Weak supervision for quark/gluon tagging in CMS Open Data

Matthew Dolan,
John Gargalionis,
Ayodele Ore

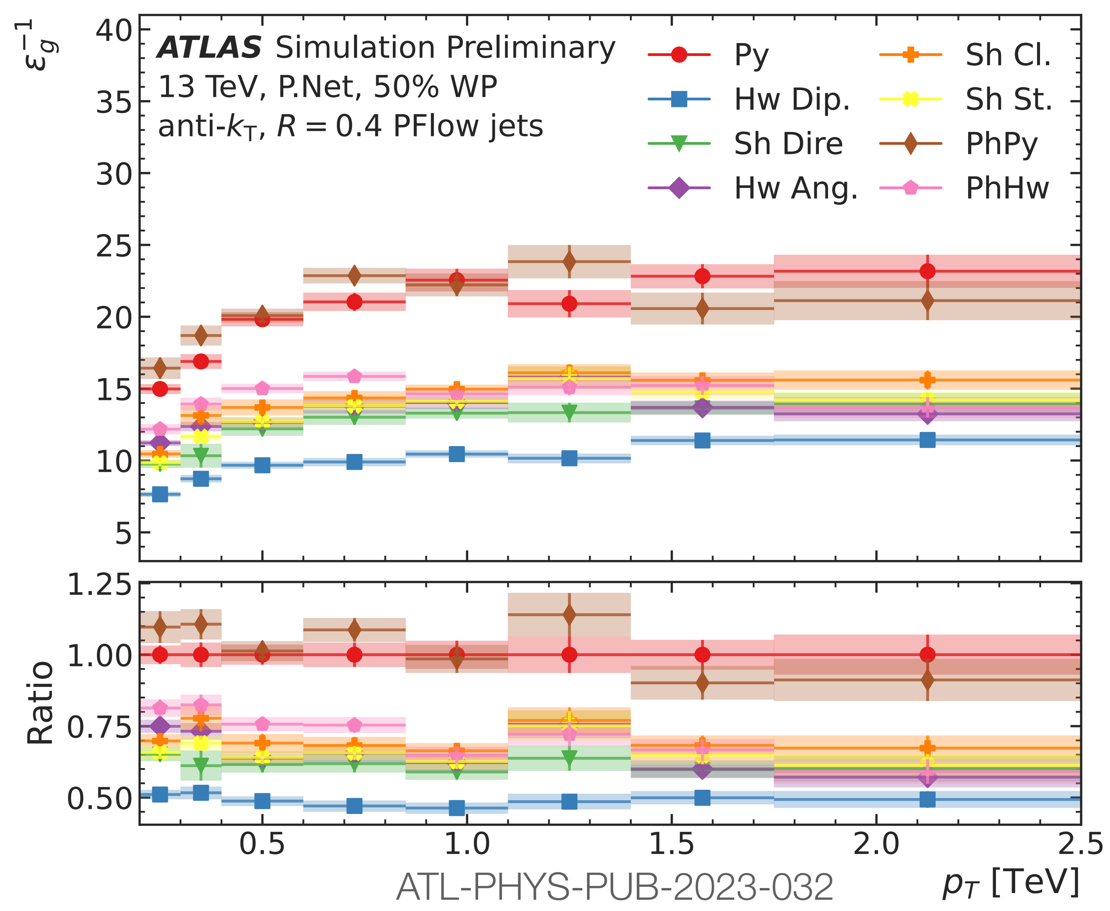
Motivation

When produced at high energy, quarks and gluons both lead to ***jets***, making them hard to distinguish at LHC.

Deep neural networks are powerful jet classifiers, but they are sensitive to details of simulation that suffer large theoretical uncertainty.

Weakly-supervised classifiers may avoid this issue by training on real data using unlabelled mixtures [1].

$$p_{M_1}(x) = f_1\, p_Q(x) + (1 - f_1)\, p_G(x)$$
$$p_{M_2}(x) = f_2\, p_Q(x) + (1 - f_2)\, p_G(x)$$



ATL-PHYS-PUB-2023-032

Data

We use the 2011 CMS Open dataset, which includes both real collisions at 7 TeV, as well as full Monte Carlo (MC) simulation.

To serve as mixtures M_1 and M_2, we select ***Z+jet*** and ***dijet*** respectively.

We use the simulated dijet sample as a labelled quark/gluon dataset.

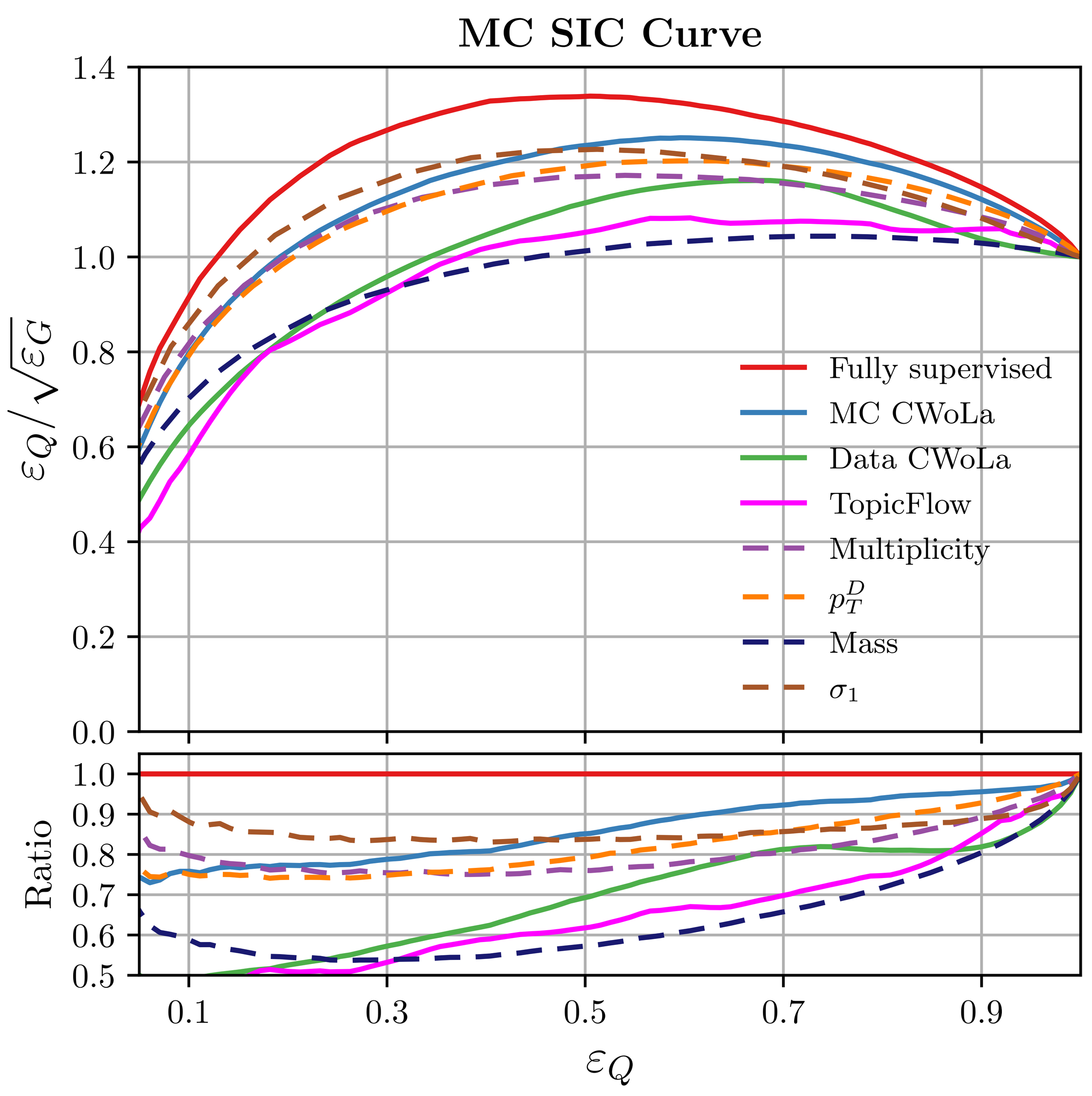
Dataset	Total events	Quarks	Gluons
Data [Zj]	41,773	—	—
Data [jj]	82,162	—	—
MC [Zj]	95,324	70,568	24,756
MC [jj]	3,064,713	868,556	2,196,157

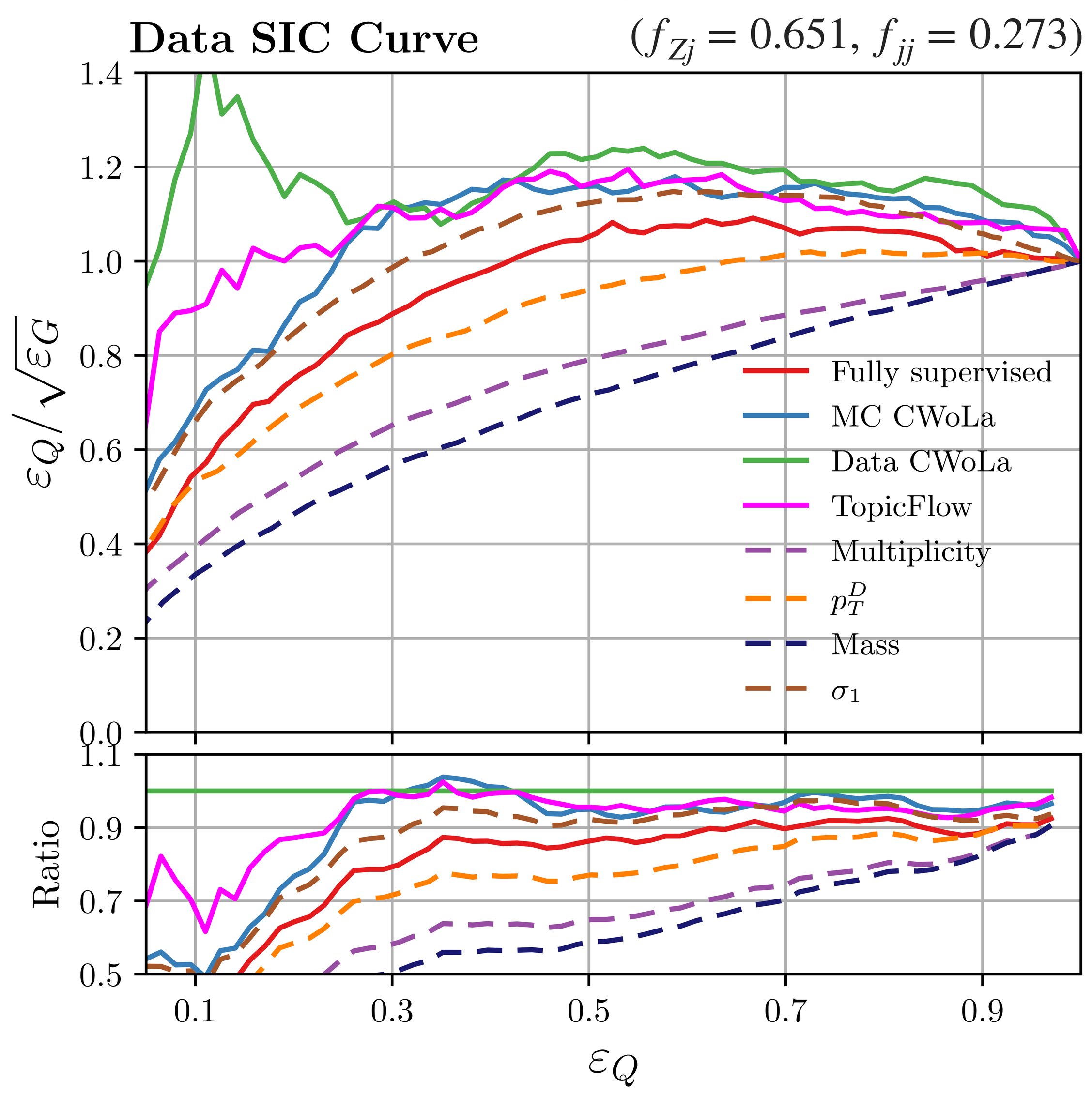
We train 3 classifiers:

Data CWoLa: Z+jet vs dijet (data)

MC CWoLa: Z+jet vs dijet (sim)

Fully Supervised: Quark vs Gluon (dijet sim)





Performance

Full supervision is best on MC, but what about on data? We need to know f_1, f_2 to answer this.

Jet Topics [2] provides a data-driven estimate. Assuming *'mutual irreducibility'*:

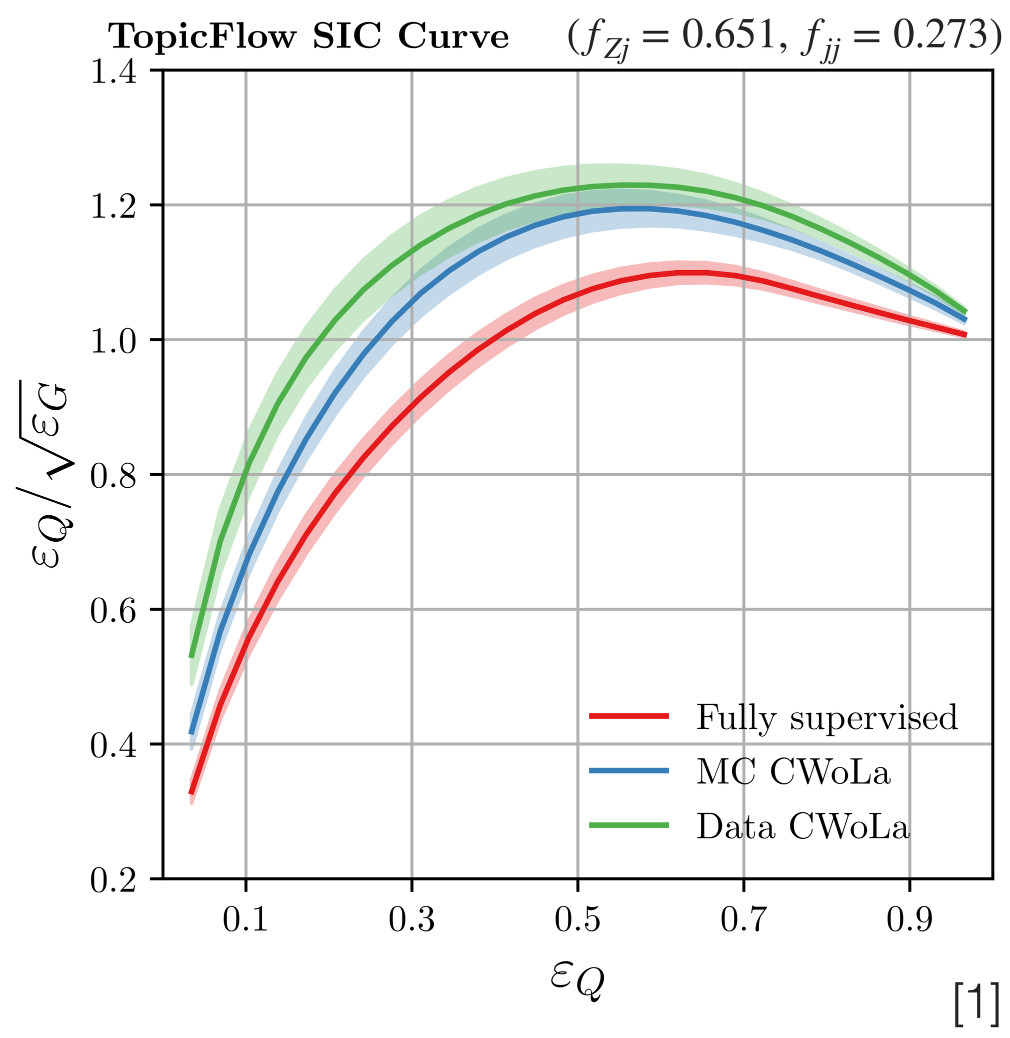
$$f_1, f_2 \iff \min_x \frac{p_{M_1}(x)}{p_{M_2}(x)},\ \max_x \frac{p_{M_1}(x)}{p_{M_2}(x)}$$

The ratios can be approximated by classifiers.

	Quark fraction	
Method	**Data [Zj]**	**Data [jj]**
MC labels	0.740	0.301
Jet Topics	0.651	0.273
Topics + MC	0.784	0.329

While the absolute discrimination power depends on the choice of fractions, the rankings are robust.

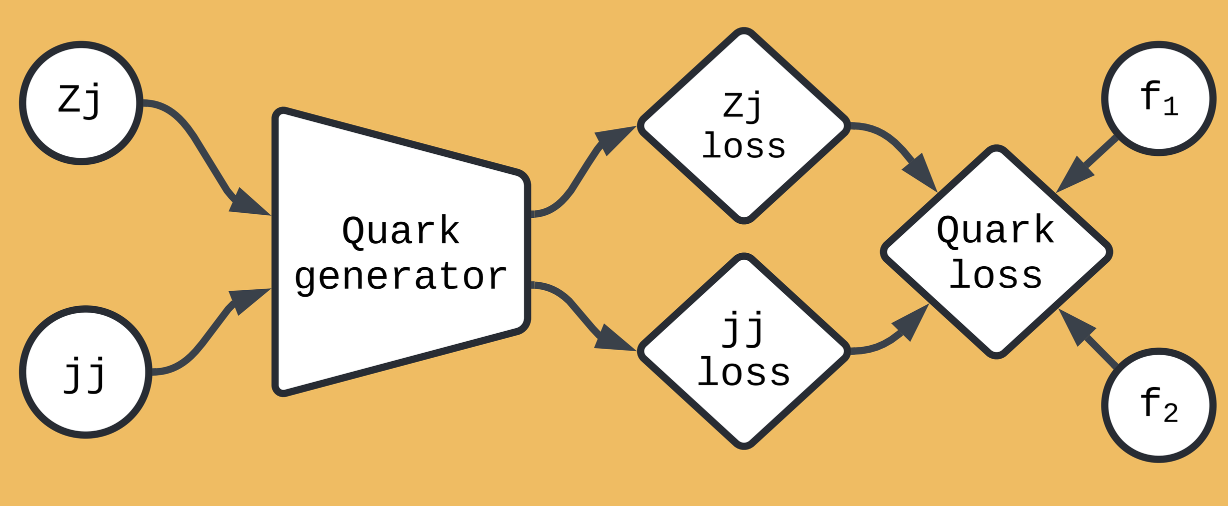
The data-trained classifier appears to be the best quark/gluon discriminator in data.



TopicFlow

With estimated f_1, f_2, we can train a generative model to **extract *pure quark/gluon distributions*** from mixed training samples [3].

We train a normalising flow in this way. It can then be used for ***generative classification***, and to smooth ***statistical fluctuations.***



[1] Metodiev et al. JHEP10(2017)174 [2] Metodiev et al. Phys.Rev.Lett.120,241602 [3] AO and MD PhysRevD.107.114003