Wednesday, March 10, 2021 10:09 AM

Lecture 15

Last time: top tagging
- HLF cut based
- HLF DNN
- DNN on 4-vectors of constituents
- CNN on jet images

current SOTA (ca 2019): top tagging community comparison paper (Link to arXiv or website)

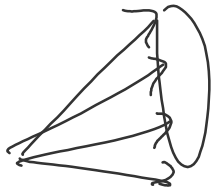
→ "winner" was top tagger based on graph neural networks "point clouds" ← self-driving cars lidar

2nd place was a CNN based on ResNet architecture

Wednesday, March 10, 2021 10:43 AM

Brief description of graph NNs

What is something permutation inv't



jet = {const$_1$, const$_2$, ...}
but order doesn't matter!

represent each data instance as a graph

each node is a constituent

general idea of GNN: learn set of connections btw nodes

map from graph → output

→ GNN(graphs) = output.

specific implementation for top tagging: for each node, consider k nearest neighbors in (η, ϕ)

train to learn "edge conv." for h ← shared weights across all nodes!

$\vec{x}_i \in \mathbb{R}^F$ ($F = 3$ initially (p_T, η, ϕ))

$\vec{x}_i' \in \mathbb{R}^{F'}$

$$\vec{x}_i' = \sum_{j=1}^{k} \vec{h}(\vec{x}_i, \vec{x}_j)$$ → sum!

↓ permutation inv't

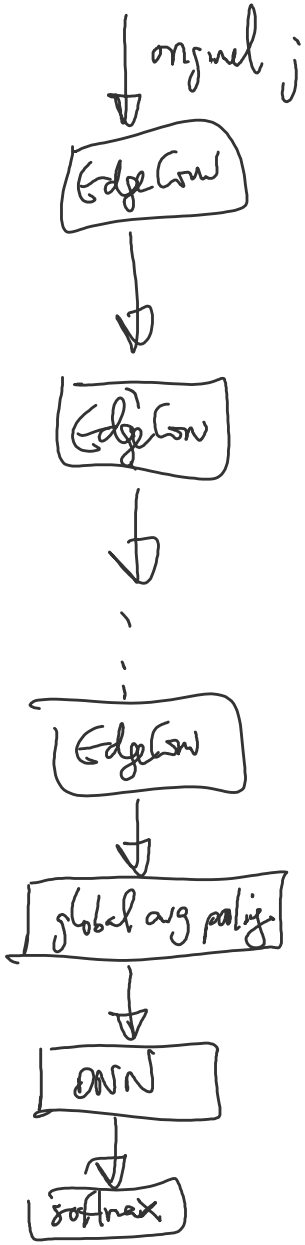
Wednesday, March 10, 2021 10:52 AM

EdgeConv produces a new graph!

↓

stackable!

"ParticleNet" ← 1902.08570 Qu & Gouskos

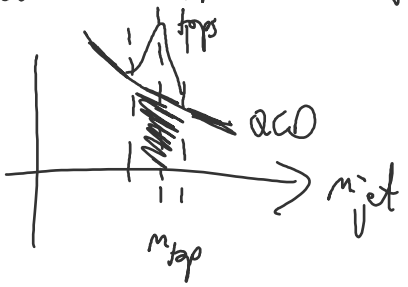


Wednesday, March 10, 2021 11:00 AM

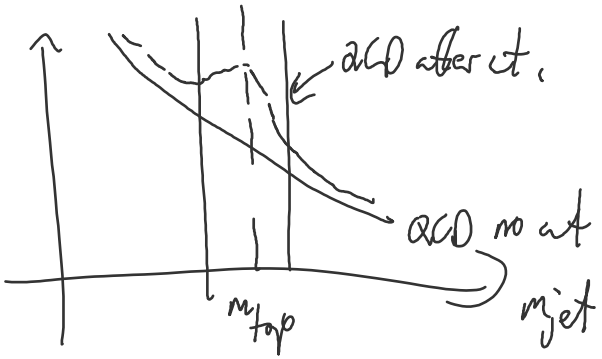
Beyond Classification — Decorrelation

In practical applications of jet taggers, raw performance not the only consideration.

- Many applications, m_{jet} reserved for sideband interpolation → background estimation



- If tagger "learns" m_{jet} → cut on classifier → sculpt a bump in background m_{jet} distribution

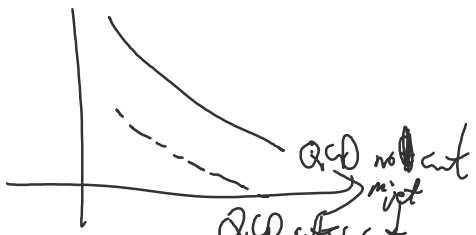


↓

cannot combine tagger & sideband method in m_{jet}!

Wednesday, March 10, 2021 11:09 AM

Want: classifier independent (statistically) of mjet in background.



"domain adaptation"
ML training data ⇔ application data
want to carry over performance from training data → other domain.

"AI fairness/bias"
many real world applications to hiring decisions, admissions, bail bonds

Commonly used metric (in HEP): Jensen-Shannon Divergence (JSD) btw mjet distr'n of bg before & after cut.

measure of similarity btw two distr'ns.
can be computed from histograms.

Wednesday, March 10, 2021 11:21 AM

$$JSD(P,Q) = \frac{1}{2}\left(KL(P\|M) + KL(Q\|M)\right) \qquad M = \frac{P+Q}{2} \text{ avg distn.}$$

$P(x), Q(x)$ prob. distns

$$KL(P\|M) = \text{"KL divergence"} = \int dx\, P(x) \log \frac{P(x)}{M(x)}$$

not symmetric in P & M

M is called "reference distn"

$KL(P\|M) \geq 0$ w/ $=0$ iff $P=M$.

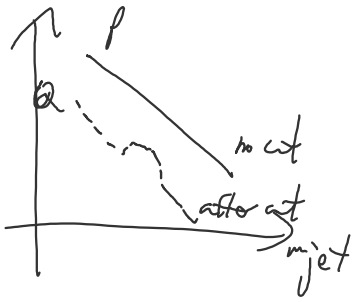
(easy to prove!)

Symmetrized KL divergence

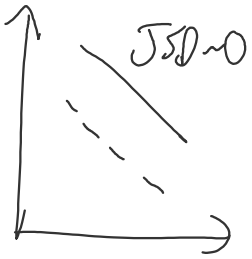
$0 \leq JSD \leq 1$ (base 2 logarithm)

iff $P=Q$

iff $P(x)$ & $Q(x)$ have no overlap.

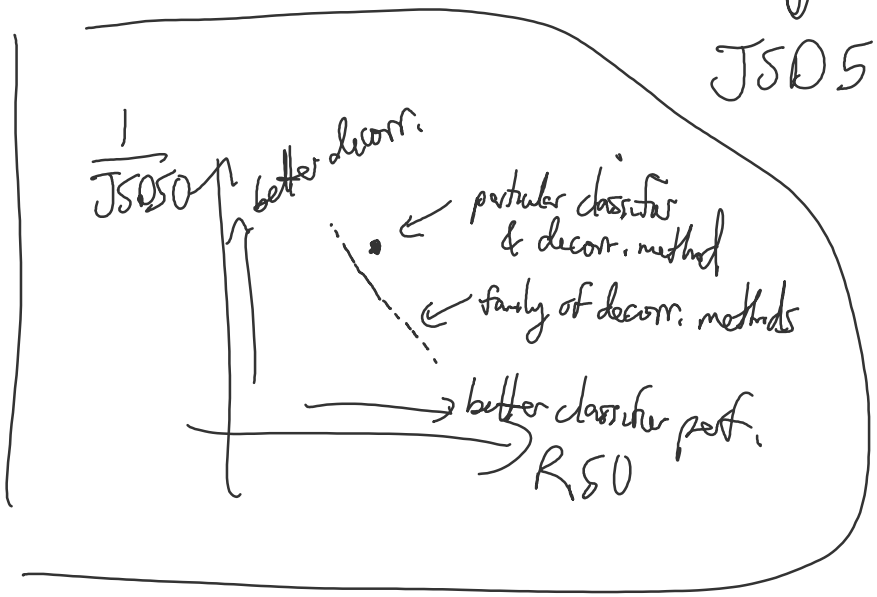


$JSD \sim 1$



Wednesday, March 10, 2021 11:27 AM

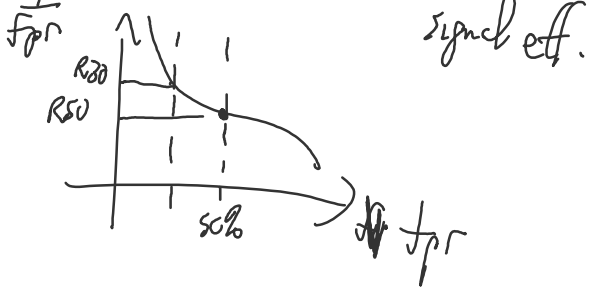
Need to choose a cut to define JSD — common choices are cut at 50% signal eff
"working point"
30% " "
⋮



fpr or background efficiency

$R = \frac{1}{fpr}$ rejection factor

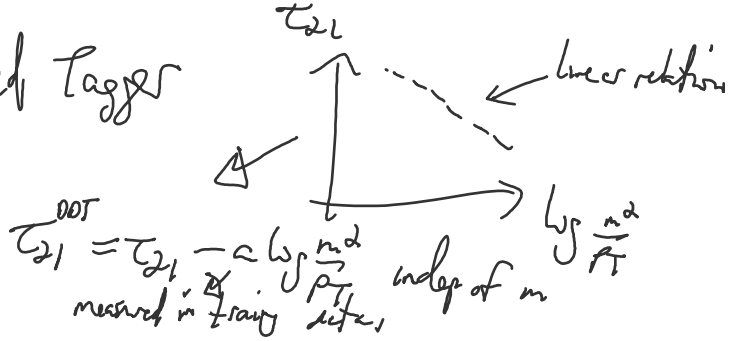
R50 rejection @ 50% signal eff.



Wednesday, March 10, 2021 11:27 AM

How can we achieve decorrelation?

1. Modifying training data to remove correl'ns
   i. How about removing m_{jet} from list of features in training data?
      in general this doesn't work!
      - other features can be highly correlated w/ m_{jet}
      - for deep NNs on low-level info like const. 4-vecs or jet images
        m_{jet} is not explicitly provided!
   ii. Hand engineer features that are indep. of m_{jet}.
      → example DDT — Designed Decorrelated Tagger
      → doesn't generalize to ML w/ many features.



Wednesday, March 10, 2021 11:41 AM

— "Planing" — reweight s_{ij} & b_{ij} mjet distr'ns to look like each other.

i.e. train classifier w/ weighted events $w(x,m) \sim \frac{1}{n_j}$ ← # events in data mass bin j

on features $\vec{x}$ ↓ m = mass bin j

— often works well in practice

— but not guaranteed

removes "leading order" mass information ⟶ guarantees marginal mass distr'ns are same

— doesn't generalize well to multiple features

but could still have nontrivial correlations btw X & m

$$\left\{ \begin{array}{l} p_{sig}(x,m) \\ p_{bg}(x,m) \end{array} \right. \longrightarrow \begin{array}{l} p_{sig}(m) \\ \| \\ p_{bg}(m) \end{array} \text{ after planing.}$$

$f_{classifier}(x)$ indep of m