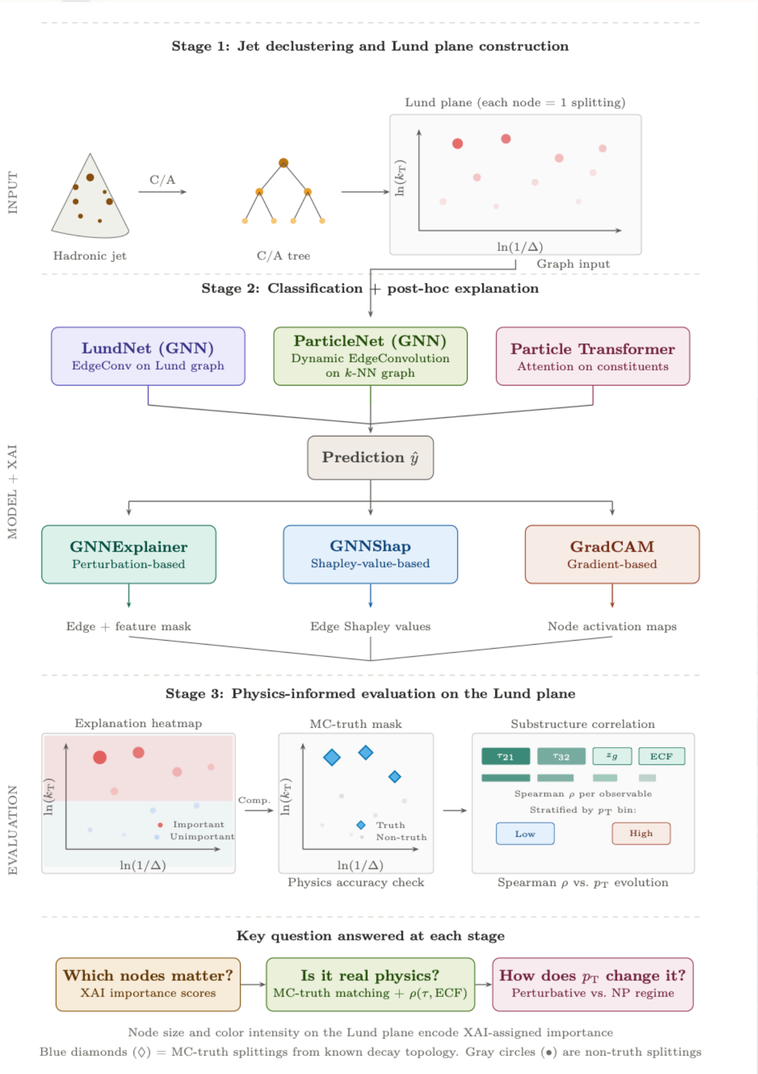
Abhijith Gandrakota **Jennifer Ngadiuba**
Fermi National Accelerator Laboratory
Batavia, IL 60510
{abhijith,ngadiuba}@fnal.gov

Vivekanand Sahu **Priyansh Bhatnagar** **Elham E Khoda** **Javier Duarte**
University of California San Diego
La Jolla, CA 92093
{vsahu,prbhatnagar,ekhoda,jduarte}@ucsd.edu

ParT is a transformer, therefore attention map reveals the underlying mechanism

How to read a jet (tagger)?

- Interpretation methods exist, with great motivations from **physics** and **ML** point of views

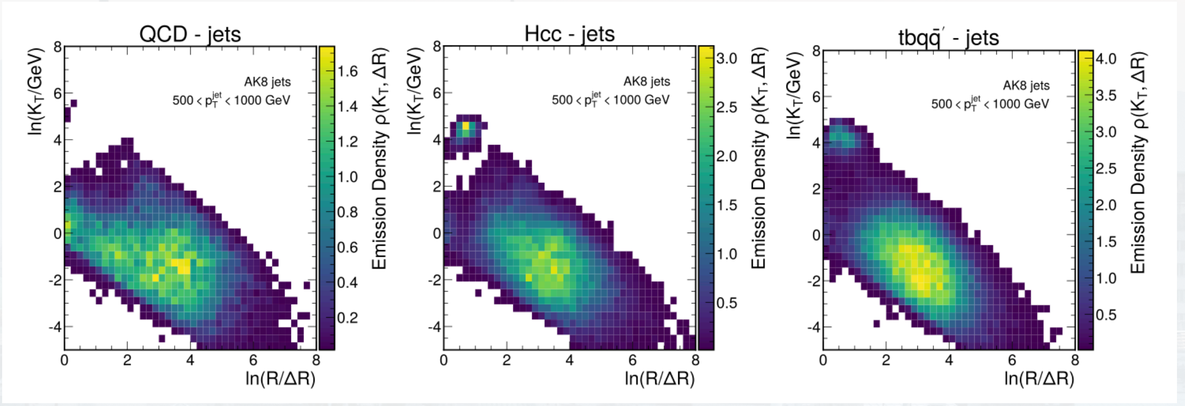


Explainable AI for Jet Tagging: A Comparative Study of GNNExplainer, GNNShap, and GradCAM for Jet Tagging in the Lund Jet Plane

Pahal D. Patel[1,*] and Sanmay Ganguly[1,†]

[1] *Department of Physics, Indian Institute of Technology, Kanpur Uttar-Pradesh-208016, India*

2604.25885



And even combining both to analyze the model with a view from the Lund Plane!

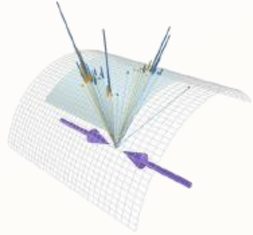
What's the Jacobian Lens here

- TL;DR:
  - A way of attributing the per input attribution from gradients (*a la the Jacobian rule*)

$$a_i = \left| \sum_f x_i^f \, \partial s / \partial x_i^f \right|$$

  - *Analogue to gradient based feature importance, though x could be a particle here*

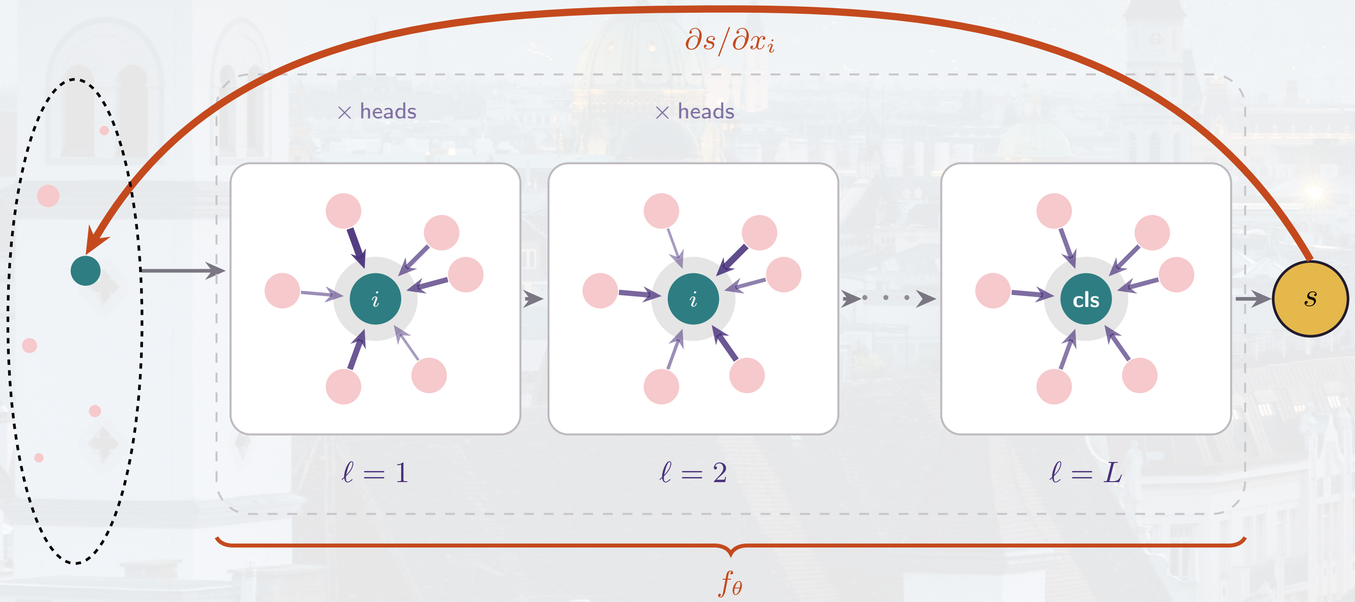
Recently revived by Anthropic for LLMs, inspiration from here, though simplified :)

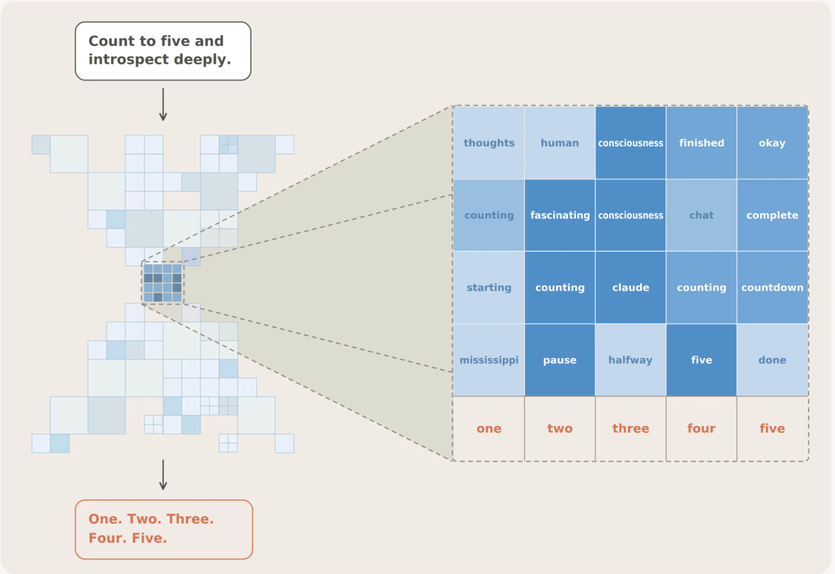


A Jacobian computed on a single prompt, however, conflates two kinds of structure: the model's general disposition to verbalize a given concept, and the particular use to which that concept is being put in the current context. We isolate the former component by averaging within and across contexts. For each layer ℓ, we compute

$$J_\ell = \mathbb{E}_{t,\, t' \geq t,\, \text{prompt}} \left[\frac{\partial h_{\text{final},t'}}{\partial h_{\ell,t}} \right],$$

where the expectation is taken over the source position t, all subsequent positions t' within the context, and a corpus of one thousand prompts sampled from a pretraining-like distribution. The result is a single $d_{\text{model}} \times d_{\text{model}}$ matrix per layer that maps from a source layer ℓ to the final layer L.

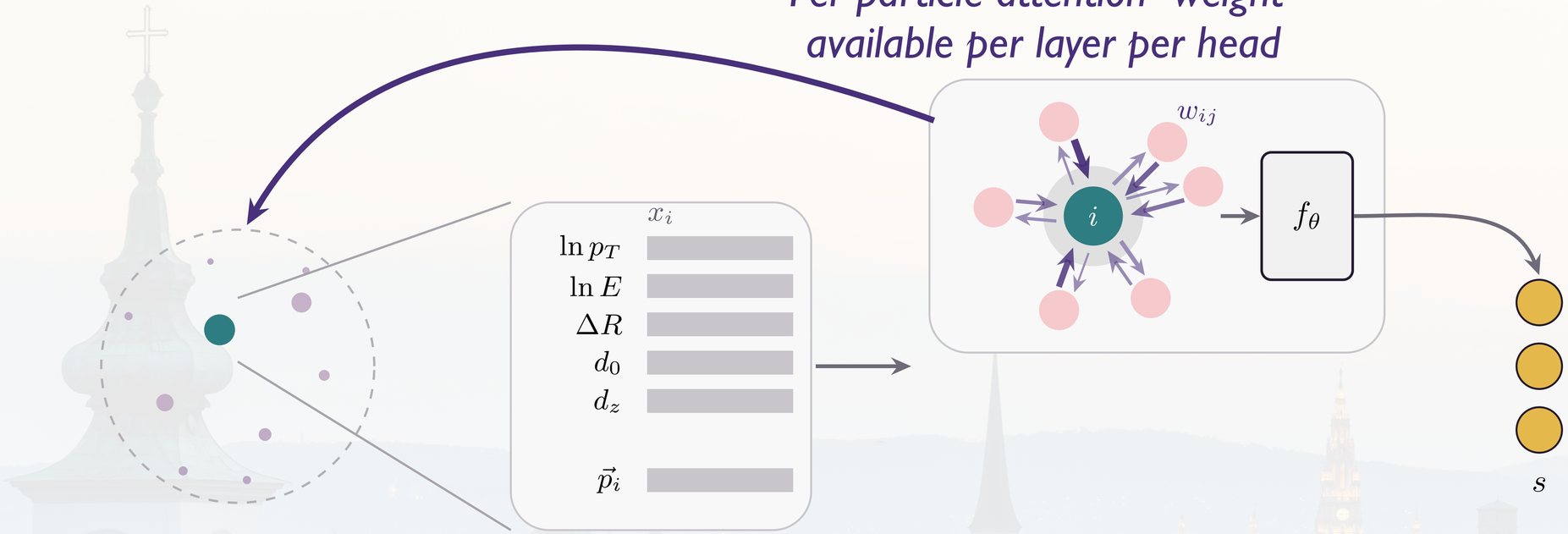




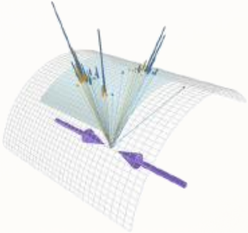
The J-space reveals internal thoughts that don't appear in the model's output.

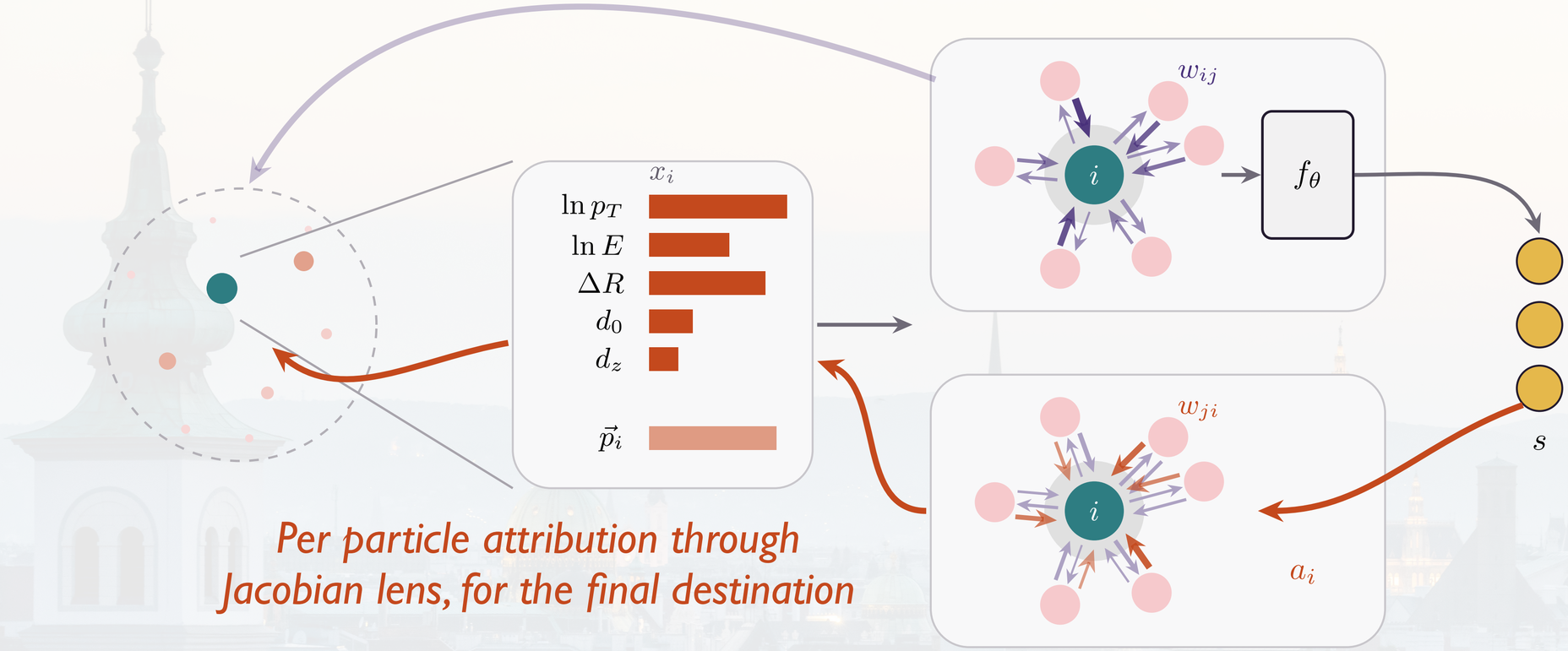
What's a Jacobian Lens for ParT

Per particle attention "weight" available per layer per head



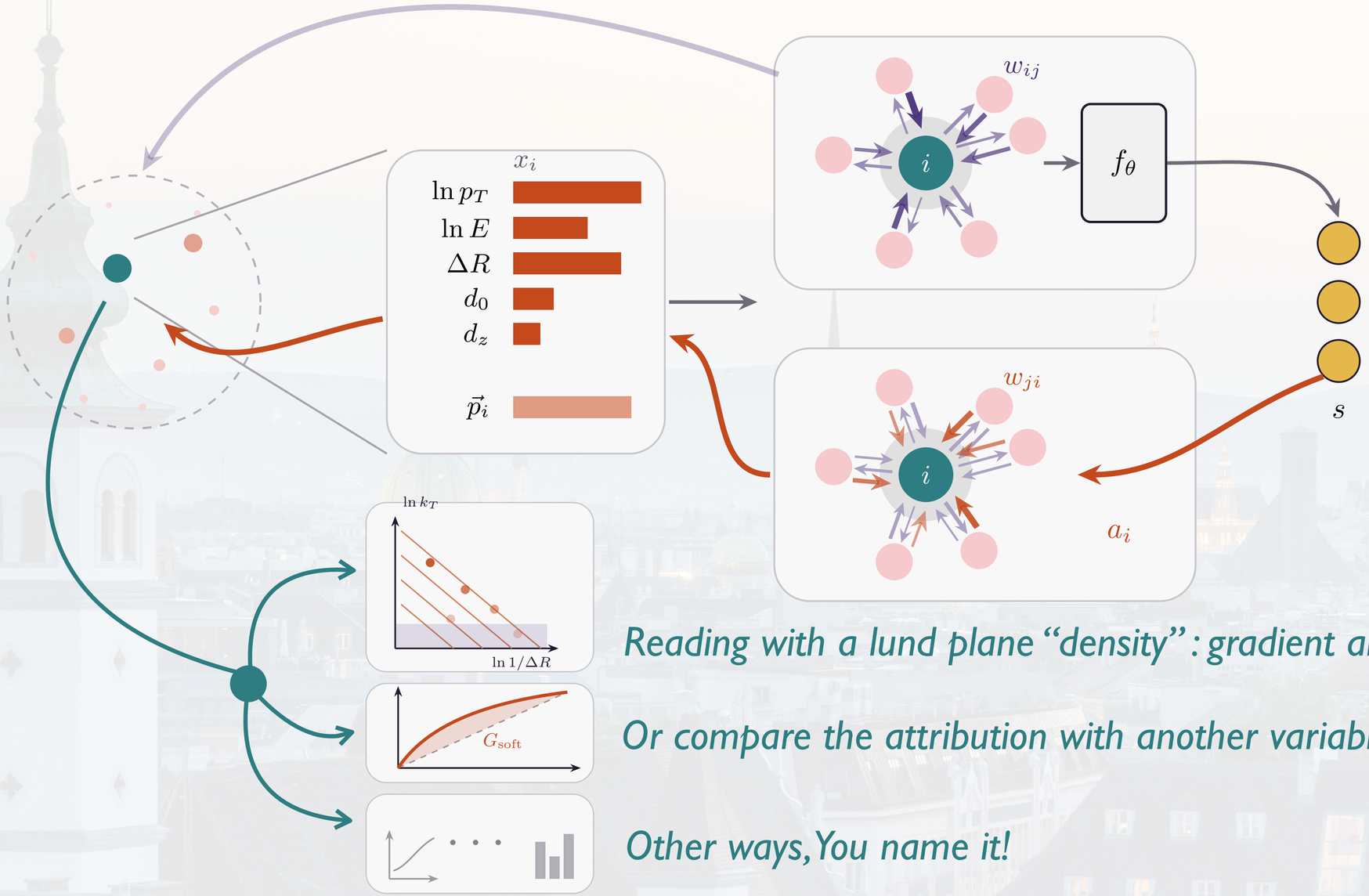
What's a Jacobian Lens for ParT





Per particle attribution through Jacobian lens, for the final destination

What's a Jacobian Lens for ParT

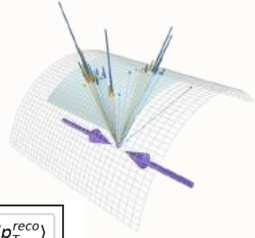


Reading with a lund plane "density" : gradient allows aggregation per split

Or compare the attribution with another variable with Gini Index

Other ways, You name it!

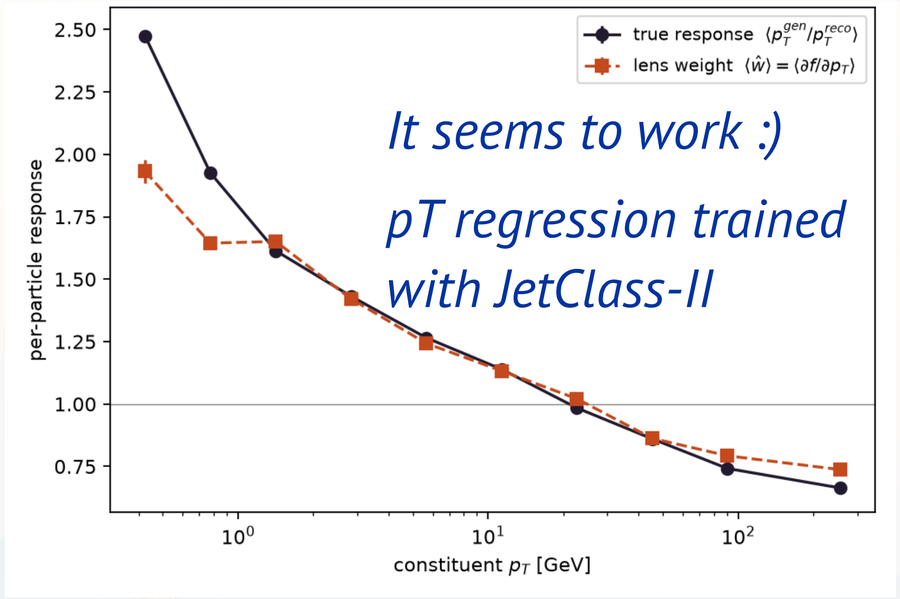
How does Jacobian Lens read a ParT?

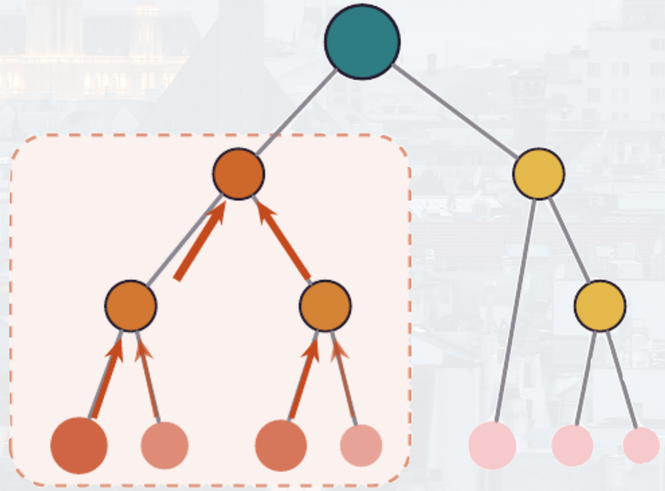


- More straightforward: it is a derivative
  - if we take gradient w.r.t. a jet energy regression output node, the attribution behaves like a per-particle correction weight?!

$$a_i = \left| \sum_f x_i^f \, \partial s / \partial x_i^f \right| \sim s := p_{\mathrm{T}}^{\mathrm{GEN}} / p_{\mathrm{T}}^{\mathrm{RECO}}$$

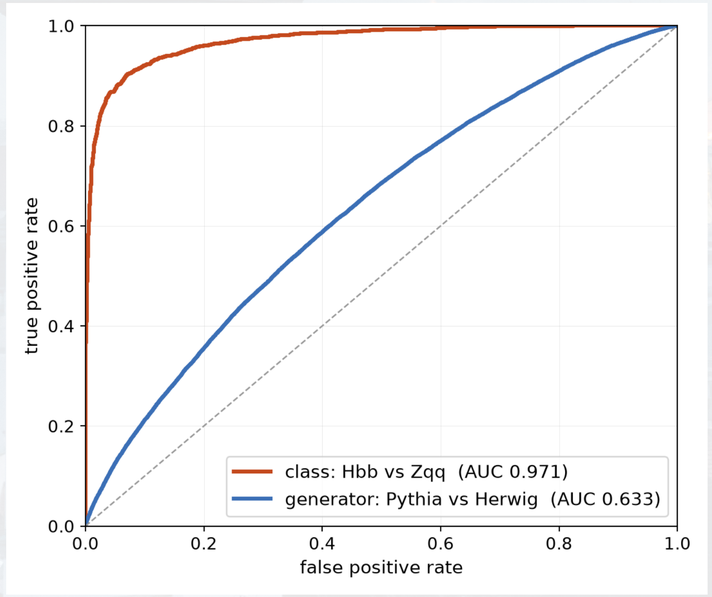
- Lund plane is for “split”, how to connect it with a NN?
  - C/A decluttering introduce a strict sequence
  - Hence the gradients can be aggregated per group from linearity!
    - i.e. lensing at each split :)





How does Jacobian Lens read a ParT?

- Same ParT backbone:
  - Out-of box for MC class physics discriminate (Hbb vs Zqq)
  - Fine tuned for MC domain classification (Pythia vs Herwig)

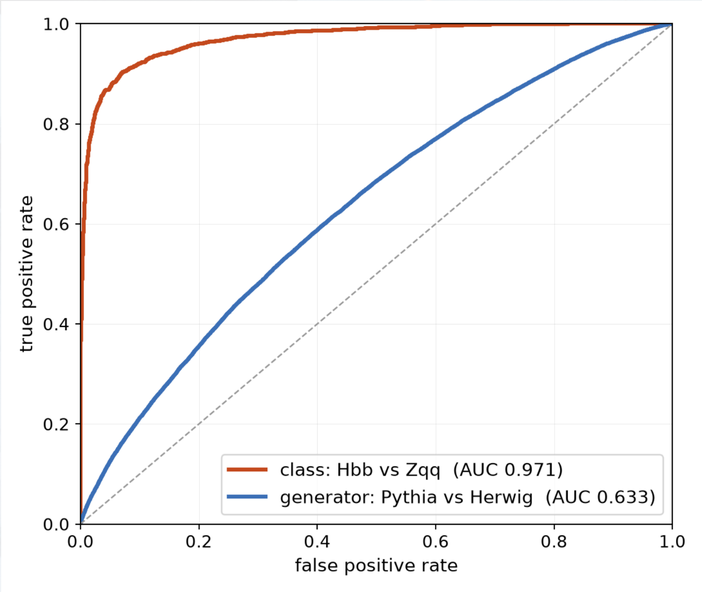


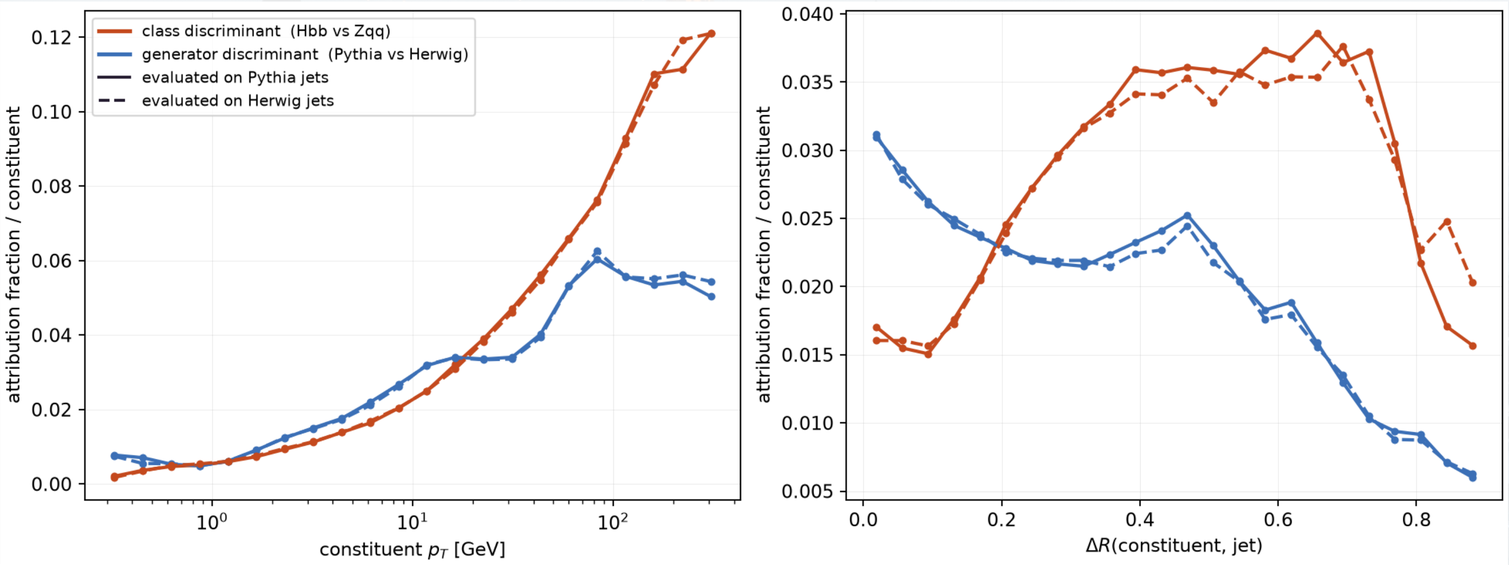
On the generator side, the discrimination power is modest…

Still, should have some domain classification / shifting power… :)

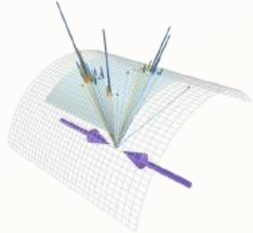
How does Jacobian Lens read a ParT?

- Same ParT backbone:
  - Out-of box for MC class physics discriminate (Hbb vs Zqq)
  - Fine tuned for MC domain classification (Pythia vs Herwig)
    - Generator classification relies on soft regime, while physics classification tracks hard constituents
    - *Two prong structure reflected into the dR profiling of the Jacobian lens*

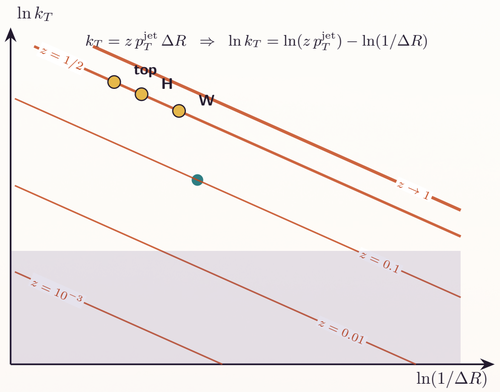




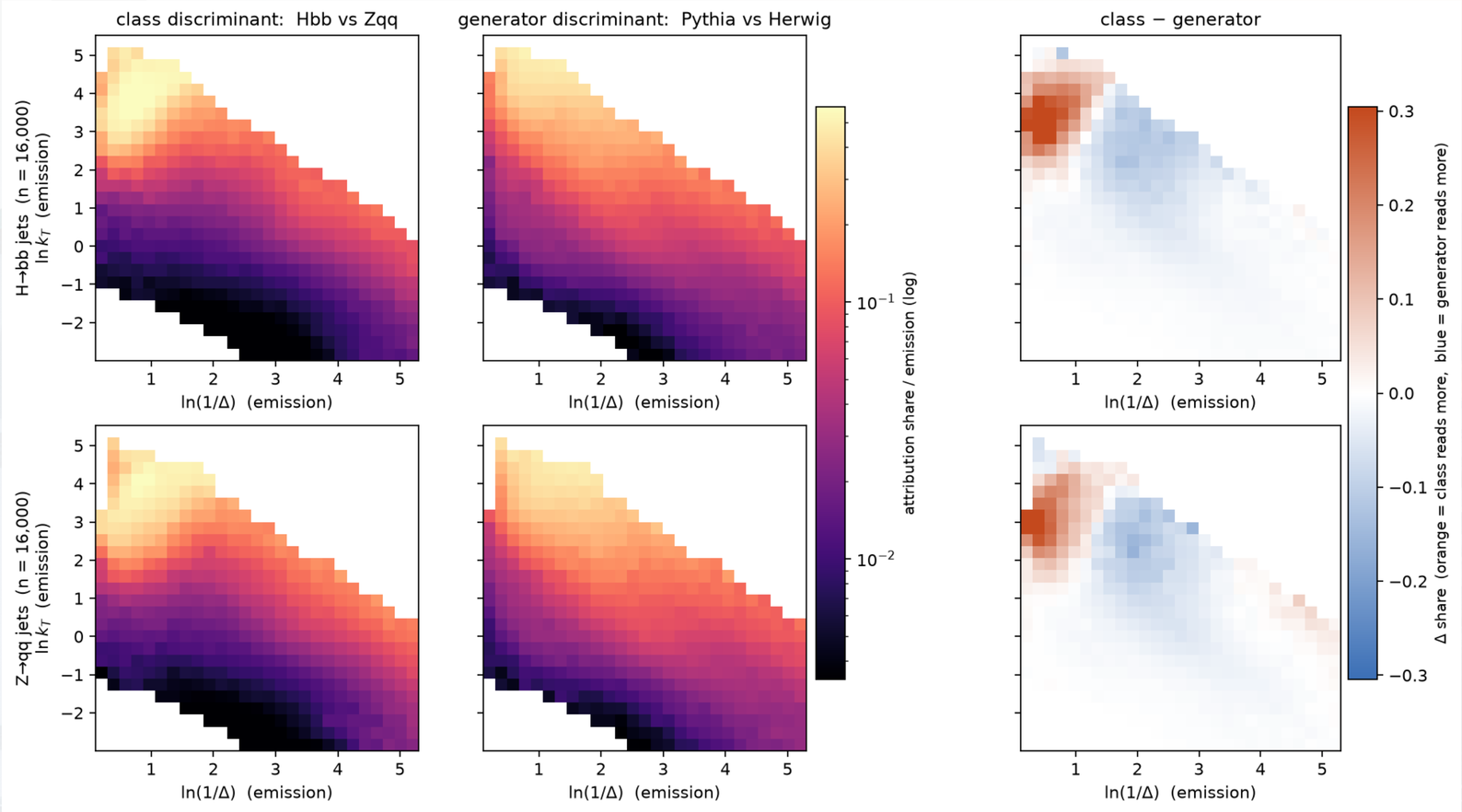
How does Jacobian Lens read a ParT?

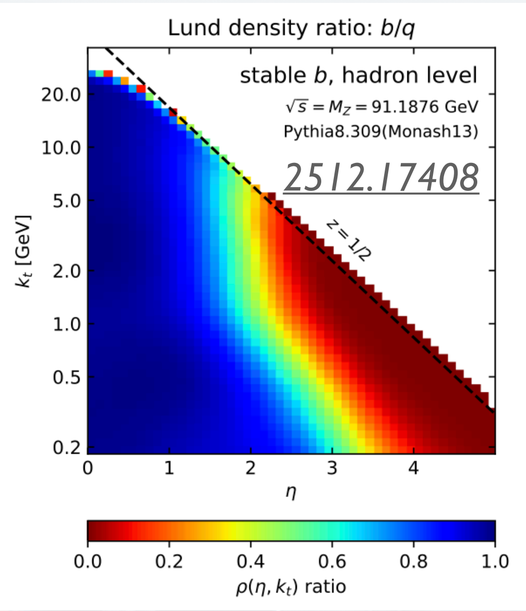


- Same ParT backbone:
  - Out-of box for MC class physics discriminate (Hbb vs Zqq,)
  - Fine tuned for MC domain classification (Pythia vs Herwig)

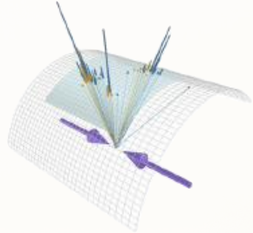


Different generator on the same **H***bb* vs **Z***qq* physics processes classification

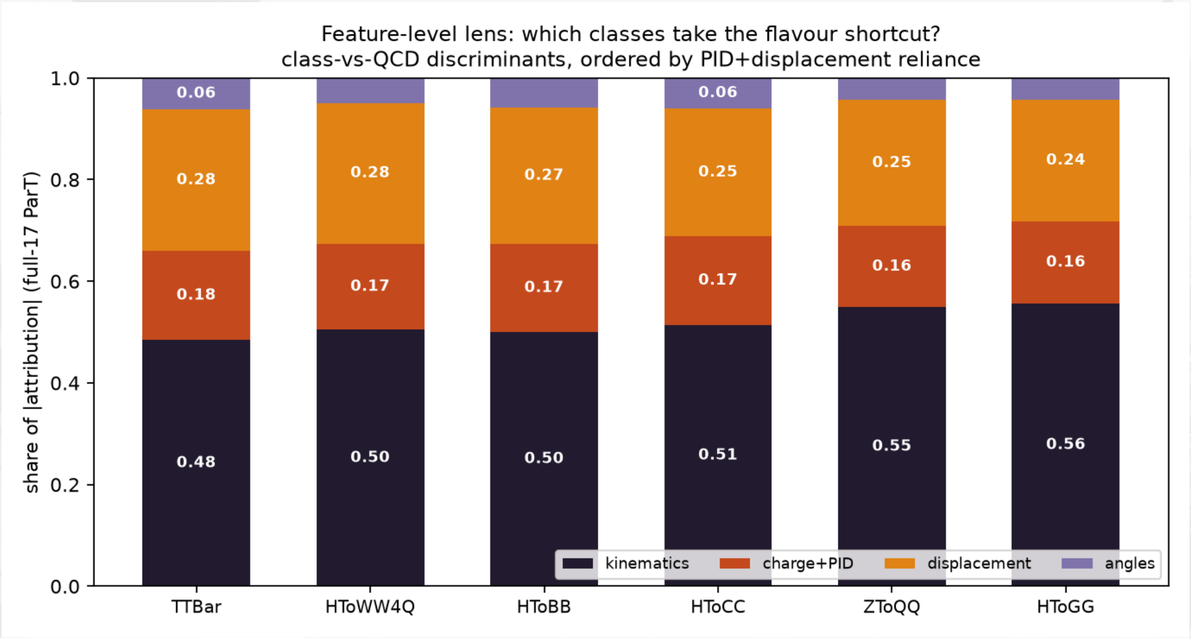




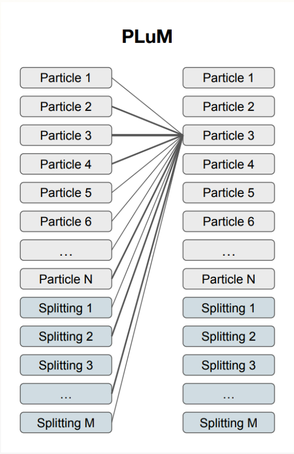
How does Jacobian Lens read a ParT?

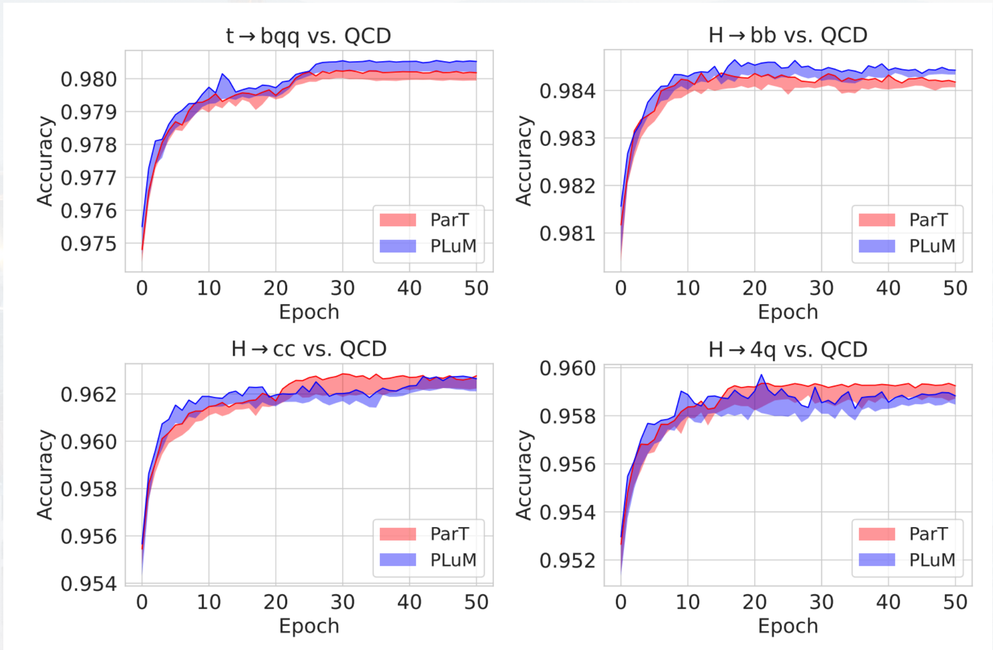


- As promised, Jacobian lens can trace down to feature set level
  - We can get the Jacobian lens per feature group
    - Behaves as expected :)

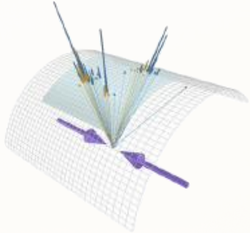


- Hence the quest from PLuM:
  - Explicit adding Lund splittings to enhance the performance
    - Works for heavy flavor
    - Tiers with ParT for lighter

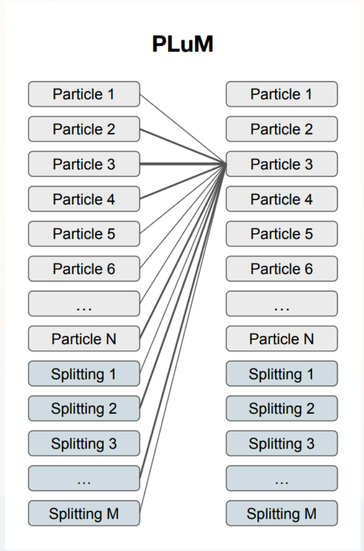


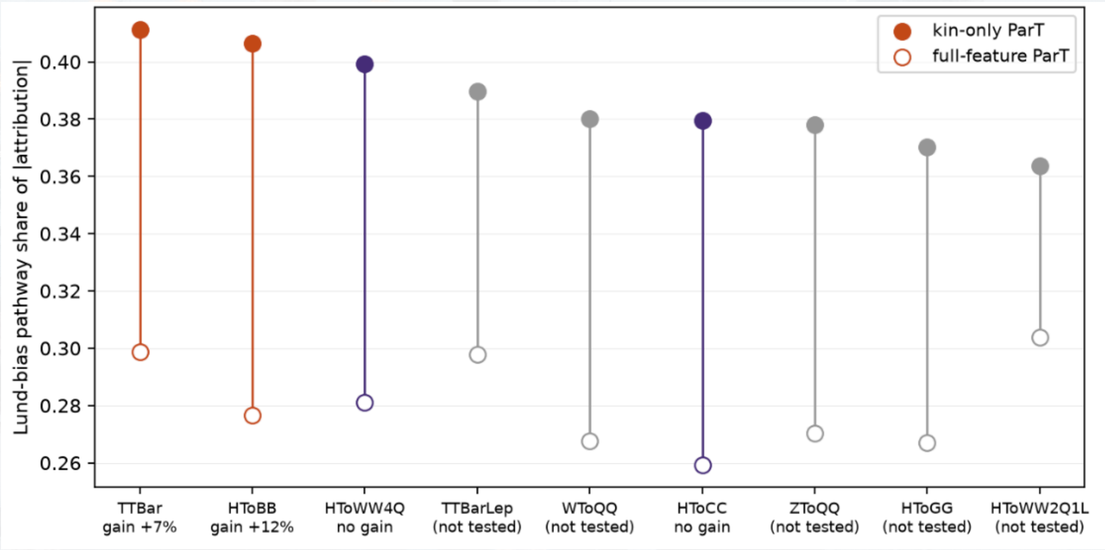


How does Jacobian Lens read a ParT?

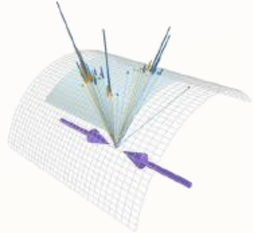


- A Jacobian Lens answer to PLuM:
  - Taking the lens to the Pair-wise feature bias route, then project on the Lund plane:
    - The gain lives align with classes where attention more concentrated
    - With more input features the lund bias share will **dilute**
      - PLuM could bring it back **by enforcing explicit Lund splitting**
        - But if original Lund attention share is *little*, then the gain is *little* too…
    - Therefore, my humble guess for the rest:
      - Gain won't be much…



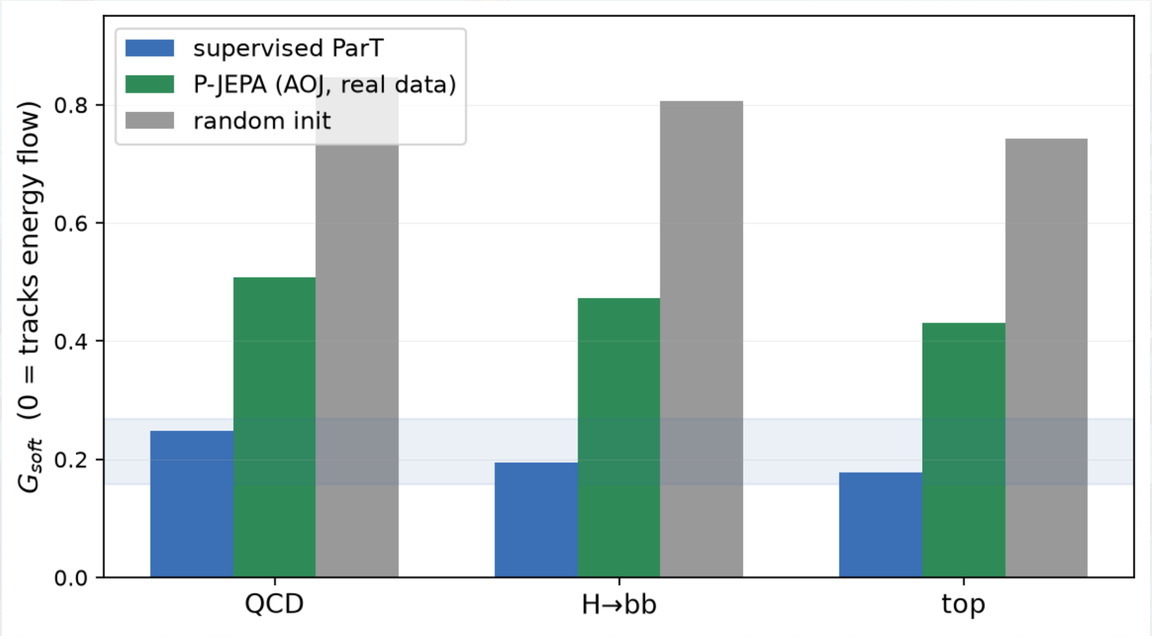


How does Jacobian Lens read a ParT?



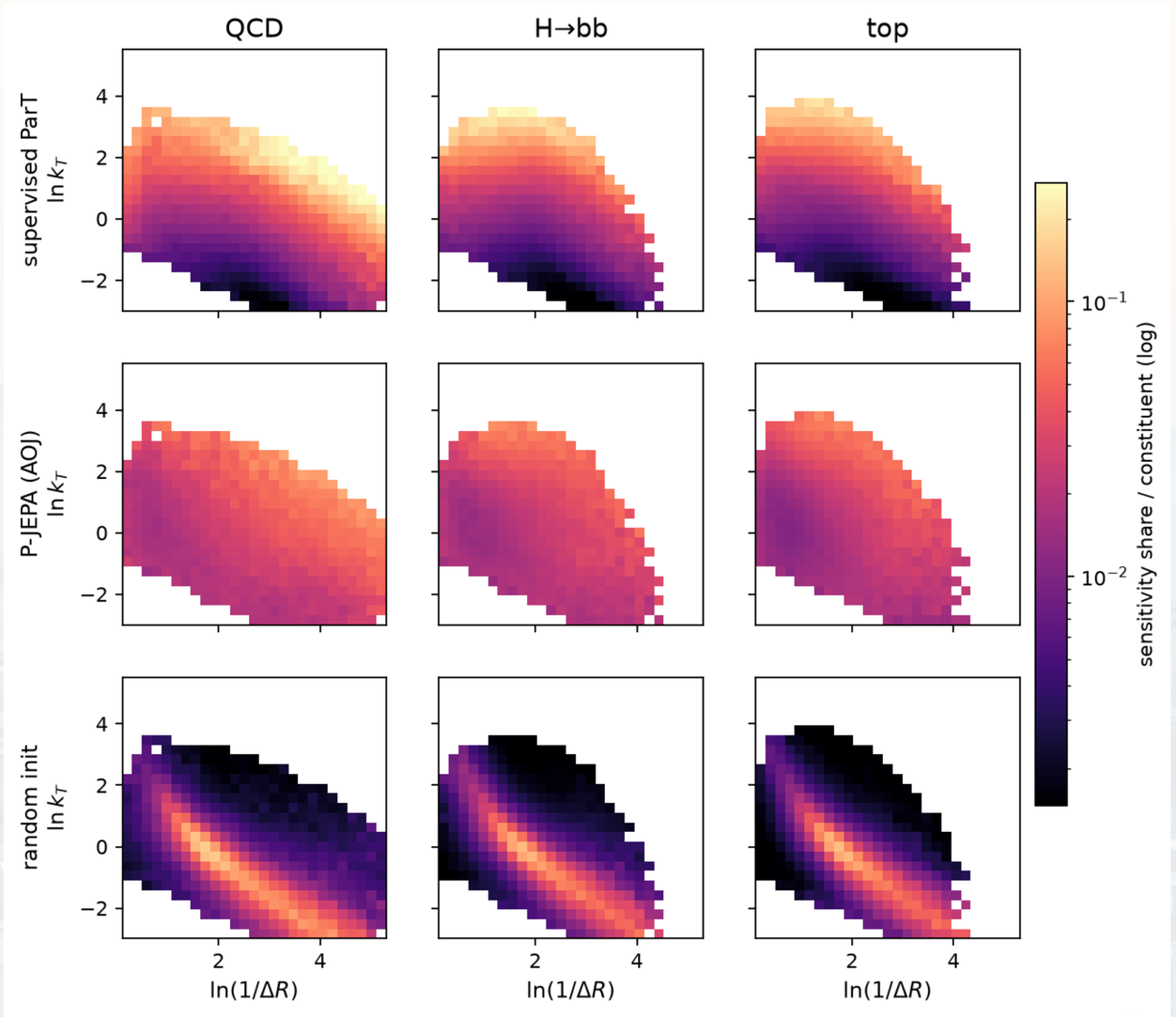
- How about another corpus from real data with self-supervised learning (SSL)?
  - Attempted SSL methods with Aspen Open Jet (AOJ) dataset —> jets collected from CMS open data
  - Firstly trained P-JEPA on AOJ, then applied to JetClass to test the reading, w.r.t. supervised ParT models
    - Robustness against domain shifts can be tested —> Key capabilities for transfer learning
    - AOJ-P-JEPA -> Supervised ParT
    - Different from Random weight
    - **Advantage of real data SSL?**

SSL lens-ed through a random pulling in embedding direction and get the gradient



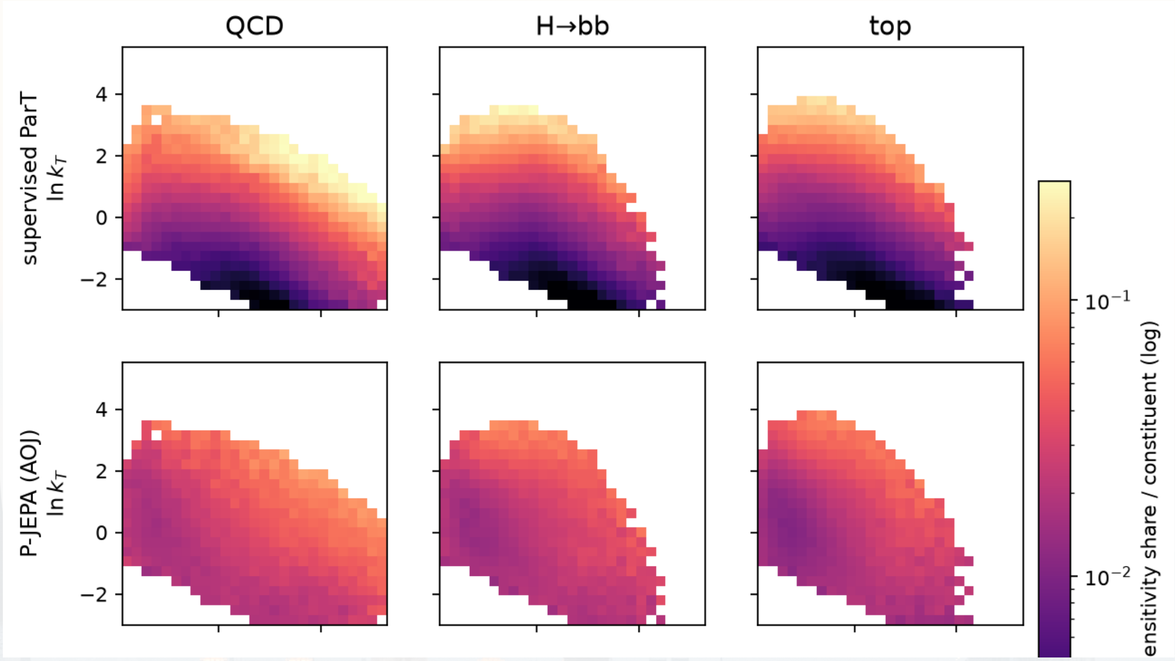
How does Jacobian Lens read a ParT?

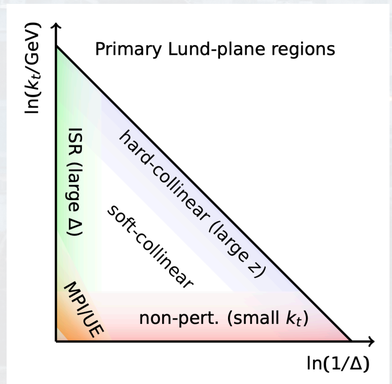
- How about another corpus from real data with self-supervised learning (SSL)?
  - Firstly trained P-JEPA on AOJ, then applied to JetClass(-I) to test the reading, w.r.t. supervised ParT models
    - OJ-P-JEPA -> Supervised ParT
    - Different from Random weight
    - **Advantage of real data SSL?**
  - We see certain structure towards hard splittings in both data and supervised training, on the Lund plane Jacobian lens

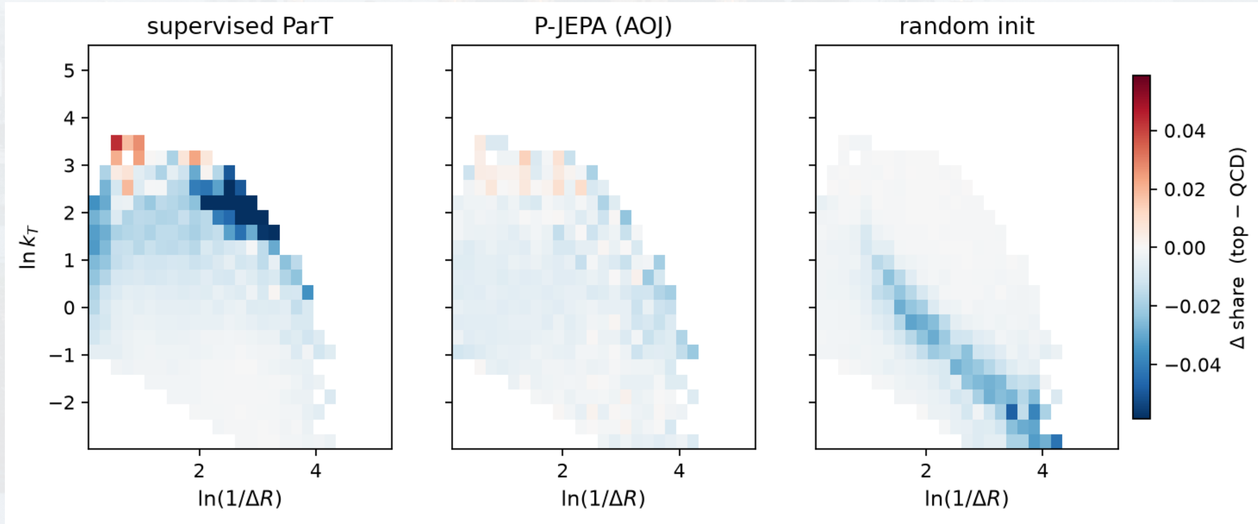


How does Jacobian Lens read a ParT?

- What is the difference?
  - Intuition: AOJ is dominated by QCD, so attention (hence the reading) will be QCD energy flow structure focused!
  - Below is the **difference between TTbar and QCD**
    - Supervised ParT demonstrates the structure
    - AOJ like random…

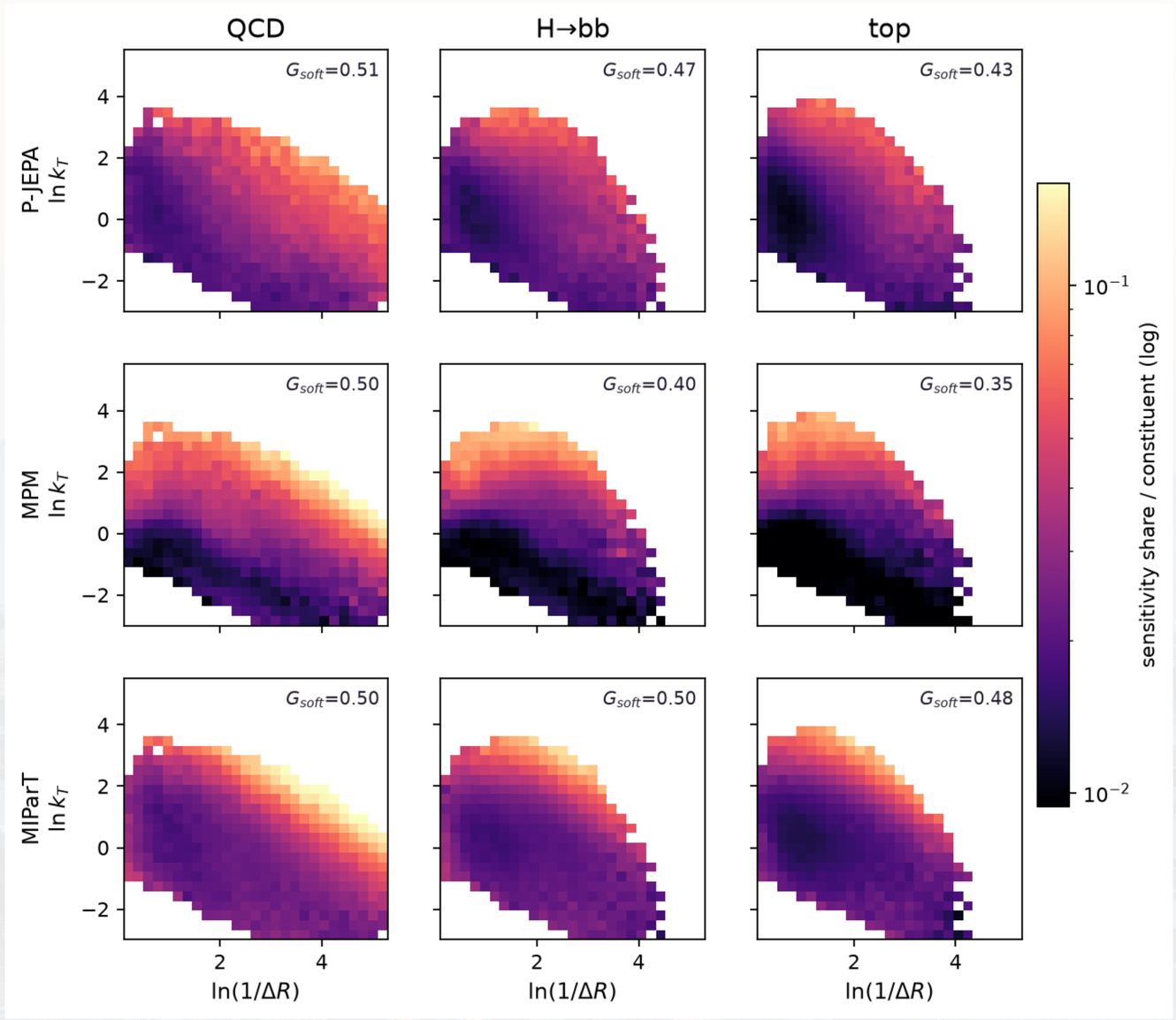




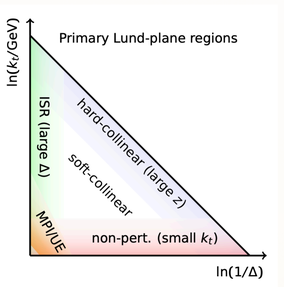


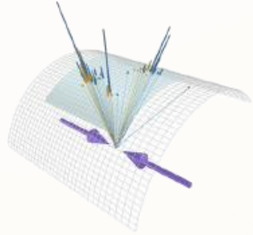
How does Jacobian Lens read a ParT?

- Do different **SSL paradigm** / *Backbone* differ?
  - **Exercised Masked Particle Modeling (MPM) vs P-JEPA**
  - *Exercised MIParT vs ParT*

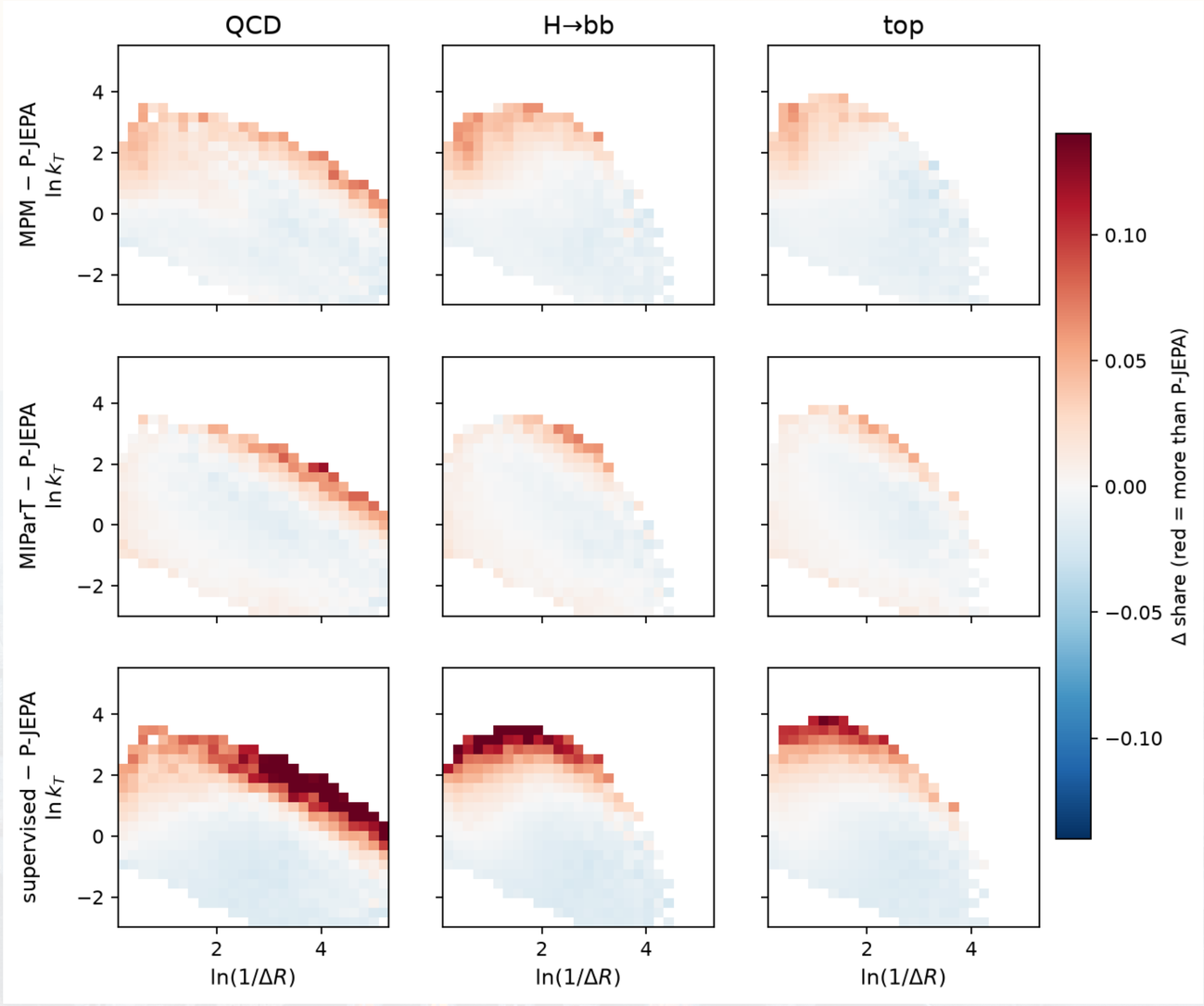


How does Jacobian Lens read a ParT?

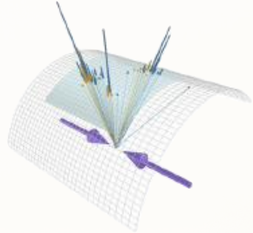




- Do different **SSL paradigm** / *Backbone* differ?
  - **Exercised Masked Particle Modeling (MPM) vs P-JEPA**
  - *Exercised MIParT vs ParT*
- Convergence is real: as long as on AOJ different training converges to similar patterns
  - But both variants demonstrated more physics pattern?
  - Again supervised has more physics!



Summary and Outlook



- Got inspired by the LLM Jacobian lens, and try the same for jets :)
  - Simple math setup: gradient from partial derivatives through the Jacobian rule
  - Not replacing attention map/shapley/… but offering a different angle
- Per particle / input / group enables mapping into physics spaces
- Paradigm can be generalized
  - For SSL studies: Offering another angle for the jet representation learnings
  - Jacobian lens is not limited to ParT or even Transformers /…, but anything auto-differentiable!
- Results are to be matured with codebase to be public
- A real intra layer LLM-style “Jacobian-lens” tracks the embedding shift?

Thank you!

PS: if you want a quick C/A decluttering or jet clustering on GPU, check Chirayu’s talk tomorrow :)

More generators in one slide

- Apart from Sherpa, all with hard process modeling from MadGraph
  - Sherpa's hard process is done with Comix -> aligned at LHE level with MG
  - 7-class generator domain classification fine-tuned

pair	AUC	G_soft generator	corr with class map (Hbb vs Zqq / Hbb vs QCD)
Pythia vs Herwig	0.63	0.4	0.60 / 0.67
Pythia vs Herwig 7.3	0.72	0.37	0.65 / 0.74
Pythia vs Sherpa	0.78	0.36	0.76 / 0.80
Pythia vs Dire	0.63	0.51	0.51 / 0.56
Pythia vs Vincia	0.61	0.52	0.48 / 0.53
Dire vs Vincia	0.56	0.39	0.78 / 0.83
Herwig vs Herwig 7.3	0.67	0.3	0.76 / 0.84