New Approaches to Anomaly Detection

David Shih

February 5, 2020

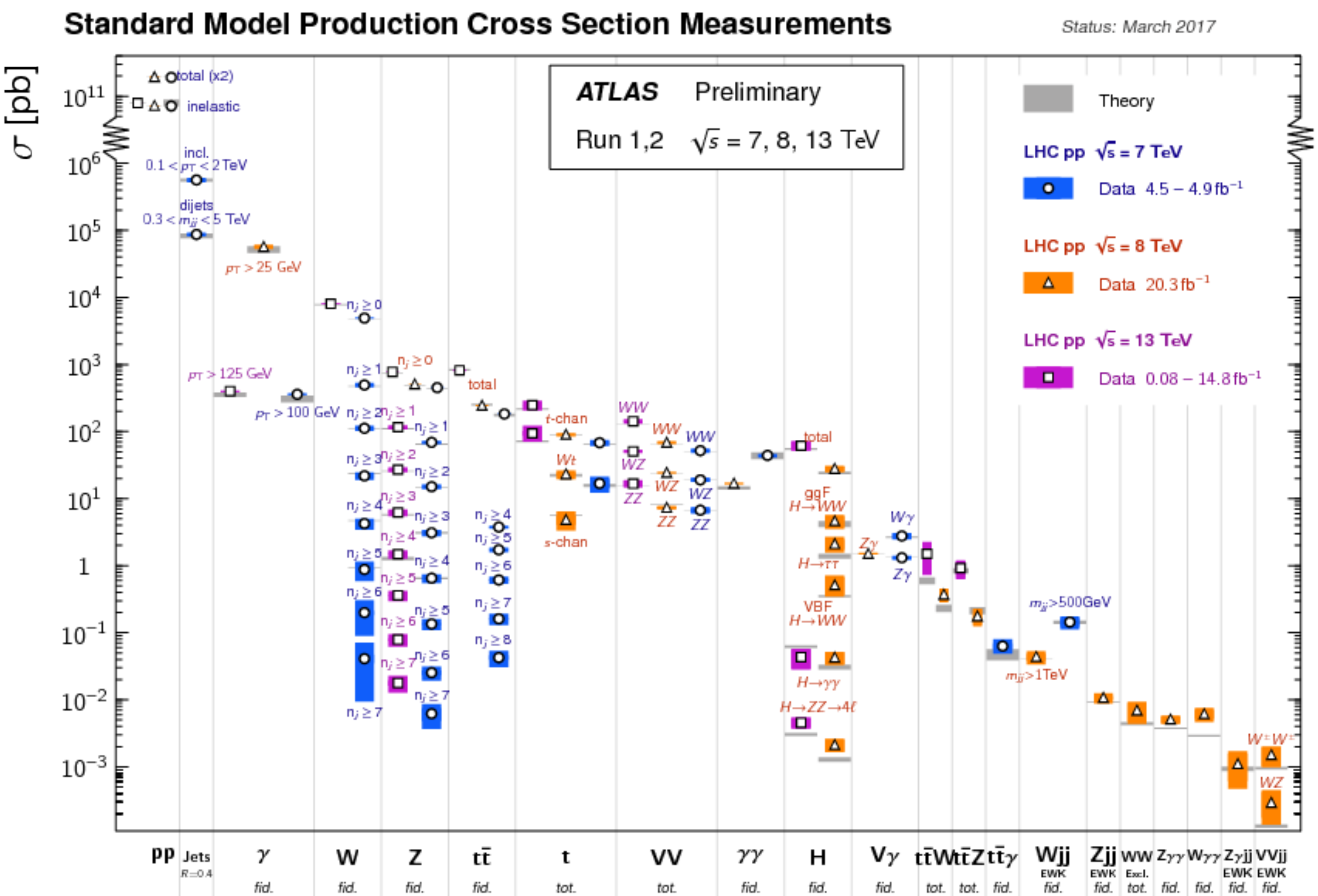
"Machine Learning at LHC"
KMI Nagoya

Outline

1. Motivation and setup
2. Conventional approaches
3. New approaches and the LHC Olympics 2020

1. Motivation and Setup

Tests of the Standard Model

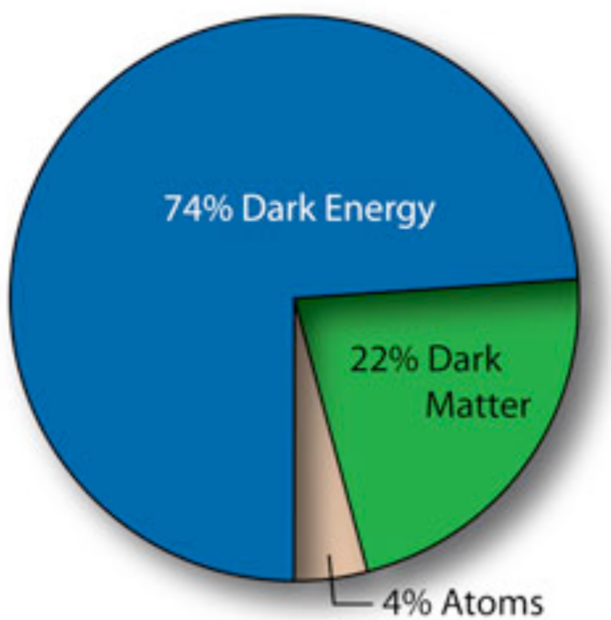


The SM has withstood the test of time. (~50 years!)

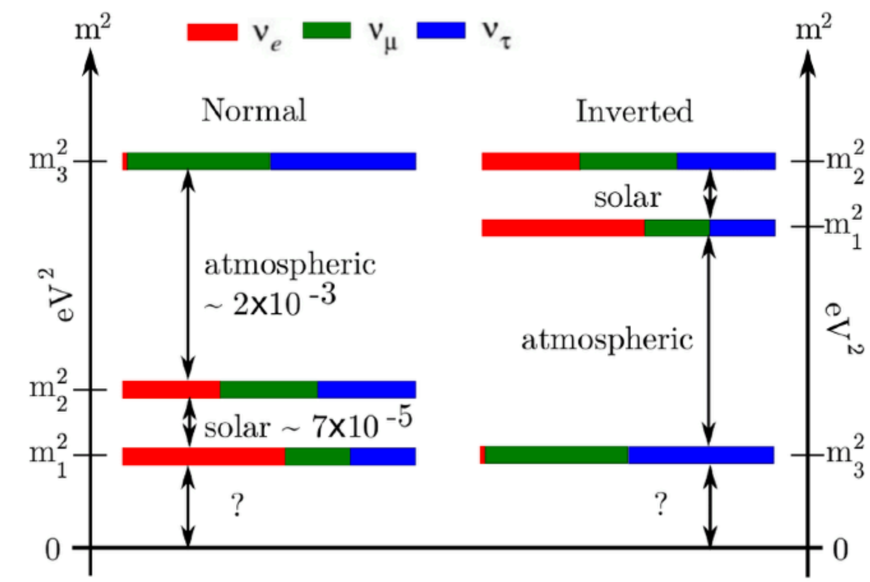
Agreement between theory and experiment across ~14 orders of magnitude.

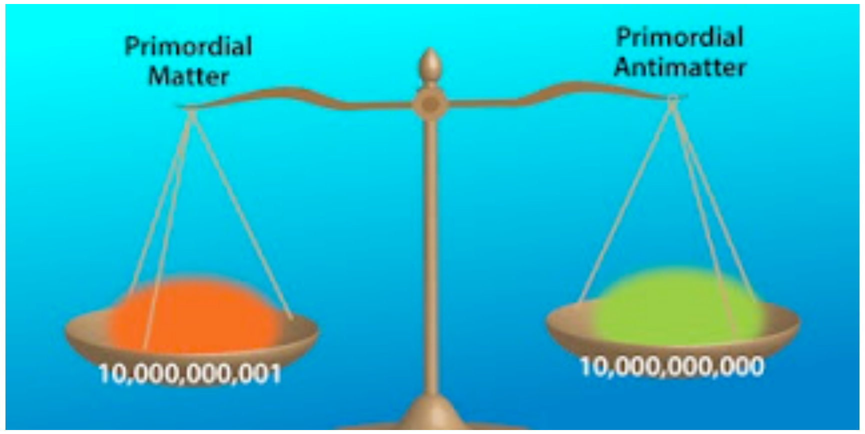
Yet we know there's new physics out there…

dark matter



neutrino masses



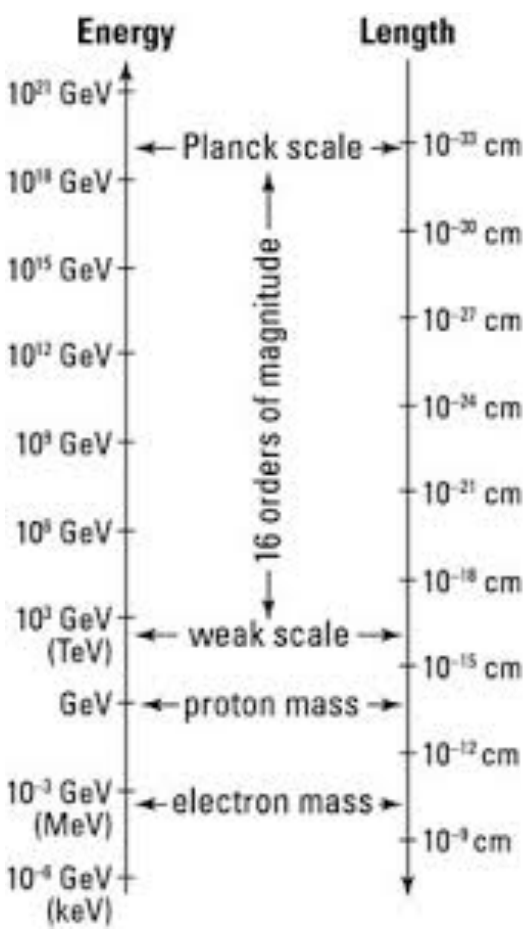


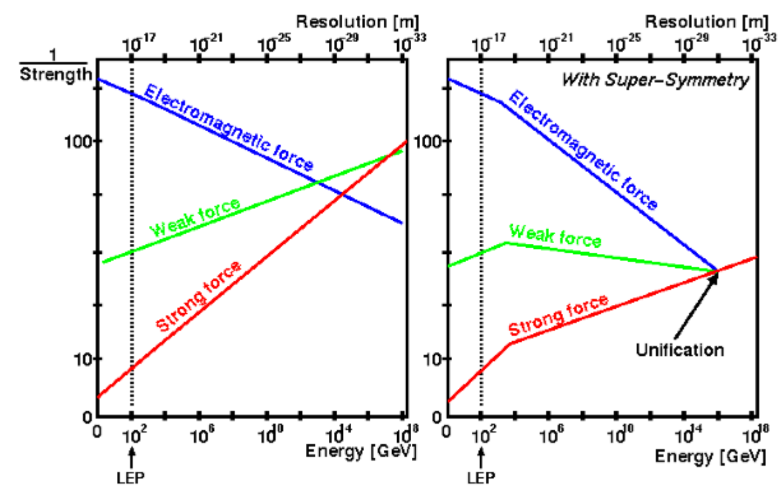
matter/anti-matter asymmetry

Yet we know there's new physics out there…

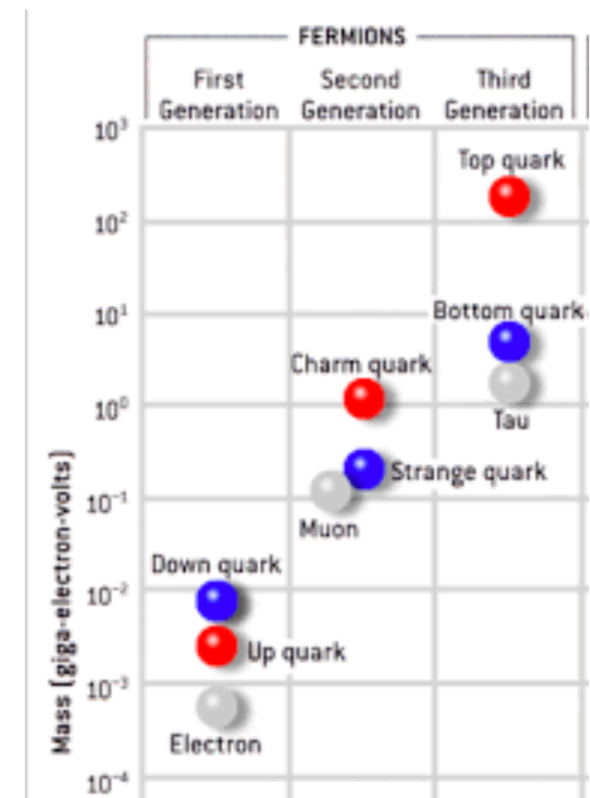
grand unification

hierarchy problem





flavor puzzle



$$\mathcal{L} \supset \theta \, \frac{\alpha_s}{8\pi} G_{\mu\nu}\tilde{G}^{\mu\nu}$$

$$\theta \lesssim 10^{-10}$$

strong CP problem

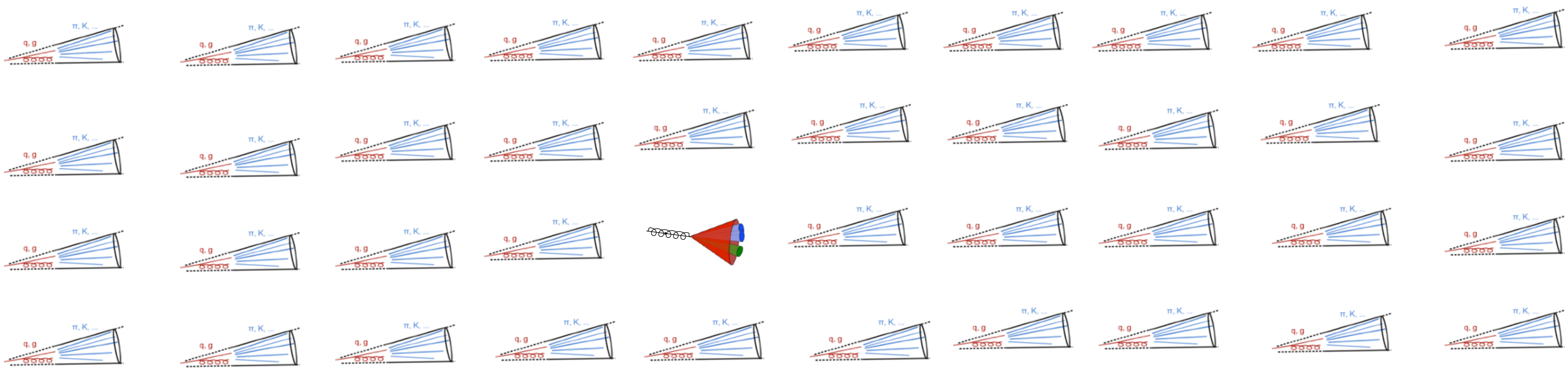
LHC and Big Data

At the LHC we have the highest energy collider ever built, generating copious amounts of data.

- 600 million collisions per second
- Raw data rate ~ 1 PB/s (1 PB=10^6 GB)
- Actual data rate ~ 25 GB/s
    - Need to trigger on 1 out of 40,000 events
- ~ 10's of PB annually

There is enormous discovery potential for new physics!

Anomaly Detection at the LHC

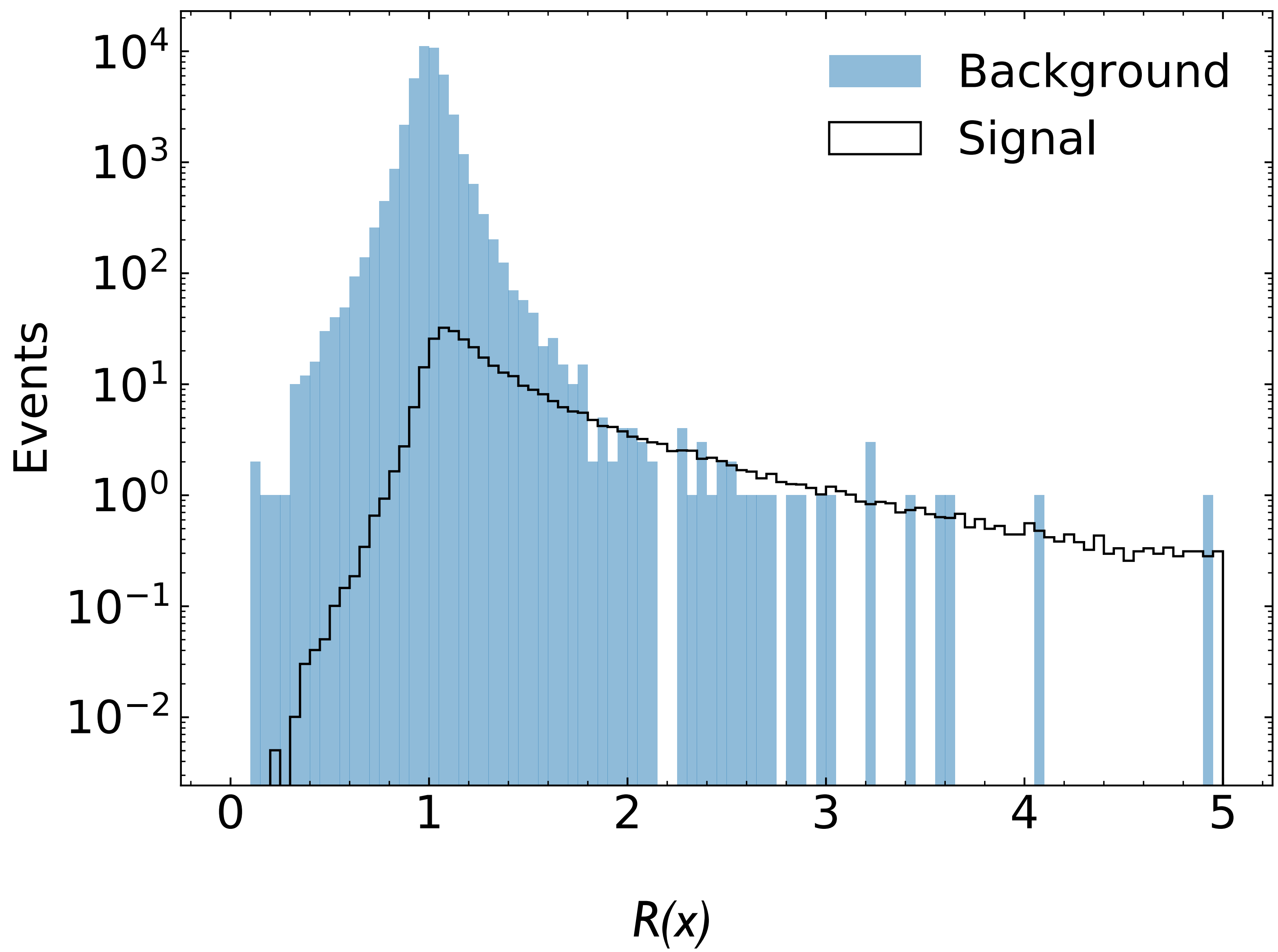


But if there is physics beyond the SM in the data, it's likely to be rare and surrounded by SM background. *Otherwise we would have seen it already!*

This calls for

- sophisticated techniques to dig the signal out of the data.
- careful and precise background estimation

Anomaly Detection at the LHC

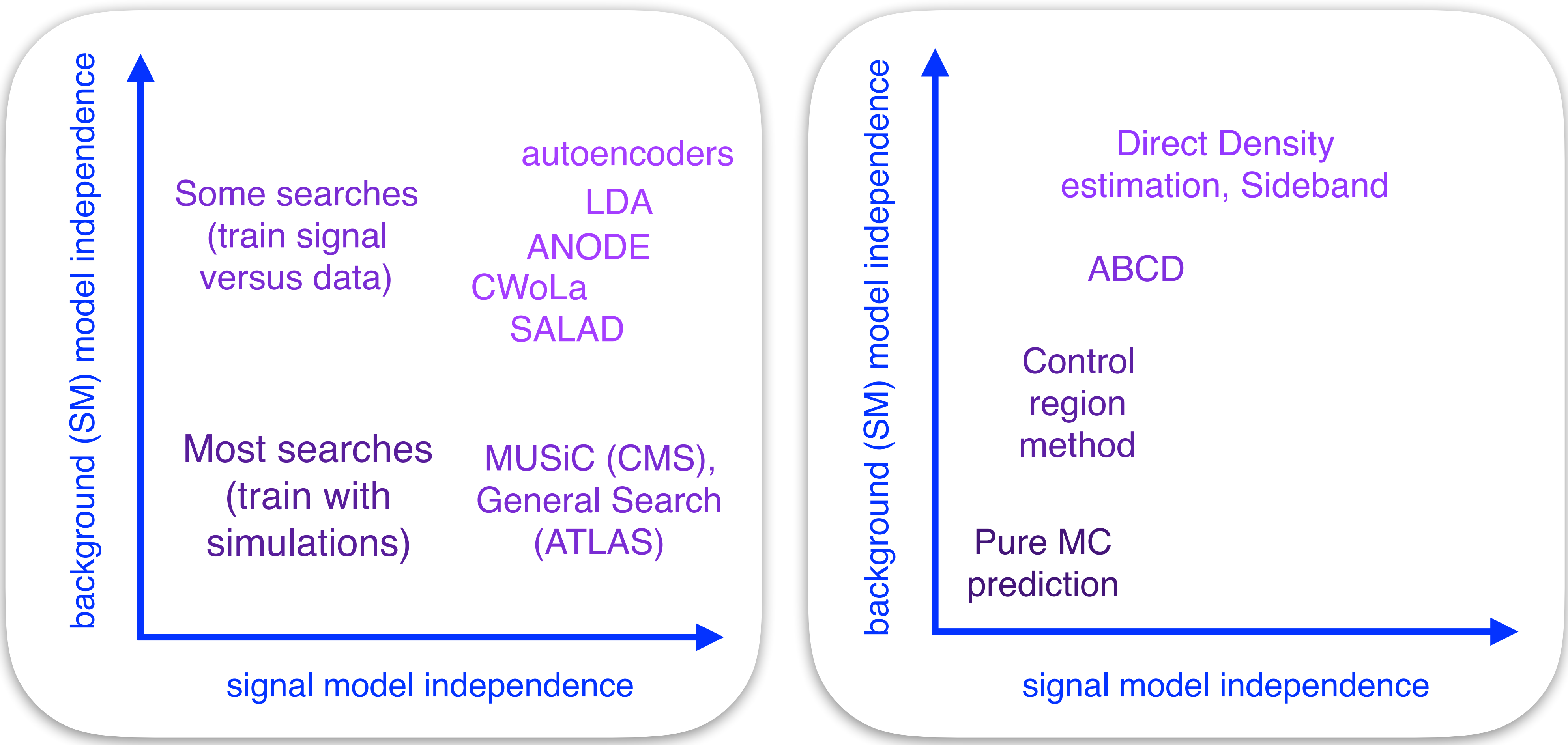


Generally, the idea is to design and optimize a discriminant sensitive to new physics vs. SM background.

Anomaly Detection at the LHC

Searches make different levels of assumptions about the signal model (e.g. gluinos, general resonance, anything …) and the background model (SM simulation or data-driven).

from Nachman & DS 2001.04990



(a) Signal sensitivity

(b) Background specificity

2. Conventional methods

The most common approach:

Model specific searches

$$R_S(x) = \frac{\mathcal{L}(x|S_{sim})}{\mathcal{L}(x|B_{sim})}$$

The most common approach:

Model specific searches

$$R_S(x) = \frac{\mathcal{L}(x|S_{sim})}{\mathcal{L}(x|B_{sim})}$$

x: some set of relevant features characterizing each event (almost always binned)

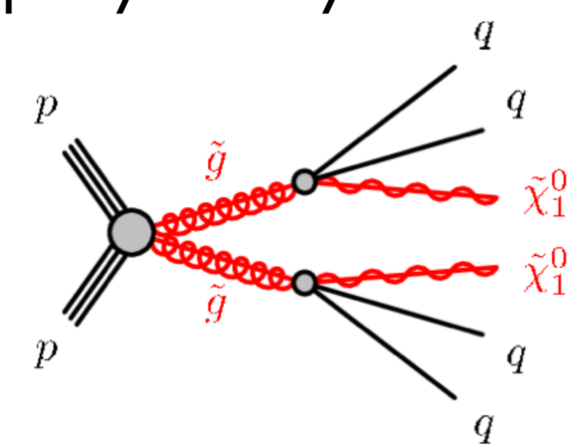
The most common approach:

Model specific searches

$$R_S(x) = \frac{\mathcal{L}(x|S_{sim})}{\mathcal{L}(x|B_{sim})}$$

x: some set of relevant features characterizing each event (almost always binned)

S: a *specific* signal model, e.g. supersymmetry



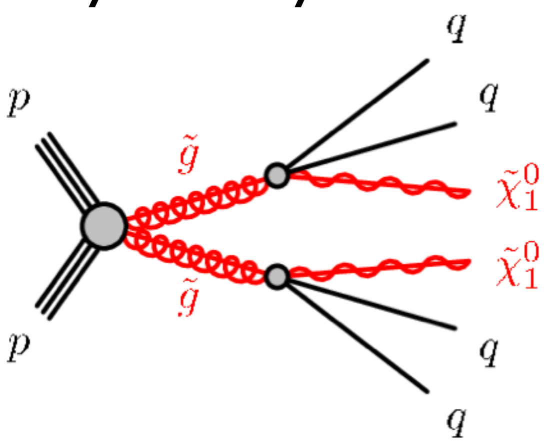
The most common approach:

Model specific searches

$$R_S(x) = \frac{\mathcal{L}(x|S_{sim})}{\mathcal{L}(x|B_{sim})}$$

x: some set of relevant features characterizing each event (almost always binned)

S: a *specific* signal model, e.g. supersymmetry



Rely on simulations of background and signal to construct likelihood ratio.

If simulations are sufficiently accurate (generally not the case!), then optimal for specific S by Neyman-Pearson Lemma

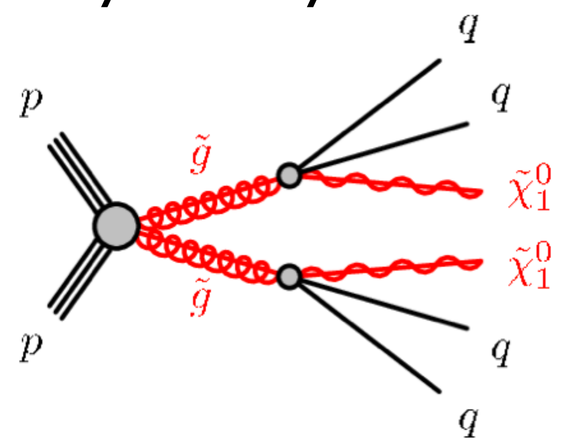
The most common approach:
Model specific searches

signal and
background model
dependent

$$R_S(x) = \frac{\mathcal{L}(x|S_{sim})}{\mathcal{L}(x|B_{sim})}$$

x: some set of relevant features characterizing each event (almost always binned)

S: a *specific* signal model, e.g. supersymmetry



Rely on simulations of background and signal to construct likelihood ratio.

If simulations are sufficiently accurate (generally not the case!), then optimal for specific S by Neyman-Pearson Lemma

The most common approach:

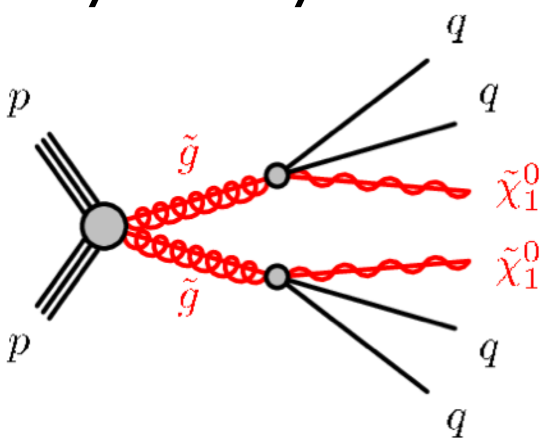
Model specific searches

signal and background model *dependent*

$$R_S(x) = \frac{\mathcal{L}(x|S_{sim})}{\mathcal{L}(x|B_{sim})}$$

x: some set of relevant features characterizing each event (almost always binned)

S: a *specific* signal model, e.g. supersymmetry

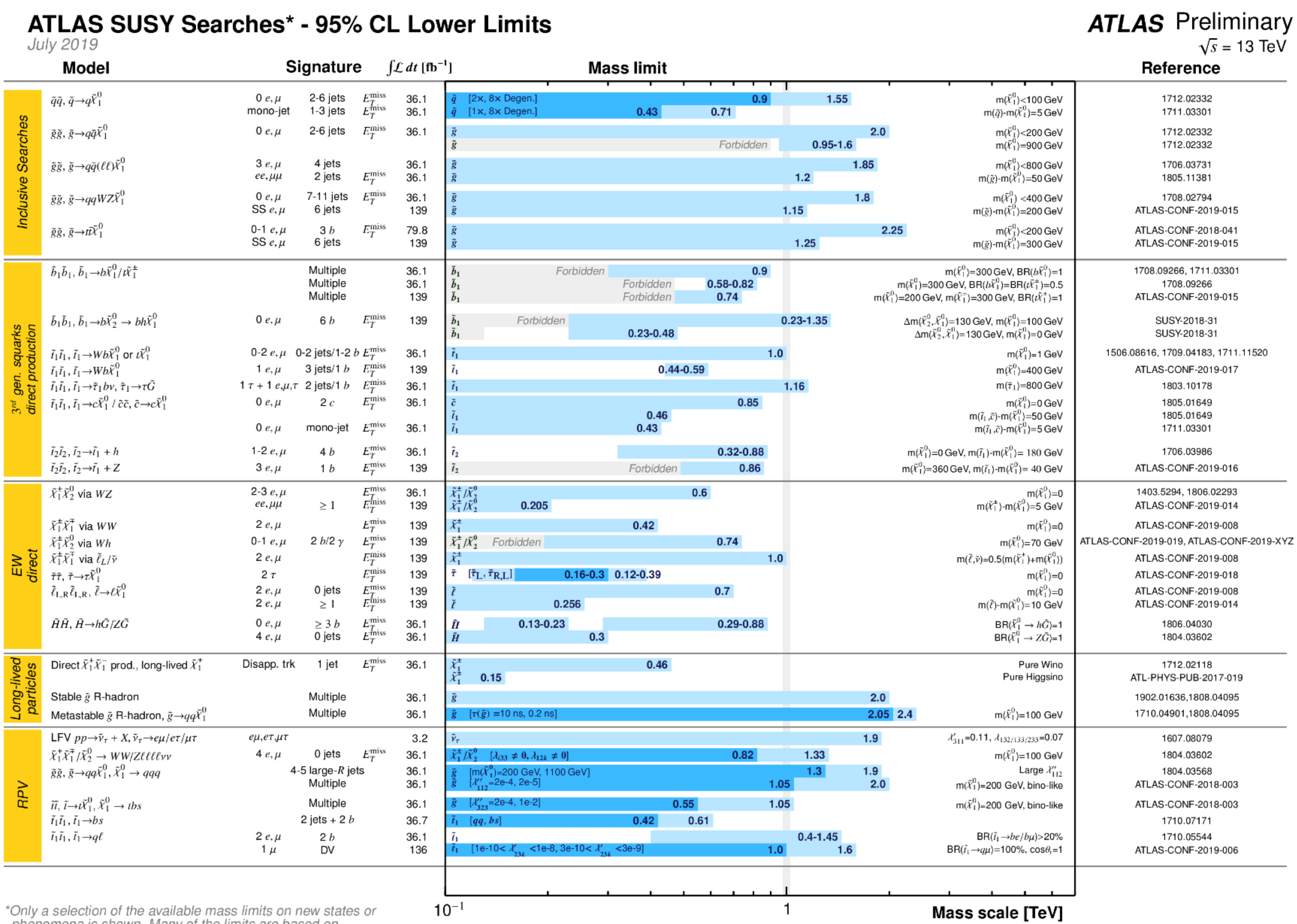


Rely on simulations of background and signal to construct likelihood ratio.

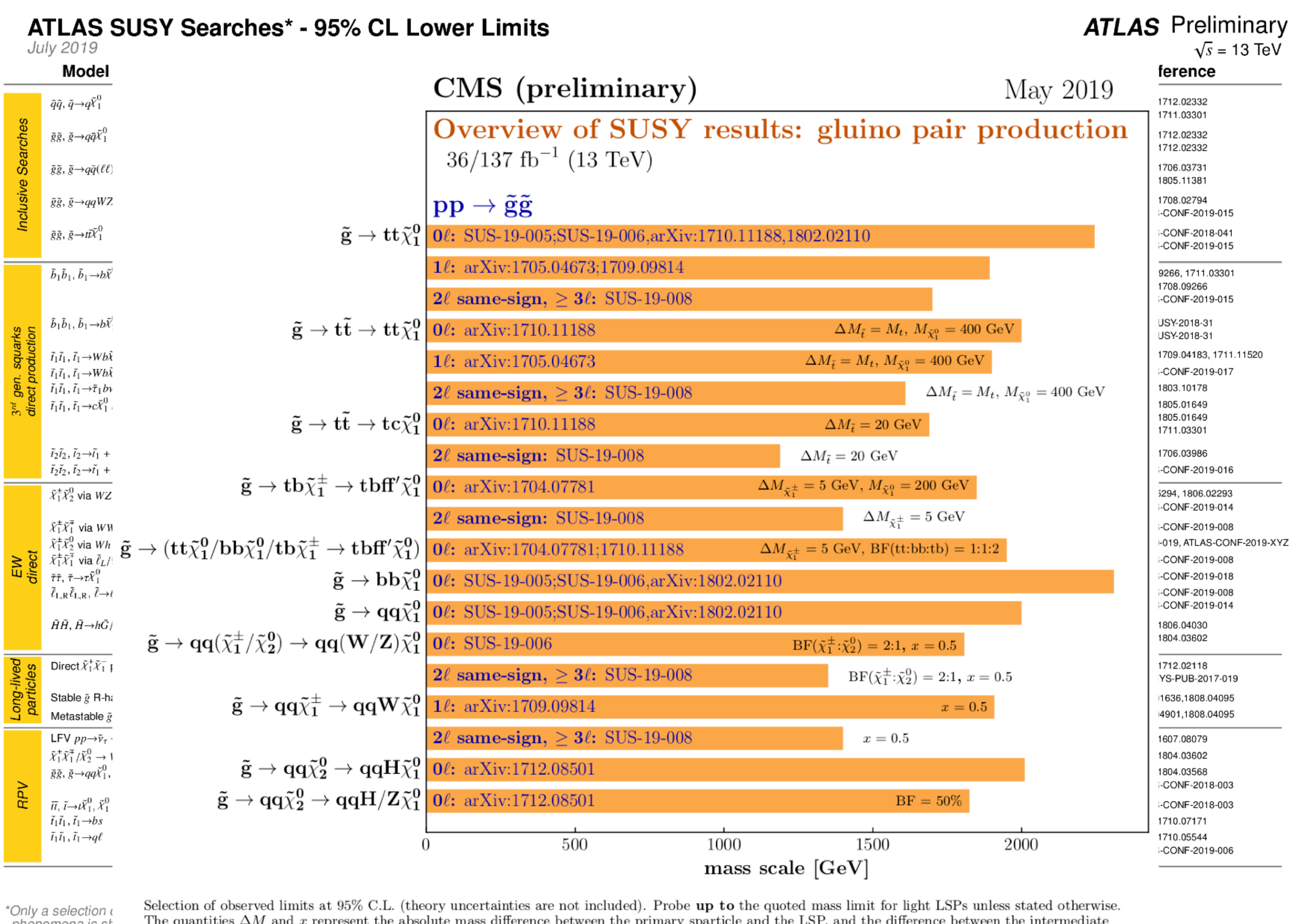
If simulations are sufficiently accurate (generally not the case!), then optimal for specific S by Neyman-Pearson Lemma

> 99% of searches at the LHC are of this type

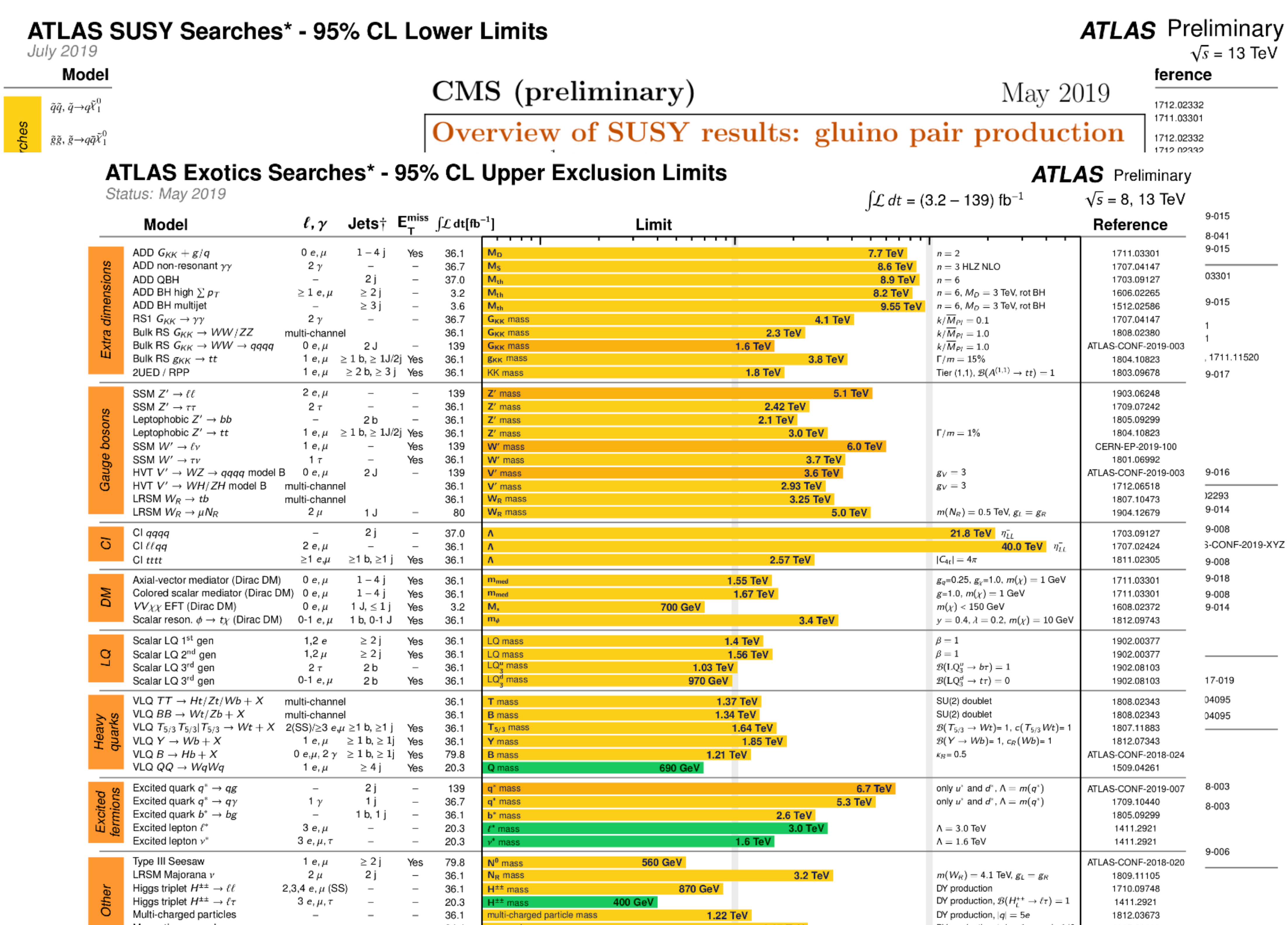
Current Status of NP Searches @ LHC



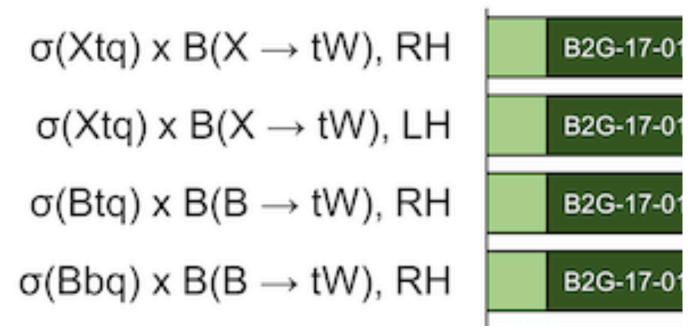
Current Status of NP Searches @ LHC

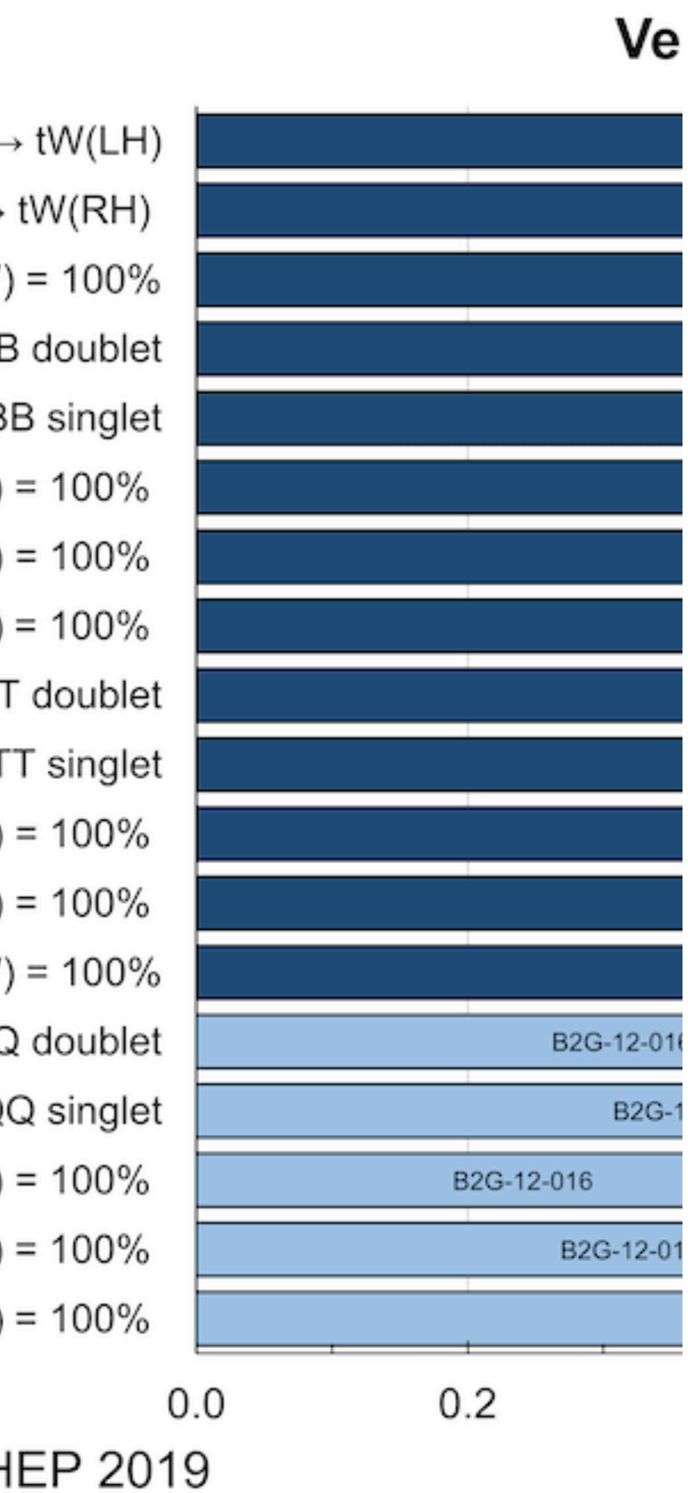


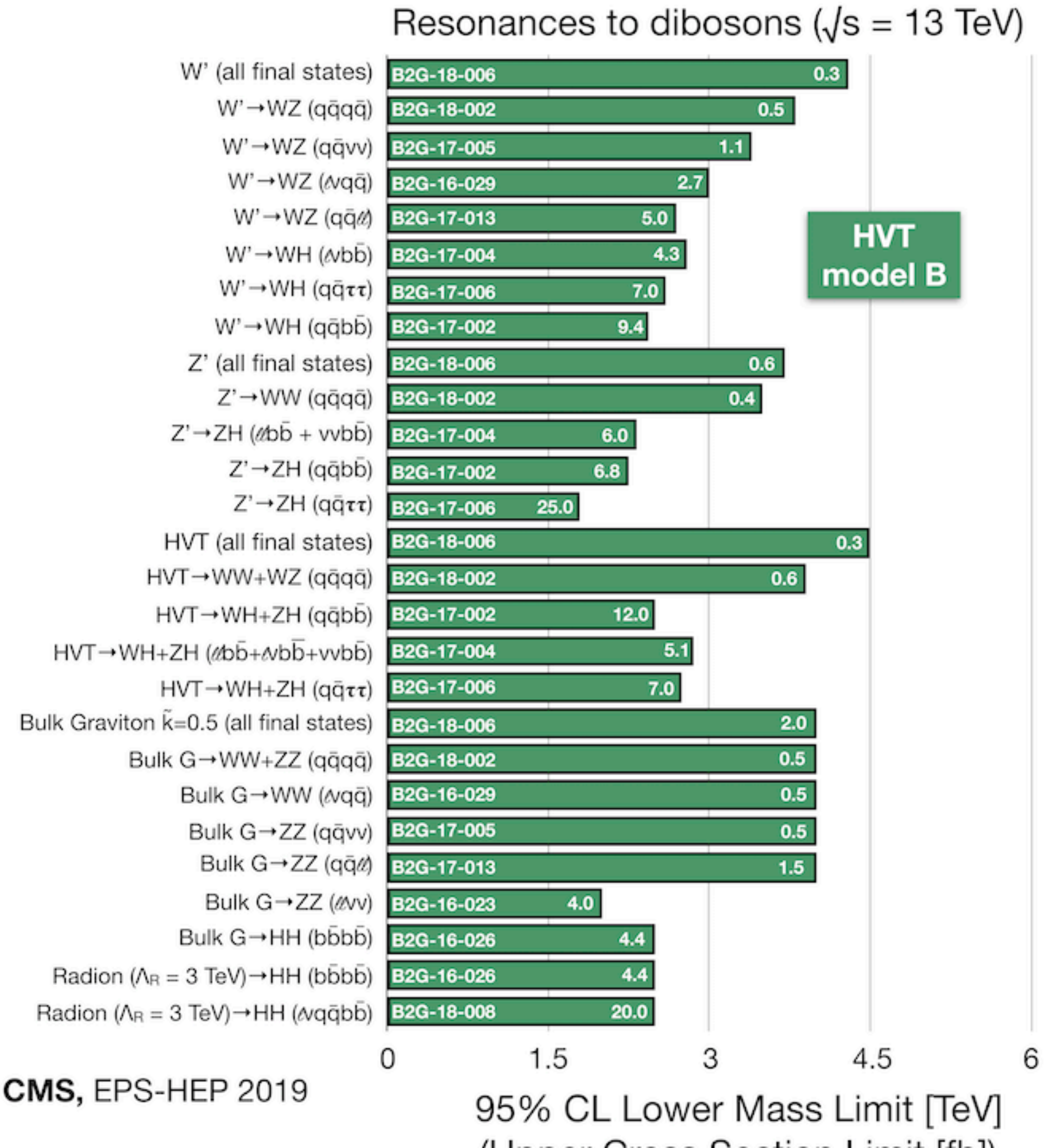
Current Status of NP Searches @ LHC



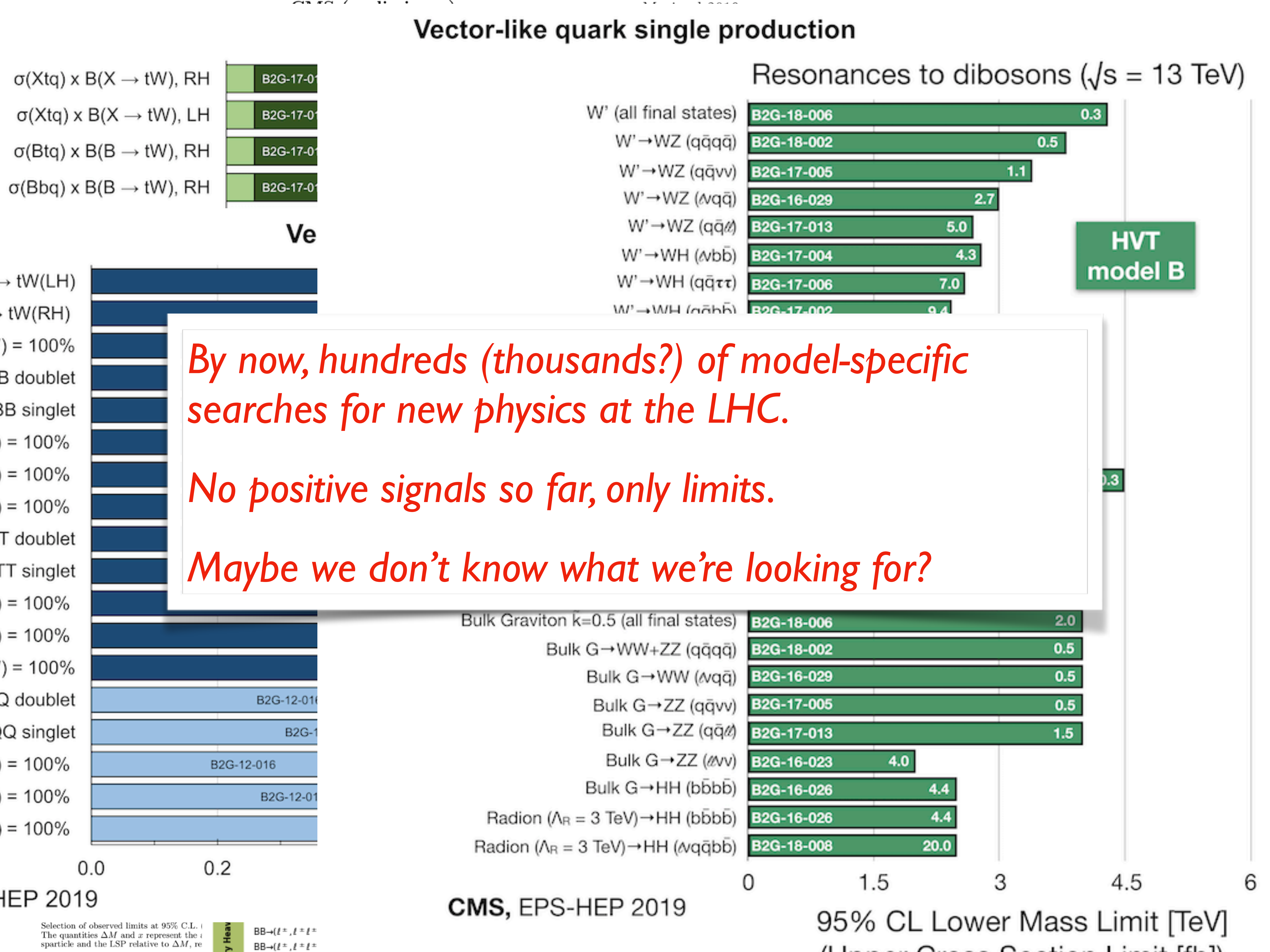
Vector-like quark single production







Selection of observed limits at 95% C.L. (
The quantities ΔM and x represent the
sparticle and the LSP relative to ΔM, re



By now, hundreds (thousands?) of model-specific searches for new physics at the LHC.

No positive signals so far, only limits.

Maybe we don't know what we're looking for?

Previous model-independent approaches

“the general search”

$$R(x) = \frac{\mathcal{L}(x|data)}{\mathcal{L}(x|B_{sim})}$$

Previous model-independent approaches

"the general search"

$$R(x) = \frac{\mathcal{L}(x|data)}{\mathcal{L}(x|B_{sim})}$$

x: a single feature (almost always binned)

Idea: compare data vs ***simulated*** SM background in 1D histograms.

Previous model-independent approaches

"the general search"

fully signal model *independent*
background model *dependent*

$$R(x) = \frac{\mathcal{L}(x|data)}{\mathcal{L}(x|B_{sim})}$$

x: a single feature (almost always binned)

Idea: compare data vs ***simulated*** SM background in 1D histograms.

Previous model-independent approaches

"the general search"

fully signal model *independent*
background model *dependent*

$$R(x) = \frac{\mathcal{L}(x|data)}{\mathcal{L}(x|B_{sim})}$$

x: a single feature (almost always binned)

Idea: compare data vs ***simulated*** SM background in 1D histograms.

[Can also imagine turbocharged version: train DNN on full phase space to distinguish data from background MC (D'Agnolo, Wulzer et al 1806.02350, 1912.12155)]

Previous model-independent approaches

"the general search"

fully signal model *independent*
background model *dependent*

$$R(x) = \frac{\mathcal{L}(x|data)}{\mathcal{L}(x|B_{sim})}$$

x: a single feature (almost always binned)

Idea: compare data vs ***simulated*** SM background in 1D histograms.

[Can also imagine turbocharged version: train DNN on full phase space to distinguish data from background MC (D'Agnolo, Wulzer et al 1806.02350, 1912.12155)]

Data vs bg likelihood ratio: not optimal for any specific signal, but could be optimal for rejecting background hypothesis.

Previous model-independent approaches

"the general search"

fully signal model *independent*
background model *dependent*

$$R(x) = \frac{\mathcal{L}(x|data)}{\mathcal{L}(x|B_{sim})}$$

x: a single feature (almost always binned)

Idea: compare data vs ***simulated*** SM background in 1D histograms.

[Can also imagine turbocharged version: train DNN on full phase space to distinguish data from background MC (D'Agnolo, Wulzer et al 1806.02350, 1912.12155)]

Data vs bg likelihood ratio: not optimal for any specific signal, but could be optimal for rejecting background hypothesis.

Long but somewhat neglected history of searches of this type

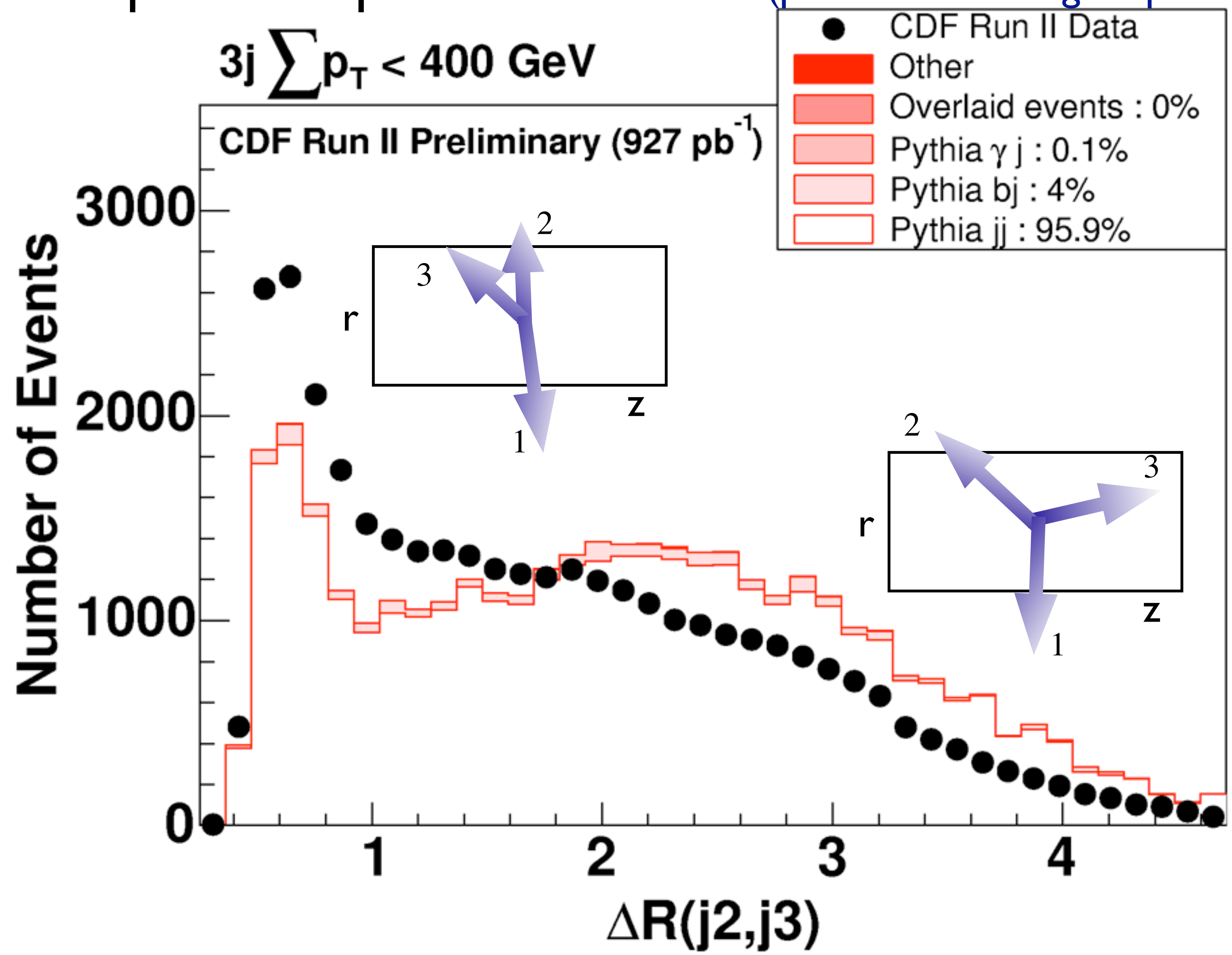
Previous model-independent approaches

A brief history of model independent searches in HEP:

- **D0** — *"Sleuth"* — PRD 62:092004 (2000)
PRD 64:012004 (2001)
PRL 86:3712 (2001)
- **H1 (Hera)** — *"General Search"* — PLB 602:14-30 (2004)
0705.3721
- **CDF** — *"Sleuth/Vista"* — 0712.1311 PRD 78:012002 (2008)
0712.2534 (submitted to PRL, NEVER PUBLISHED)
0809.3781 PRD 79:011101 (2009)
- **CMS** — *"MUSIC"* — CMS-PAS-EXO-14-016
- **ATLAS** — *"Model independent general search"* — 1807.07447 EPJC 79:120 (2019)

An example of what is found

Sample discrepant distribution (parton showering suspected)



From B. Knuteson talk at UMich (2008)

Previous model-independent approaches

"the bump hunt"

partially signal and background model *independent*

$$R(m) = \frac{\mathcal{L}(m|data)}{\mathcal{L}(m|B_{data})}$$

m: a single feature

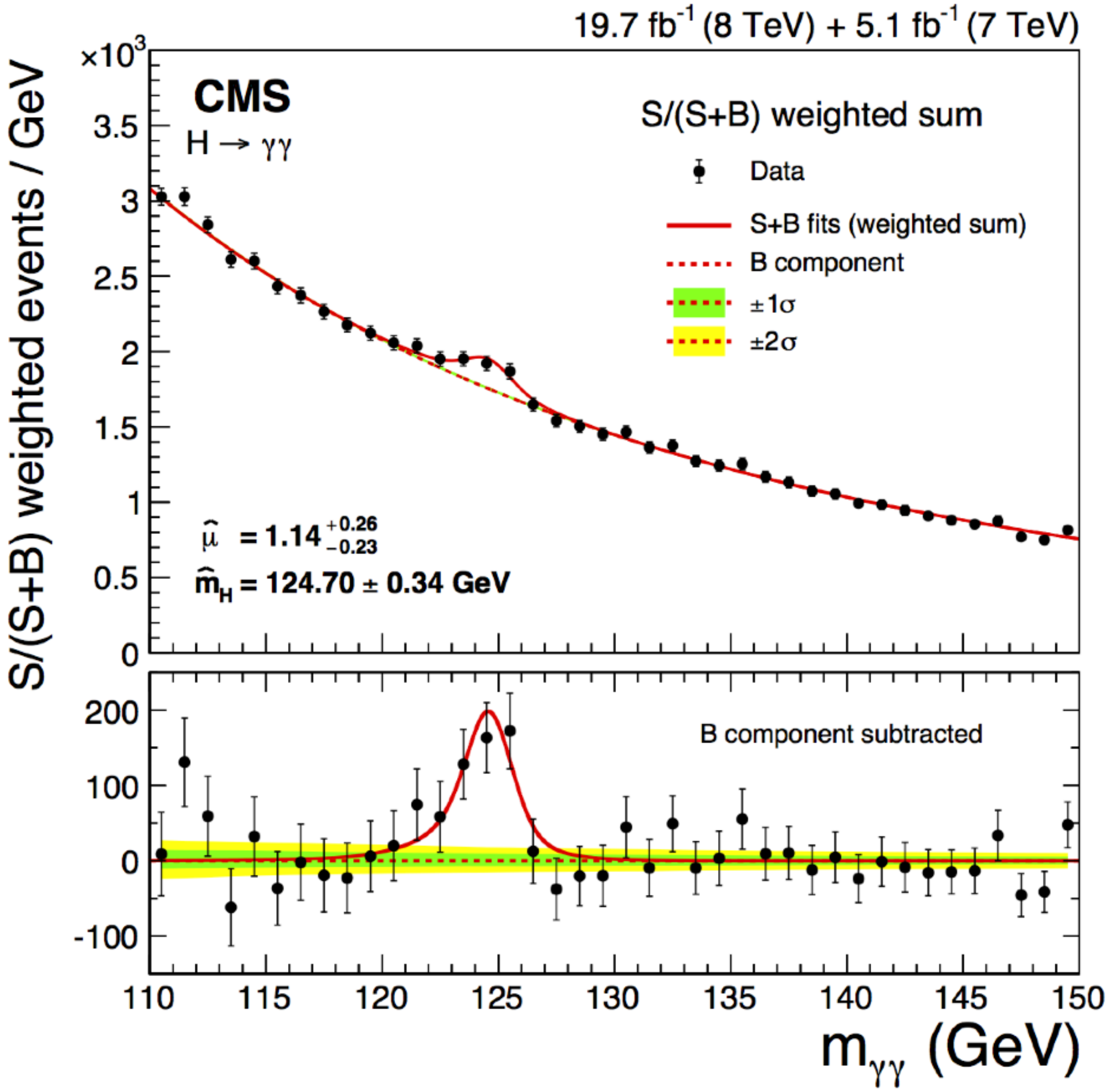
Idea: assume signal is localized in m while background is smooth.

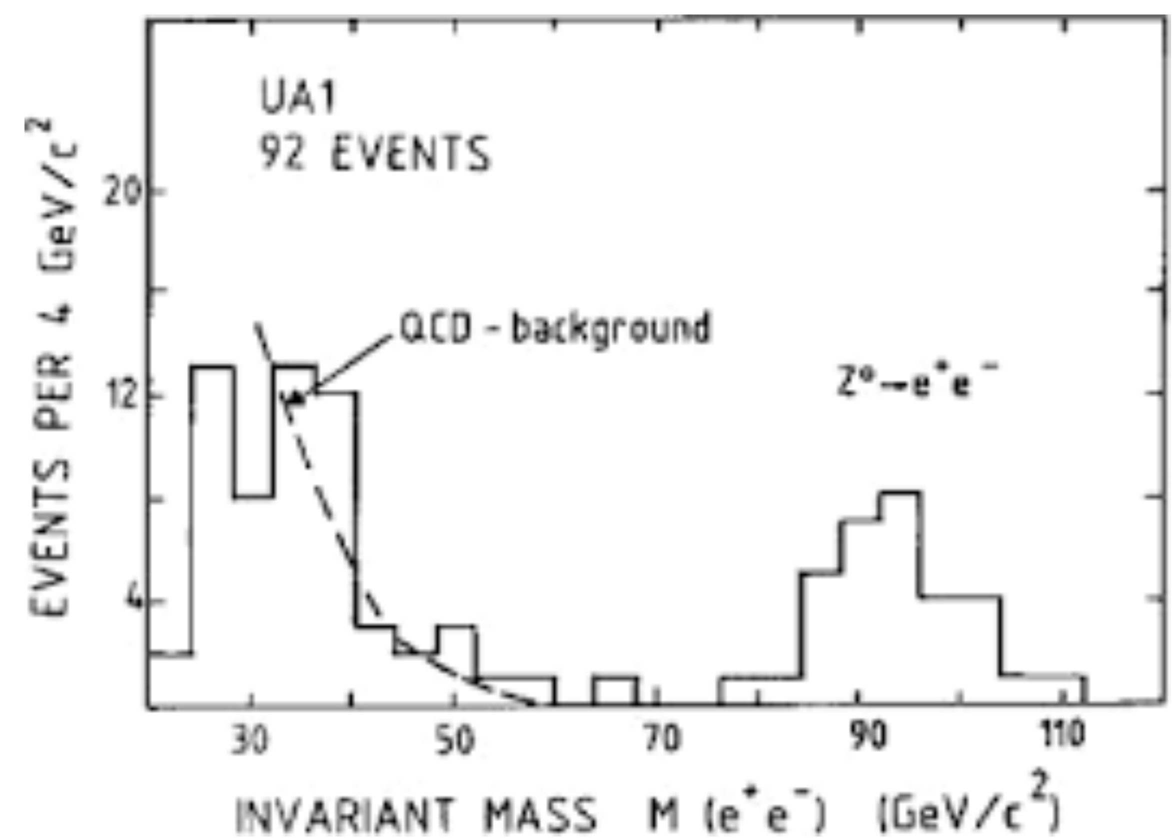
Use ***sidebands*** m∉(m0-δm,m0+δm) to interpolate background into ***signal region*** m∈(m0-δm,m0+δm).

Classic method, used in many discoveries

Previous model-independent approaches

"the bump hunt"

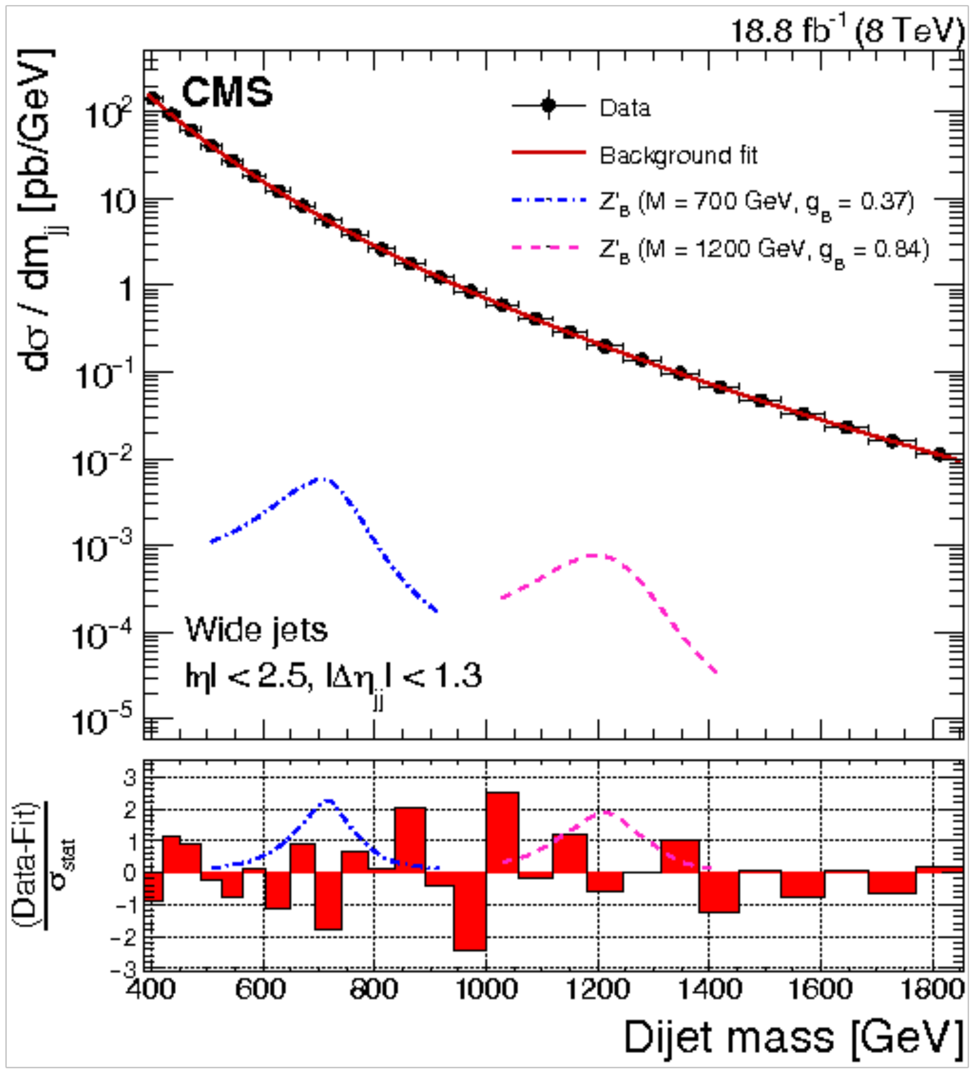


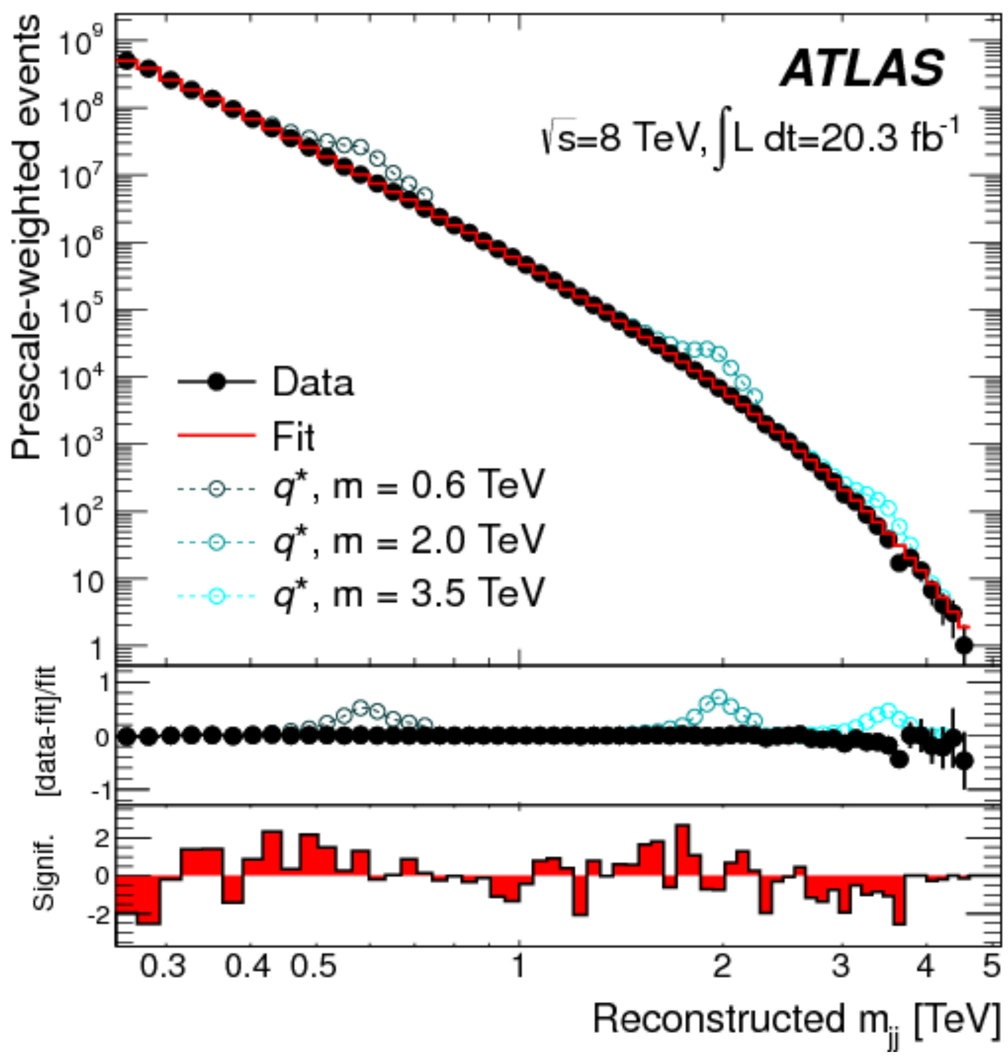


Higgs, Z, …

Previous model-independent approaches

“the bump hunt”

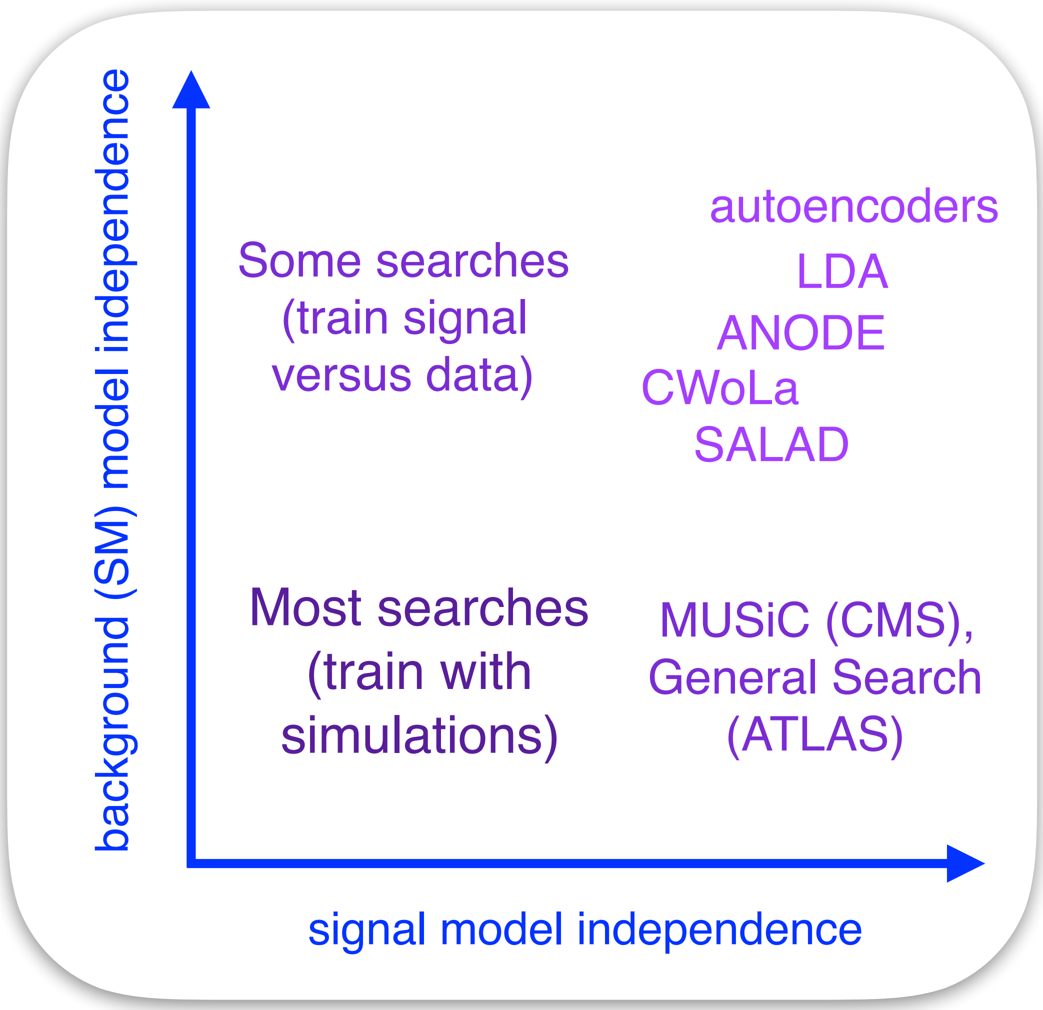




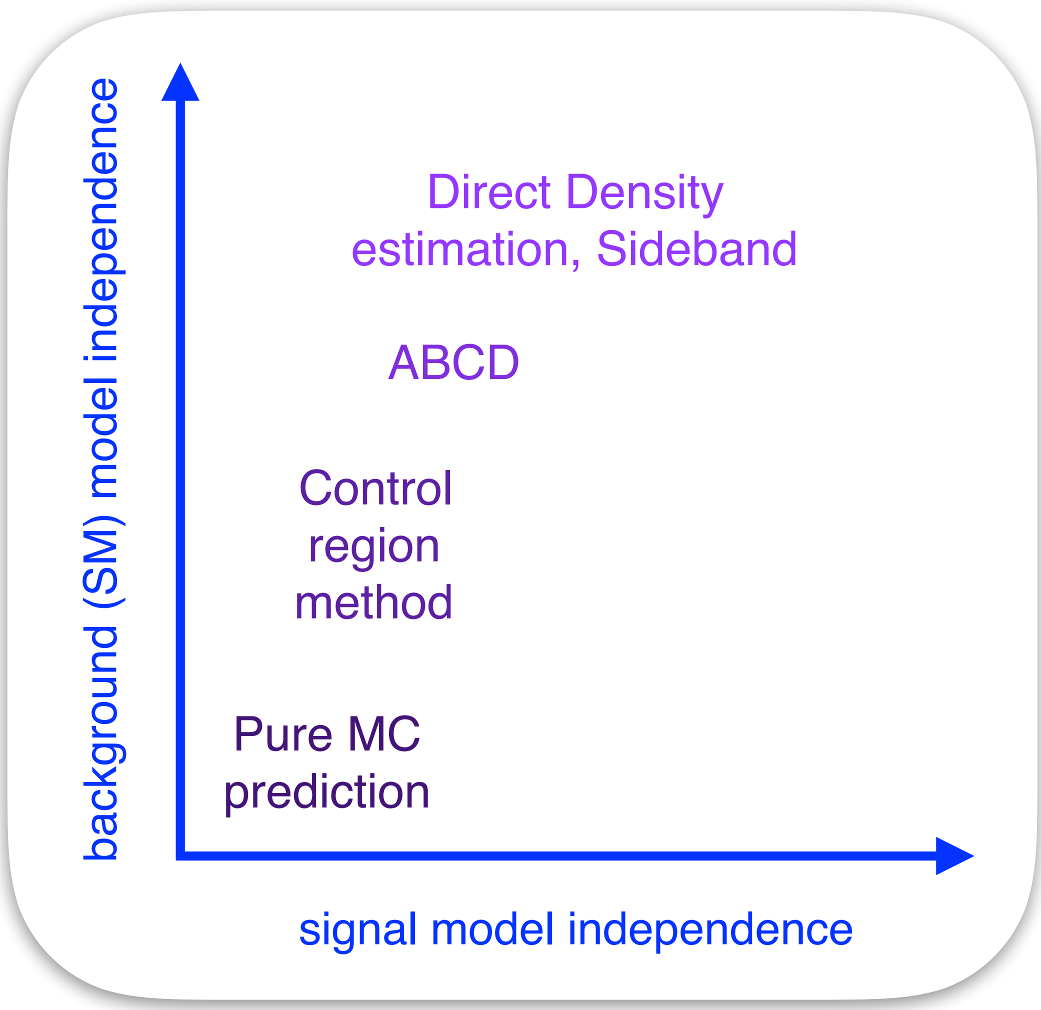
Also a classic search for new physics — e.g. a hypothetical heavy BSM particle that decays to pairs of jets

Overview of search strategies

from Nachman & DS 2001.04990

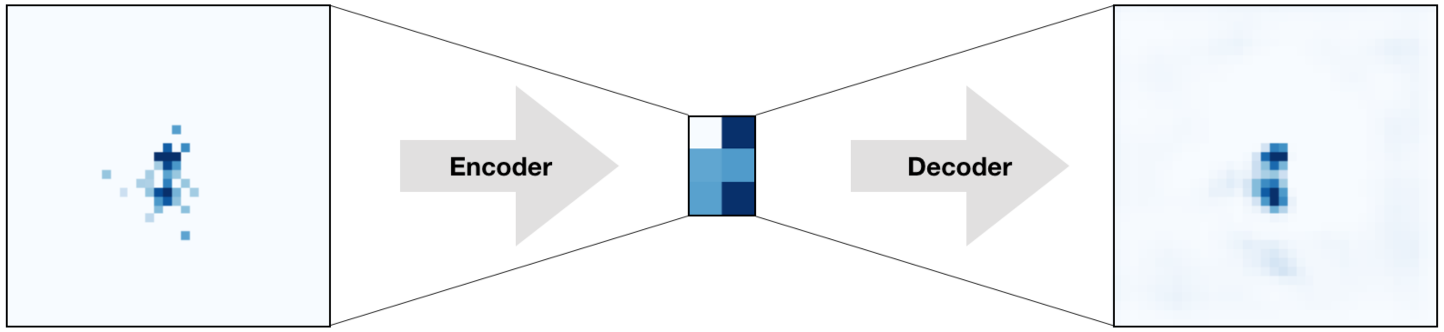


(a) Signal sensitivity

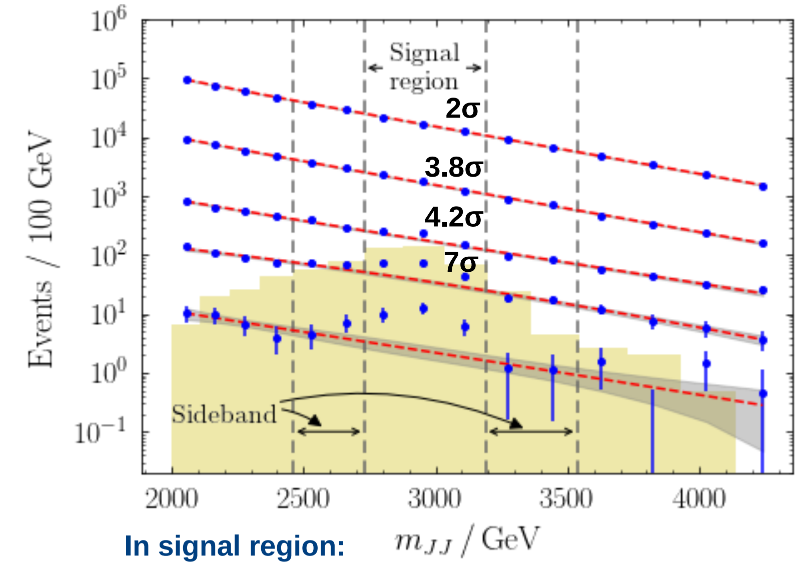


(b) Background specificity

Autoencoders

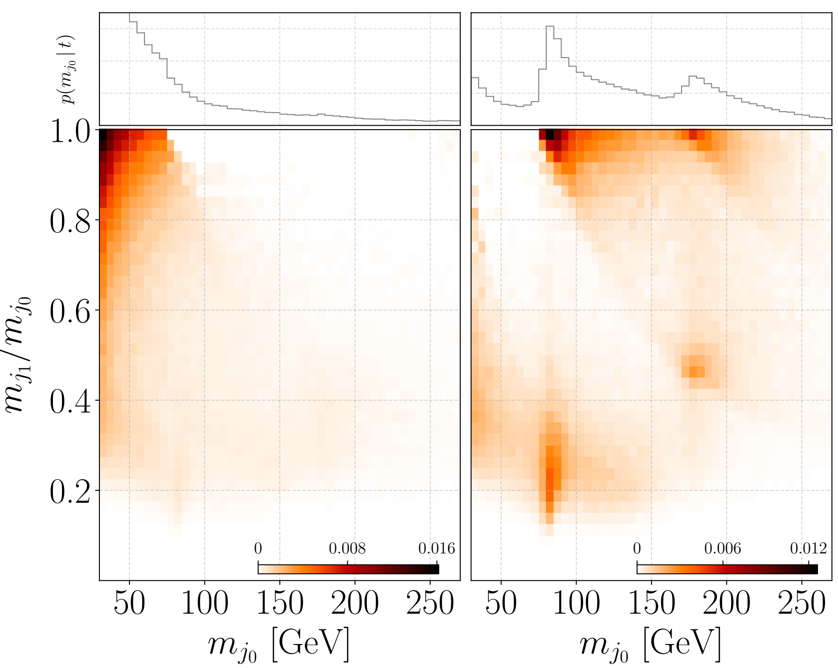


CWoLa

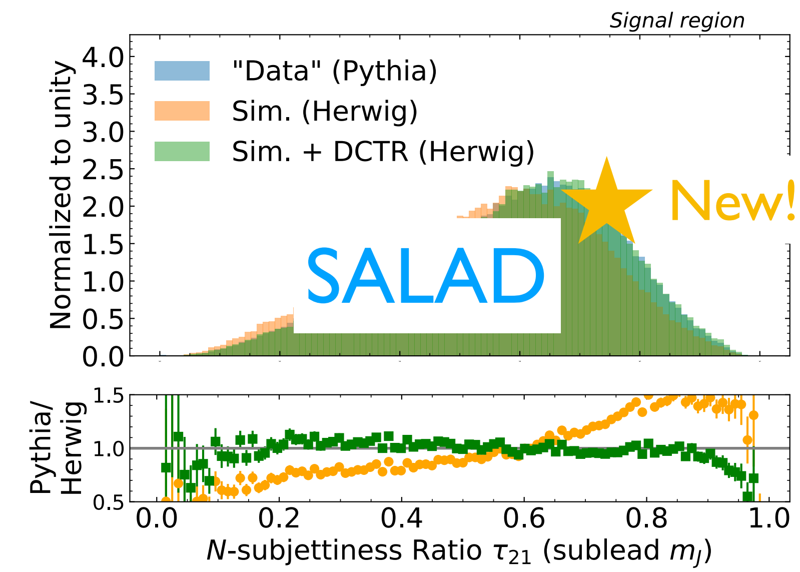


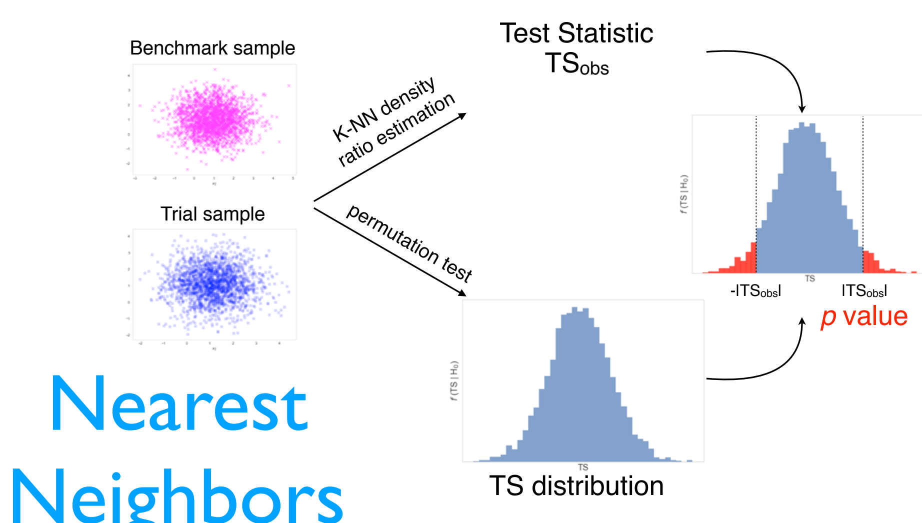
In signal region:
S = 522,
S/B = 0.64%

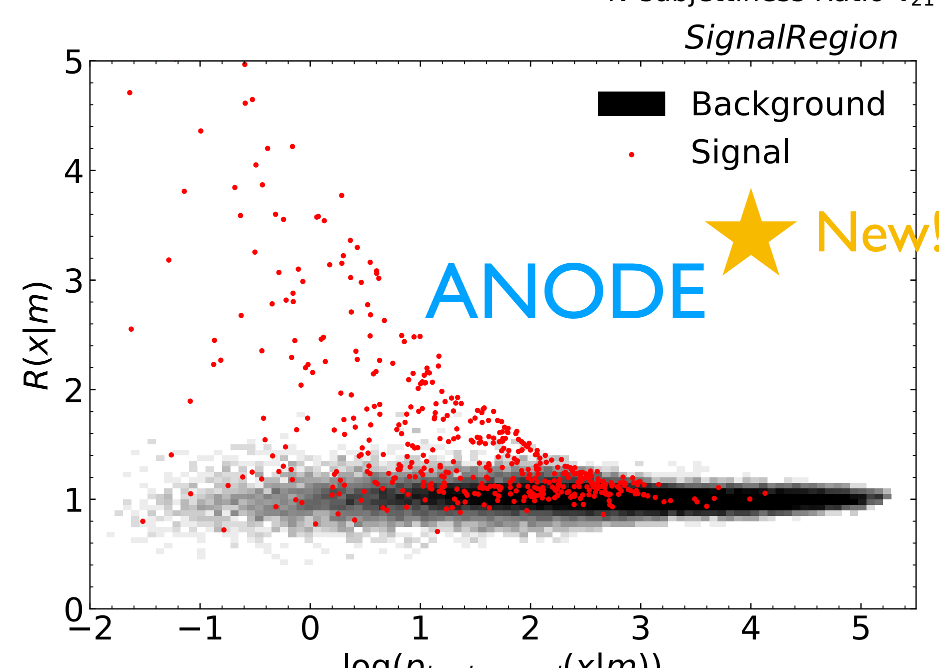
Probabilistic Modeling



3. New approaches

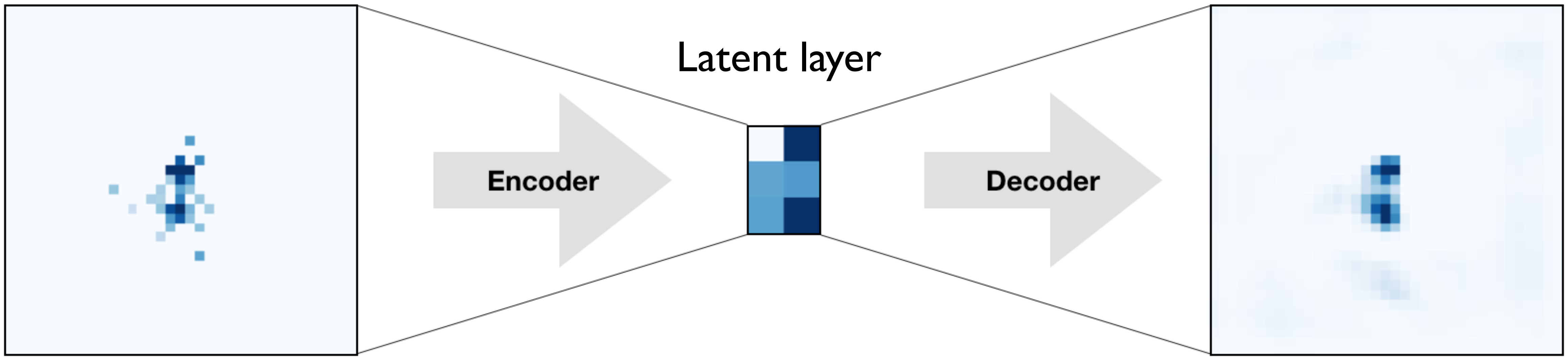






Searching for NP with deep autoencoders

Heimel et al 1808.08979;   Farina, Nakai & DS 1808.08992



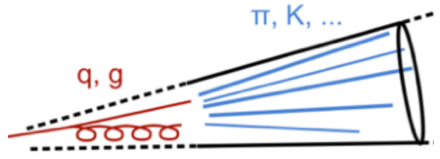
An autoencoder maps an input into a reduced "latent representation" and then attempts to reconstruct the original input from it.

Can use reconstruction error as an anomaly threshold!

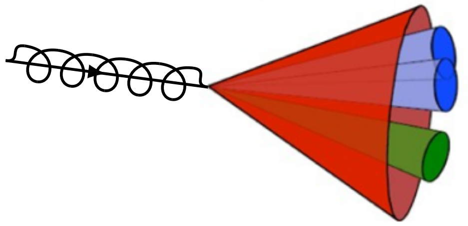
See also:
Hajer et al "Novelty Detection Meets Collider Physics" 1807.10261
Cerri et al "Variational Autoencoders for New Physics Mining at the Large Hadron Collider" 1811.10276

Performance

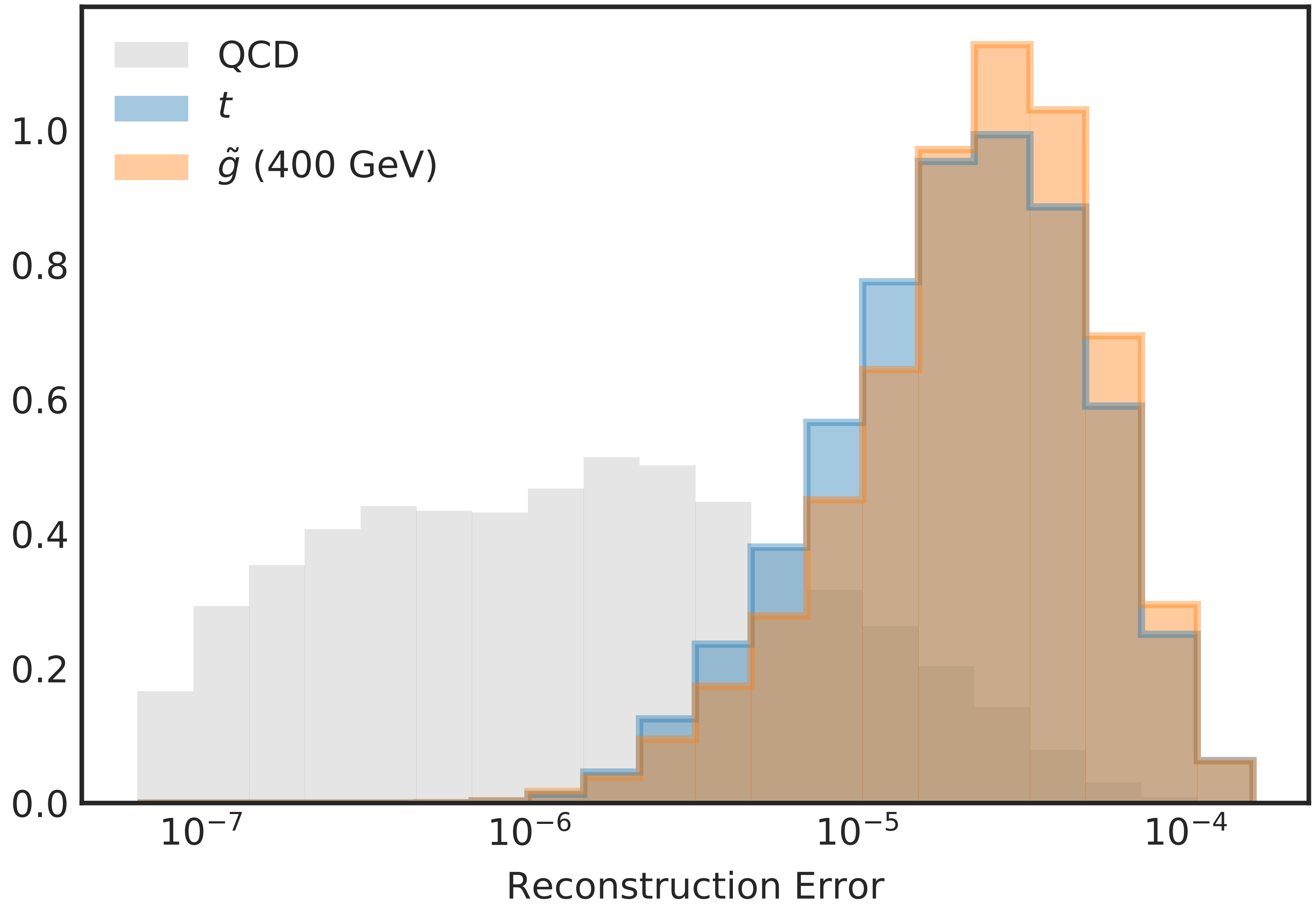
background (QCD)



signal (tops and 400 GeV RPV gluinos)



It works as an anomaly detector!



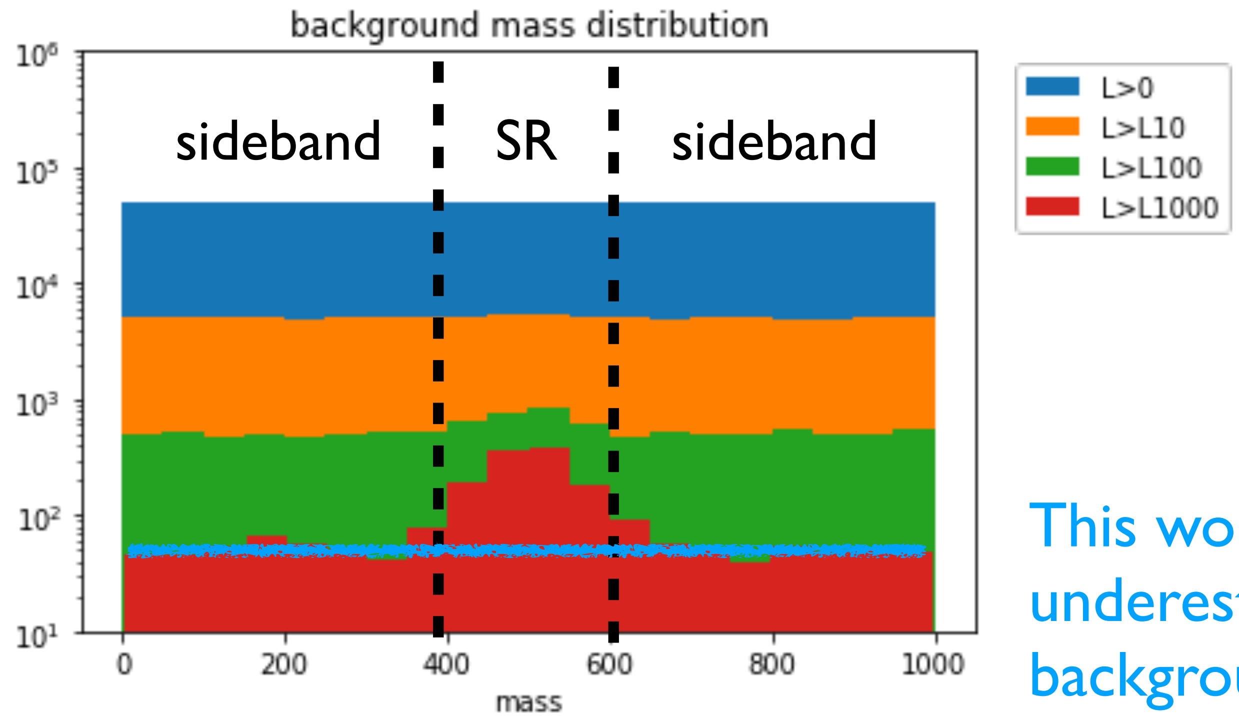
Robust against contamination with signal — can use in fully unsupervised mode

Background estimation

Discriminant is useless without an accurate background prediction.

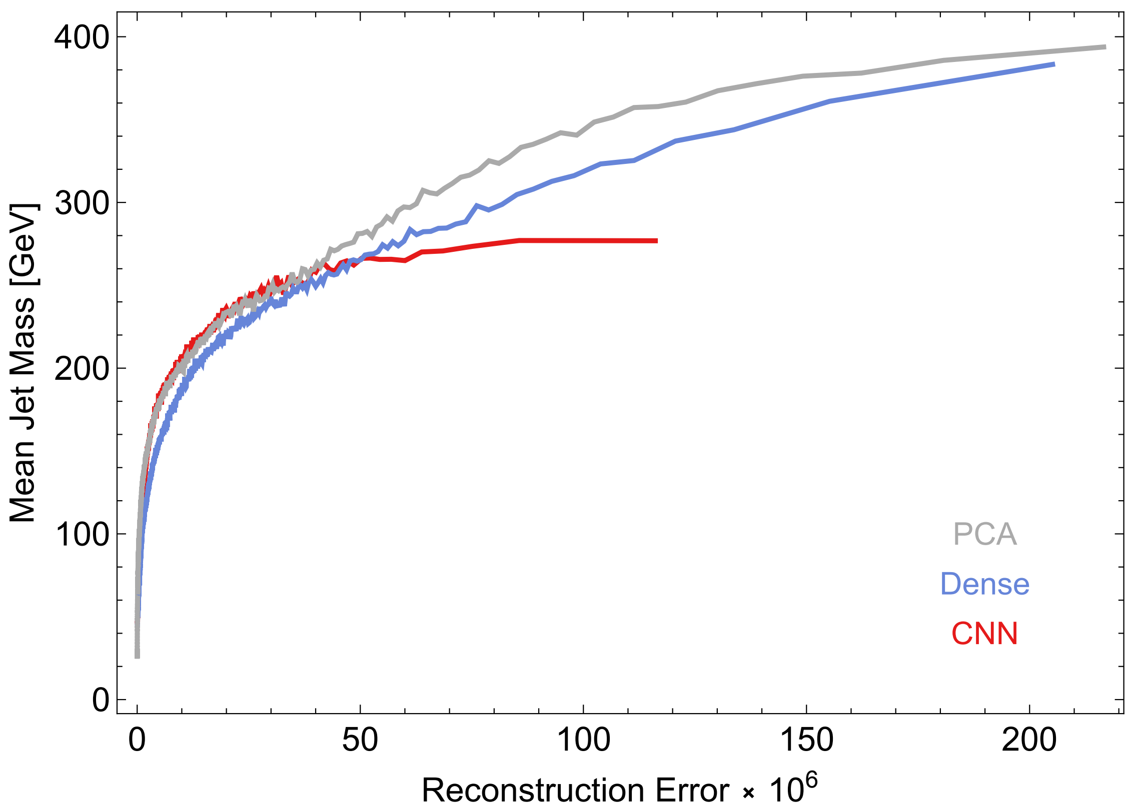
One idea: combine autoencoder with a bump hunt in jet mass.
Estimate backgrounds using sidebands in mass.

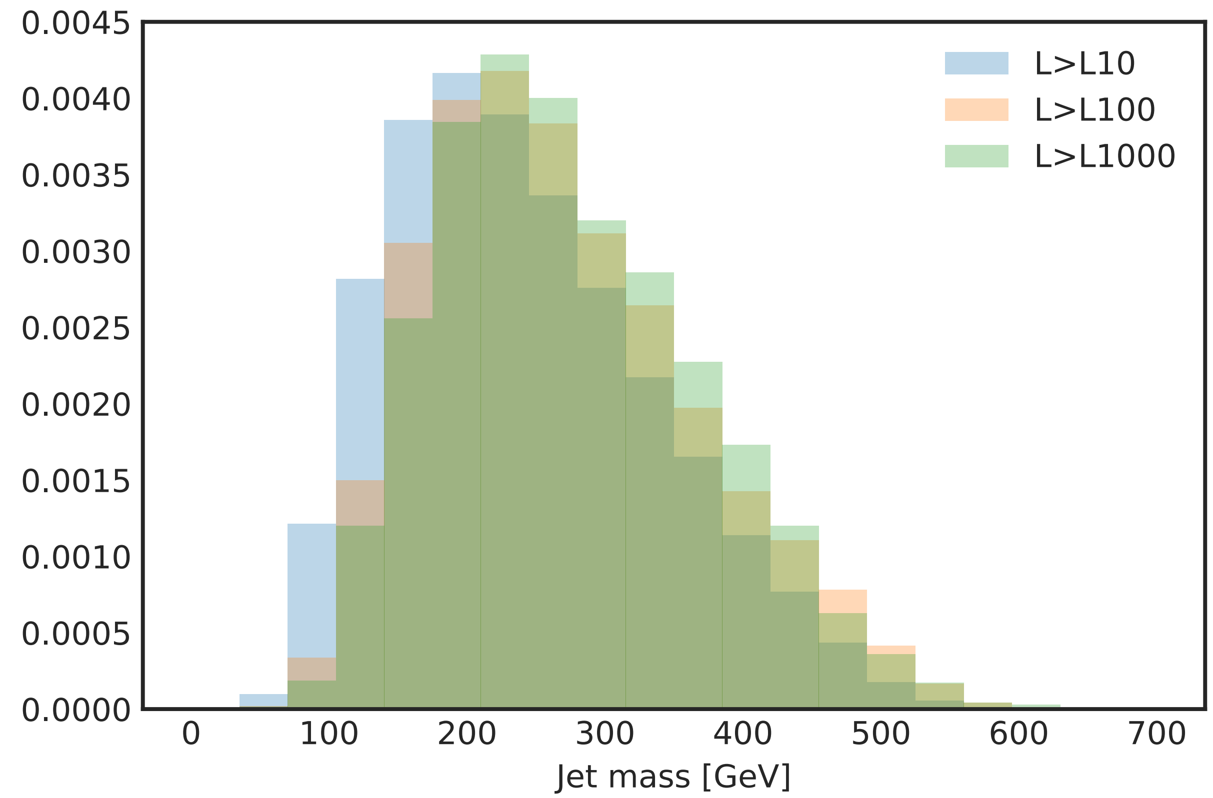
Only works if cutting on reconstruction error does not sculpt the mass distribution of the background!



This would greatly underestimate the background in the SR

Bump hunt with autoencoder





We found empirically that the background jet mass distribution is fairly stable against cuts on CNN AE reconstruction loss above ~250 GeV.

Autoencoder with explicit decorrelation

A more controlled approach to mass decorrelation would be to explicitly penalize correlations in the training of the autoencoder.

One promising method: autoencoder with adversarial decorrelation
(Heimel et al 1808.08979; based on 1611.01046, 1703.03507)

- Introduce a second NN, the adversary, that tries to predict the mass from the reconstruction loss.
- Penalize the total loss function when the adversary does well.

$$L_{adv} = \sum_i (f_{adv}(L_{AE}(x_i)) - m_i)^2$$

$$L_{tot} = L_{AE} - \lambda L_{adv}$$

See Tilman's talk for more on this (?)

Alternatives to adversaries

Adversaries are notoriously **tricky to train** — saddle point optimization

$$\min_{\theta_{\text{clf}}} \max_{\theta_{\text{adv}}} L_{\text{clf}}(y(\theta_{\text{clf}})) - \lambda L_{\text{adv}}(y(\theta_{\text{clf}}), m; \theta_{\text{adv}})$$

Would be great if we could achieve the same performance but with a convex regularizer term

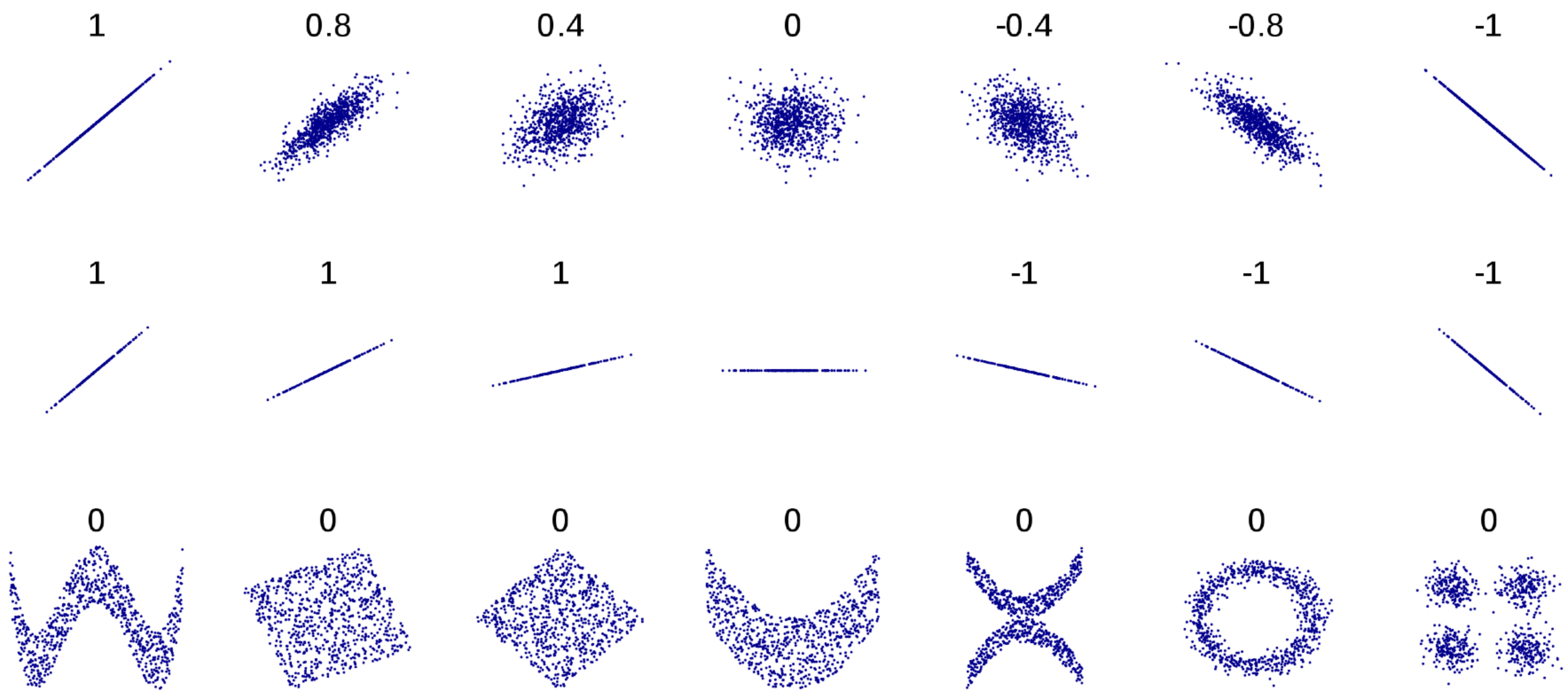
$$\min_{\theta_{\text{clf}}} L_{\text{clf}}(y(\theta_{\text{clf}})) + \lambda C_{\text{reg}}(y(\theta_{\text{clf}}), m)$$

First idea: can we just use Pearson correlation coefficient?

$$C_{\text{reg}} = R(y, m) \propto \sum_i y_i m_i$$

Problem: this only measures linear correlations

Pearson correlation



y and m can be highly correlated yet R=0

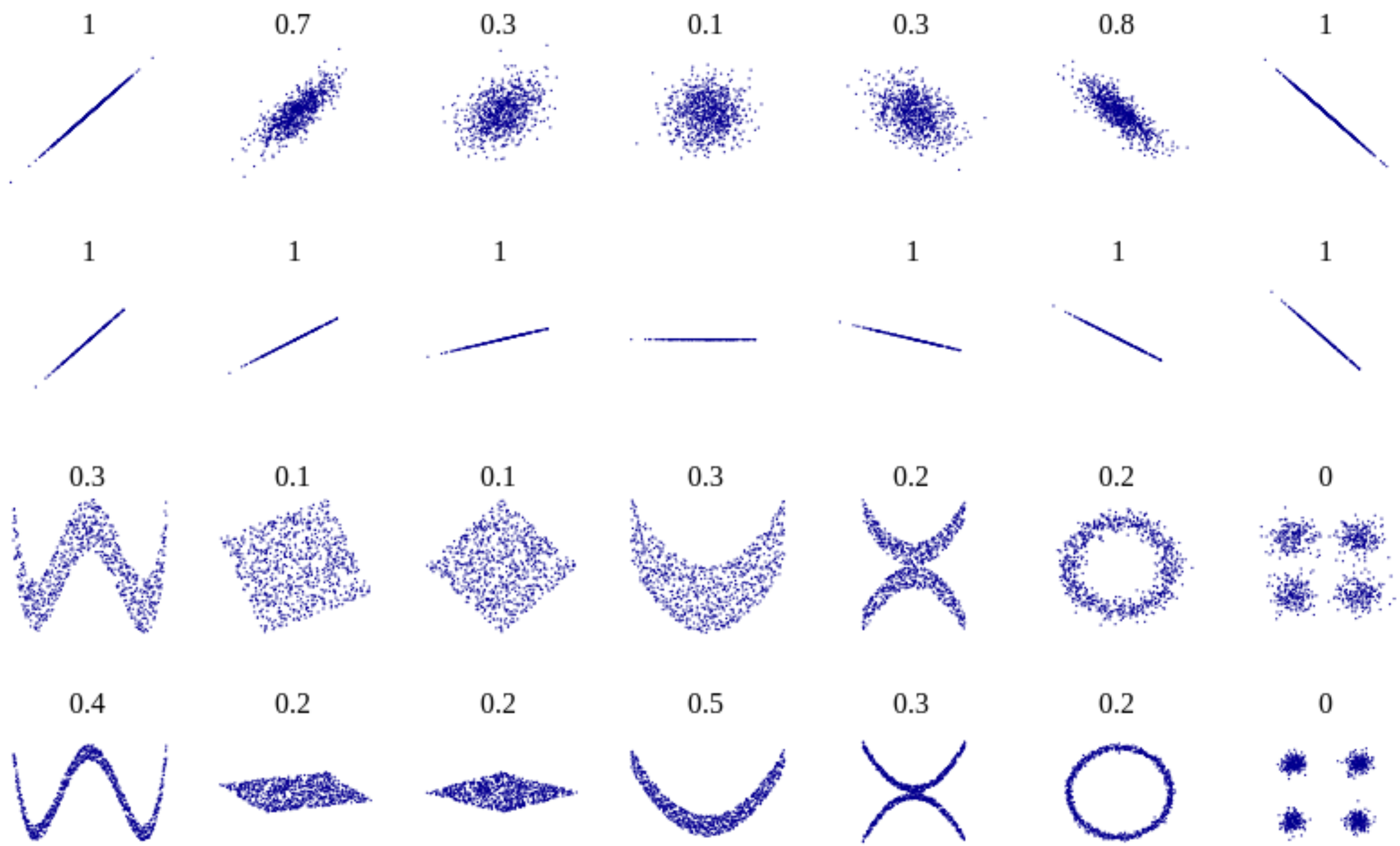
Distance correlation ("DisCo")

Promising idea: "distance correlation"
(Szekely, Rizzo, Bakirov 2007; Szekely & Rizzo 2009)

$$\mathrm{dCov}^2(X,Y) = \langle|X-X'||Y-Y'|\rangle + \langle|X-X'|\rangle\langle|Y-Y'|\rangle - 2\langle|X-X'||Y-Y''|\rangle$$

- Zero iff X,Y are independent; positive otherwise
- Computationally tractable
- Straightforward sample definition — doesn't require binning

Distance correlation ("DisCo")



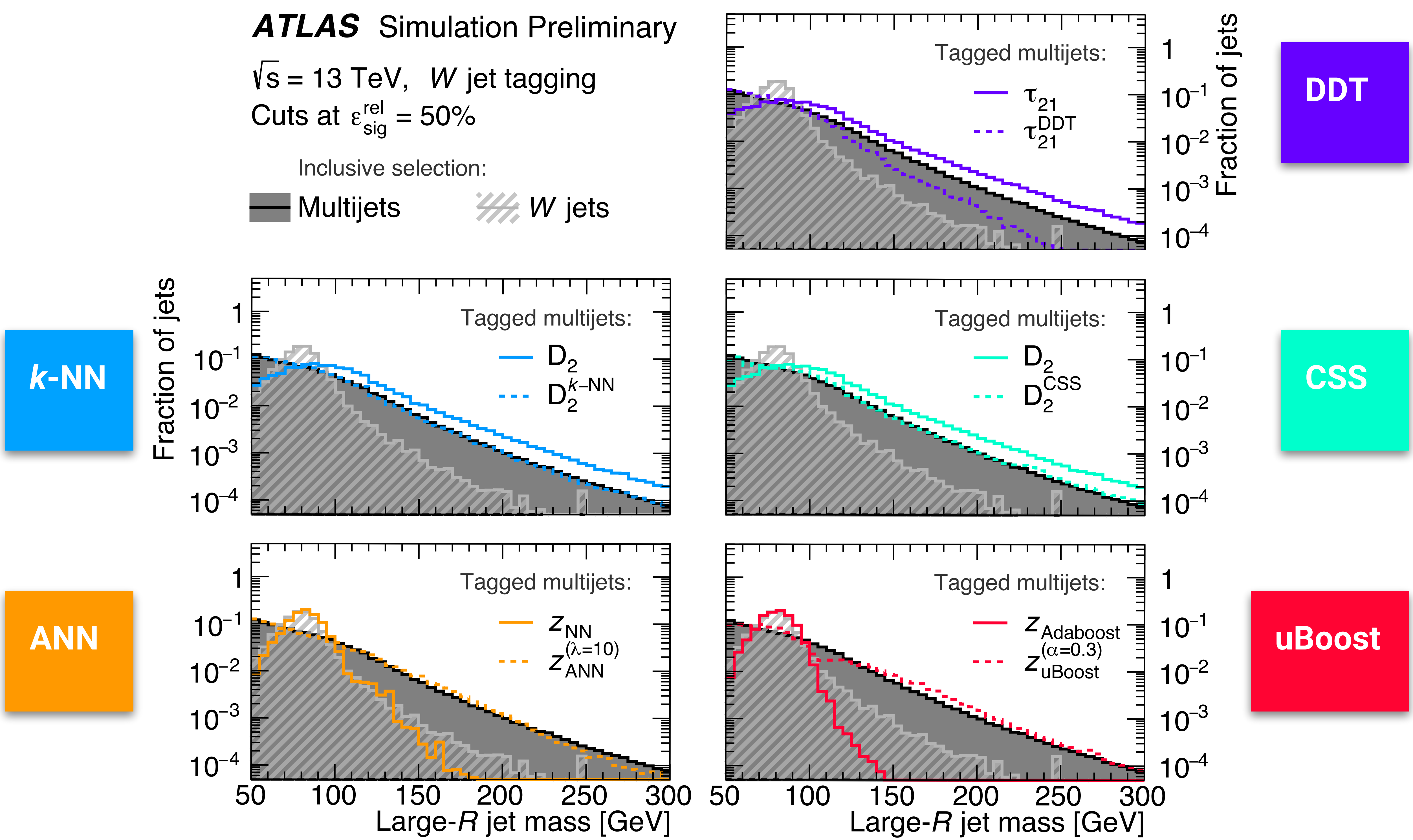
Disco is sensitive to nonlinear correlations!

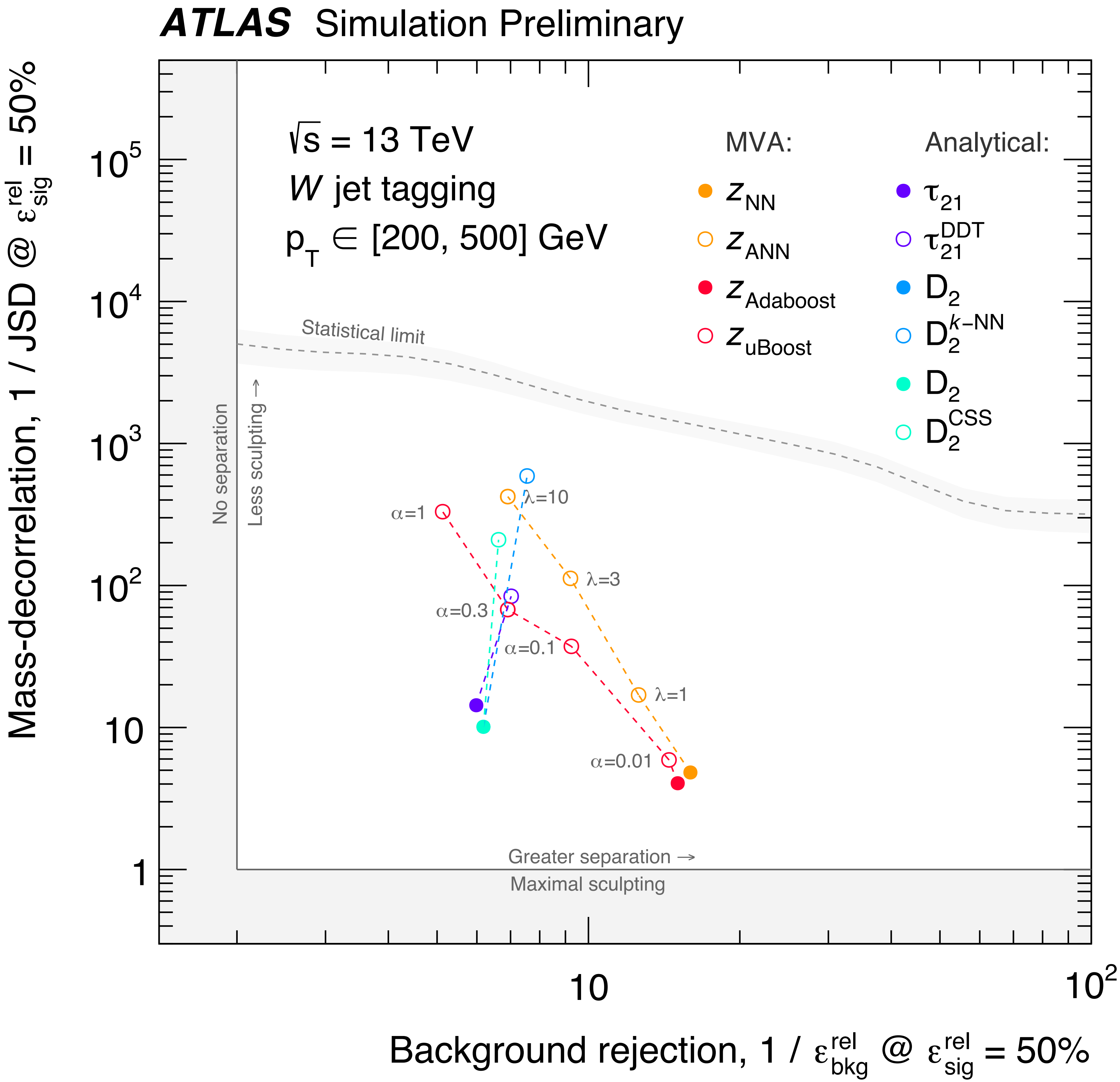
State of the art: ATLAS study of various decorrelation methods in context of boosted W-tagging.

ATL-PHYS-PUB-2018-014

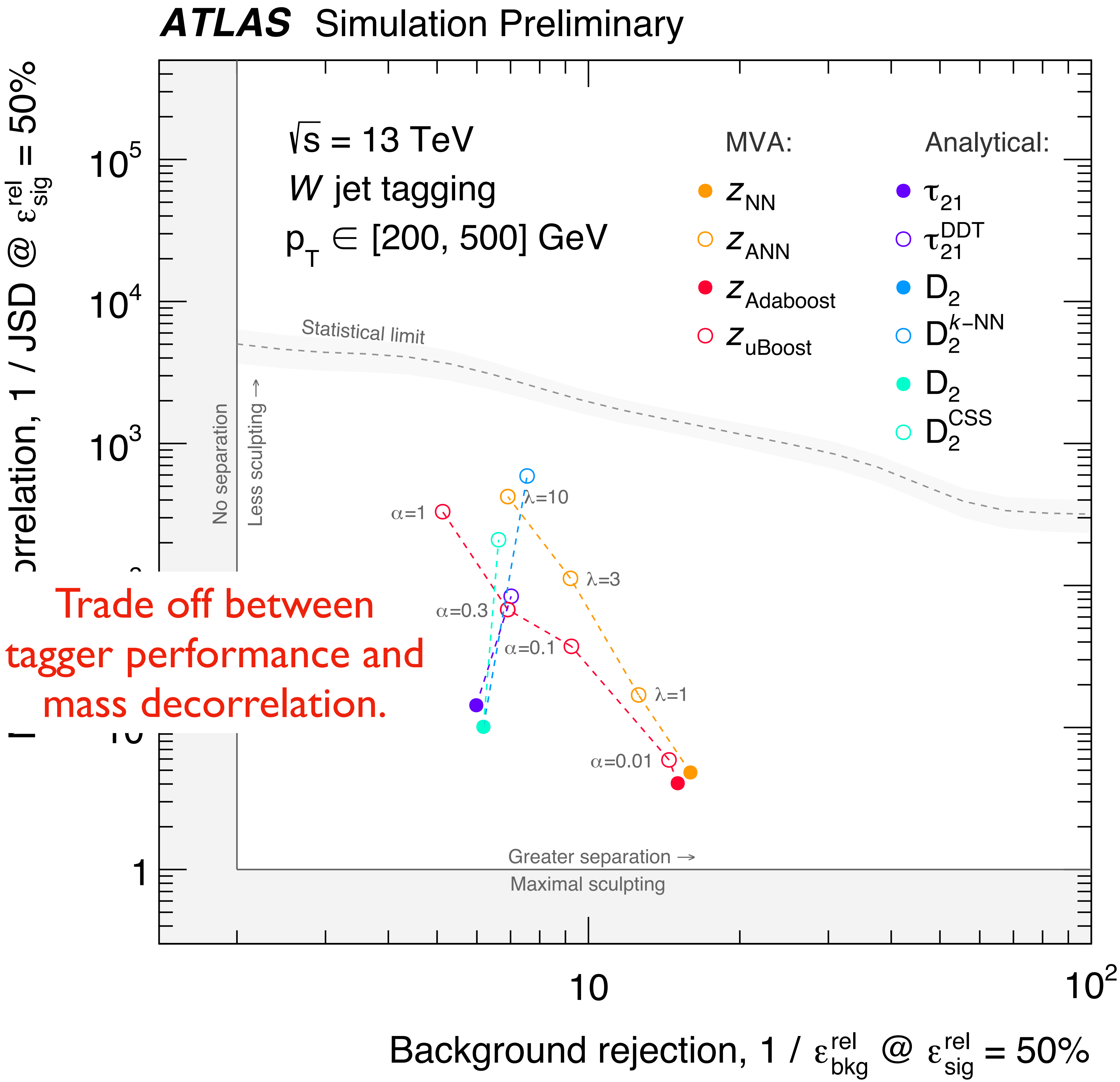
Analytical / single-variable

Multivariate (MVA)

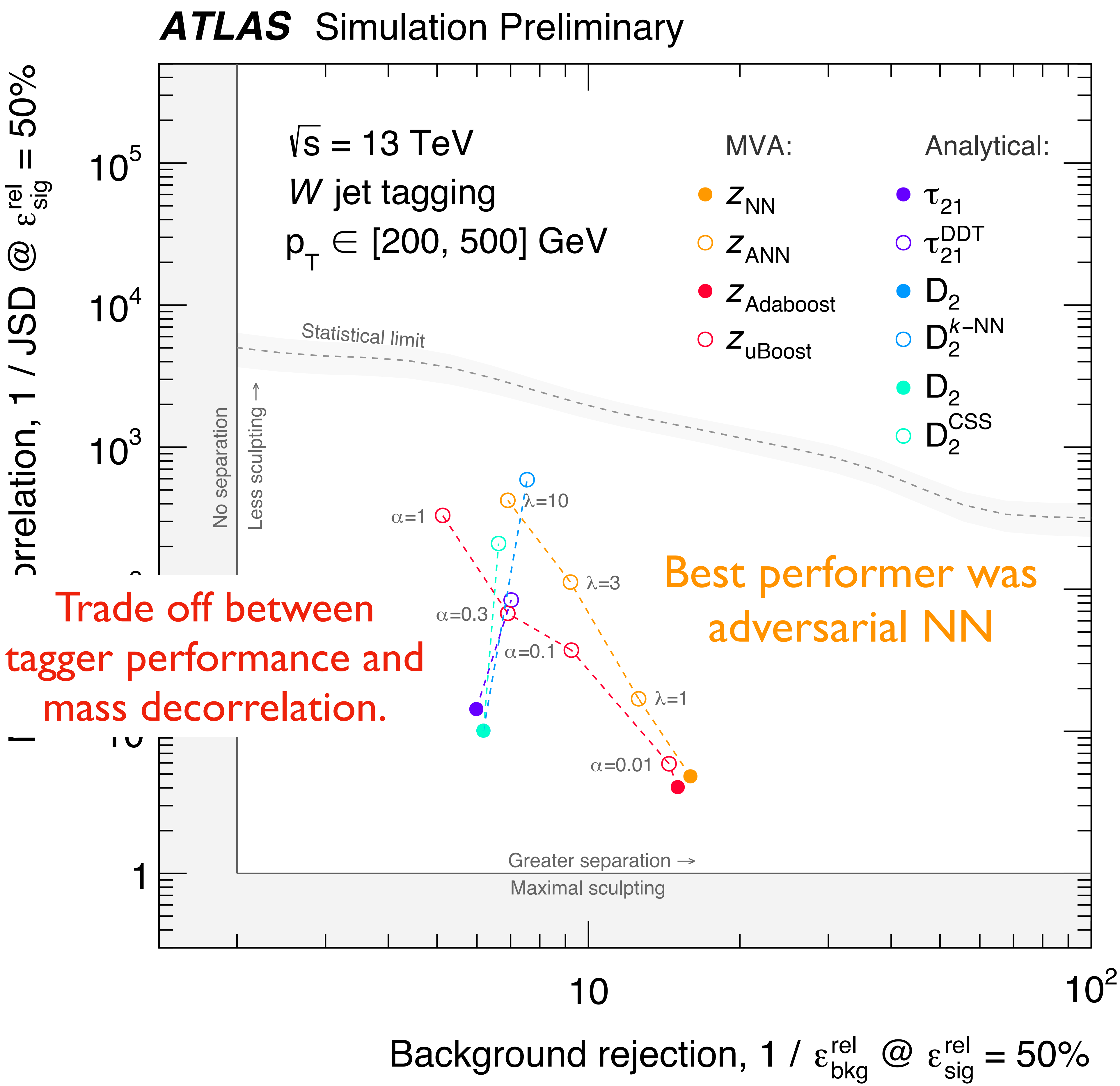




ATLAS Simulation Preliminary
√s = 13 TeV
W jet tagging
p_T ∈ [200, 500] GeV
MVA:
z_NN
z_ANN
z_Adaboost
z_uBoost
Analytical:
τ_21
τ_21^DDT
D_2
D_2^k−NN
D_2
D_2^CSS
Statistical limit
No separation
Less sculpting →
Greater separation →
Maximal sculpting
α=1
α=0.3
α=0.1
α=0.01
λ=10
λ=3
λ=1
Mass-decorrelation, 1 / JSD @ ε_sig^rel = 50%
Background rejection, 1 / ε_bkg^rel @ ε_sig^rel = 50%
1
10
10^2
10^3
10^4
10^5



Trade off between tagger performance and mass decorrelation.

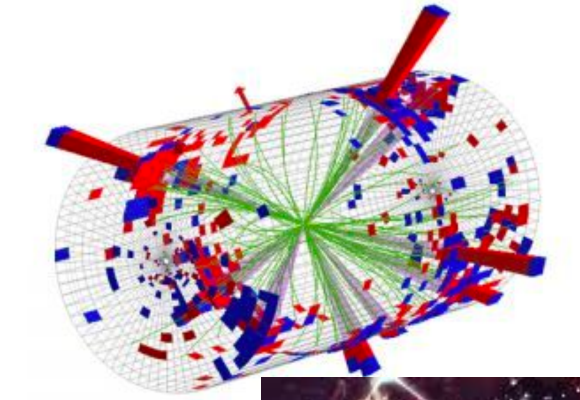


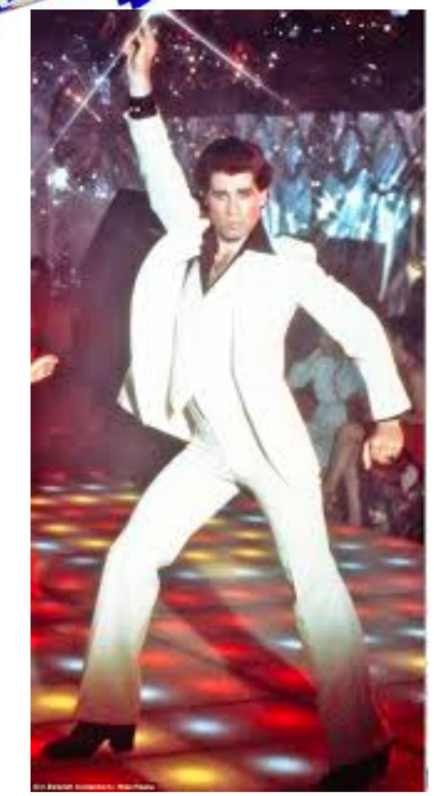
Trade off between tagger performance and mass decorrelation.

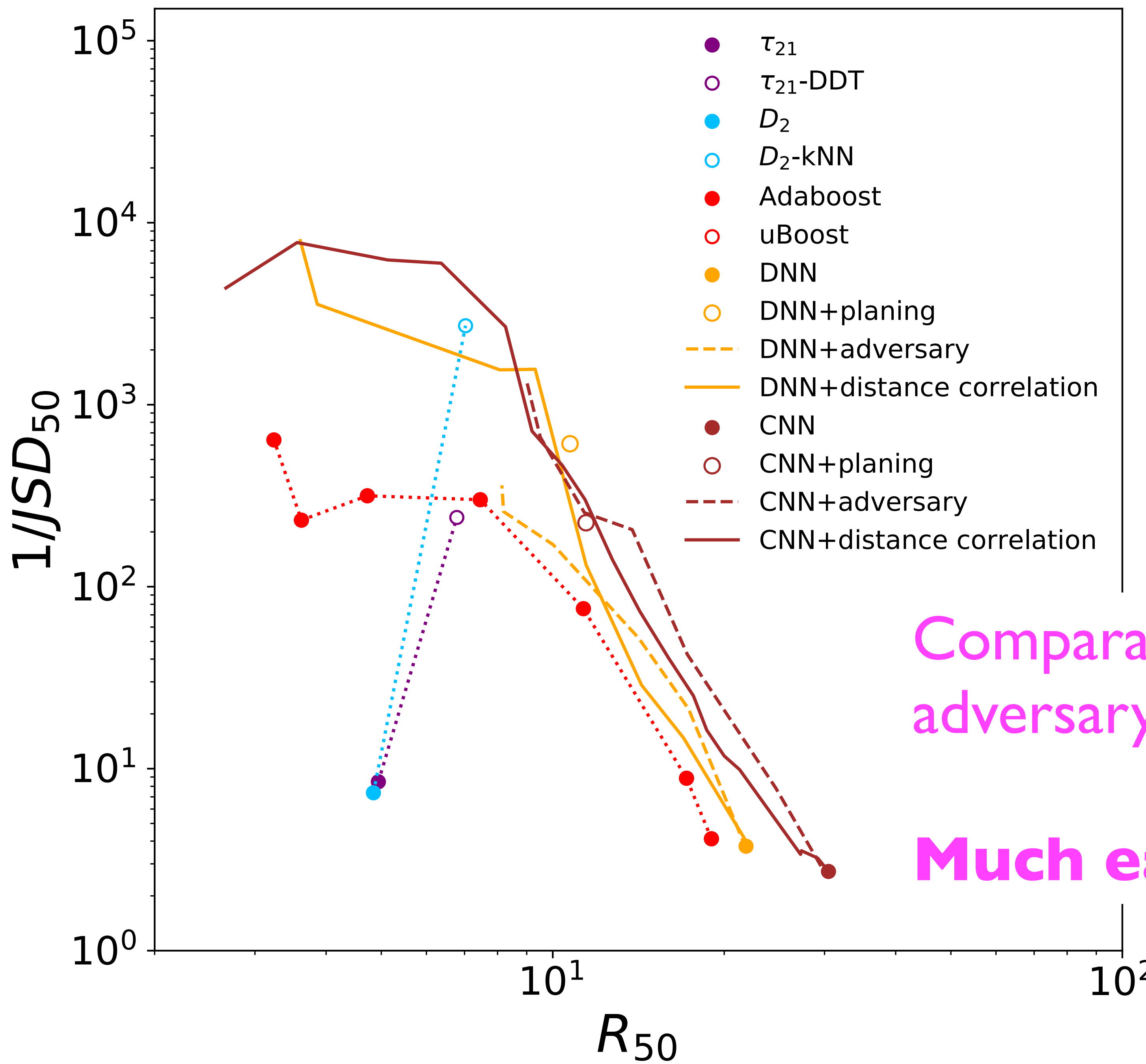
Best performer was adversarial NN

DisCo decorrelation

Gregor Kasieczka & DS 2001.05310







Comparable performance to adversary.

Much easier to train.

Enhancing the bump hunt

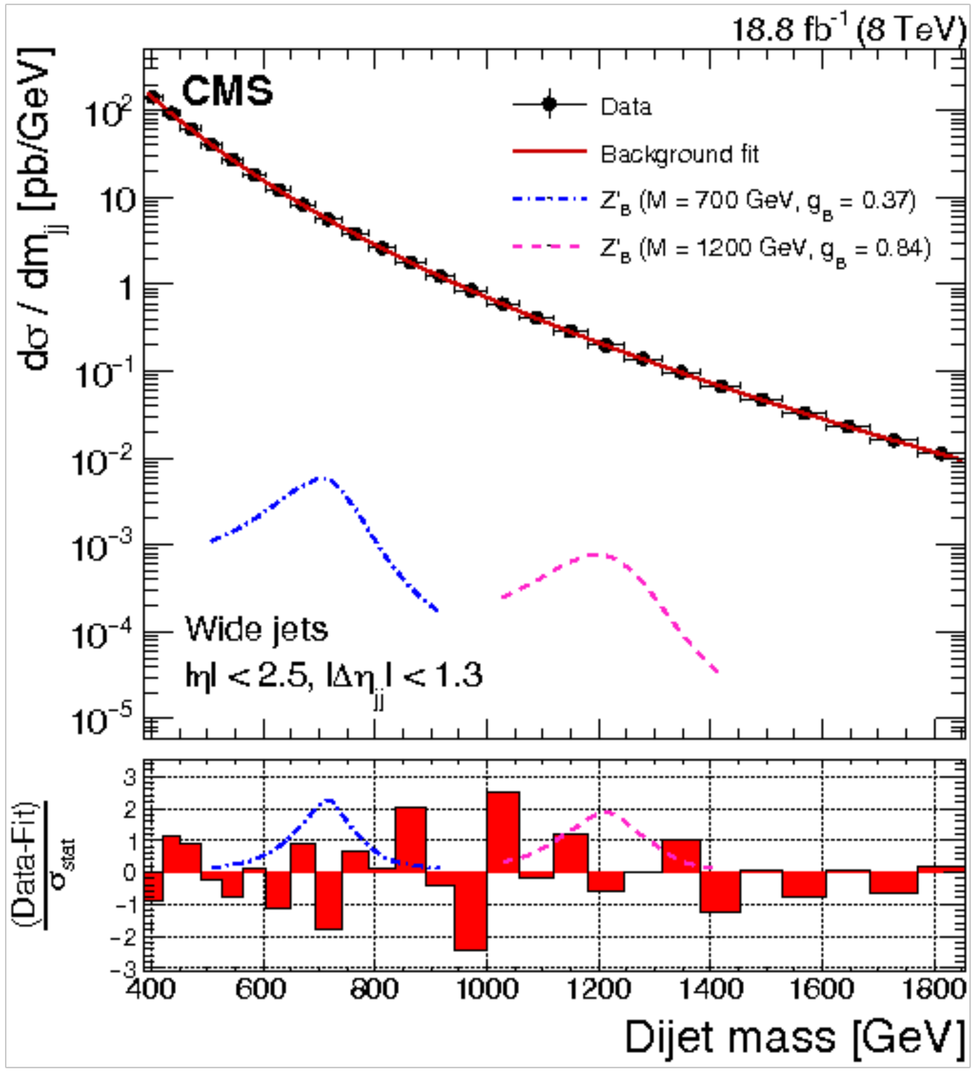
We have seen that one way to turn an autoencoder into an actual NP search is to combine it with a bump hunt.

However, what the autoencoder finds is rather uncontrolled and there is no guarantee of optimality (even asymptotically).

Can we get more if we build in the bump hunt assumption from the outset?

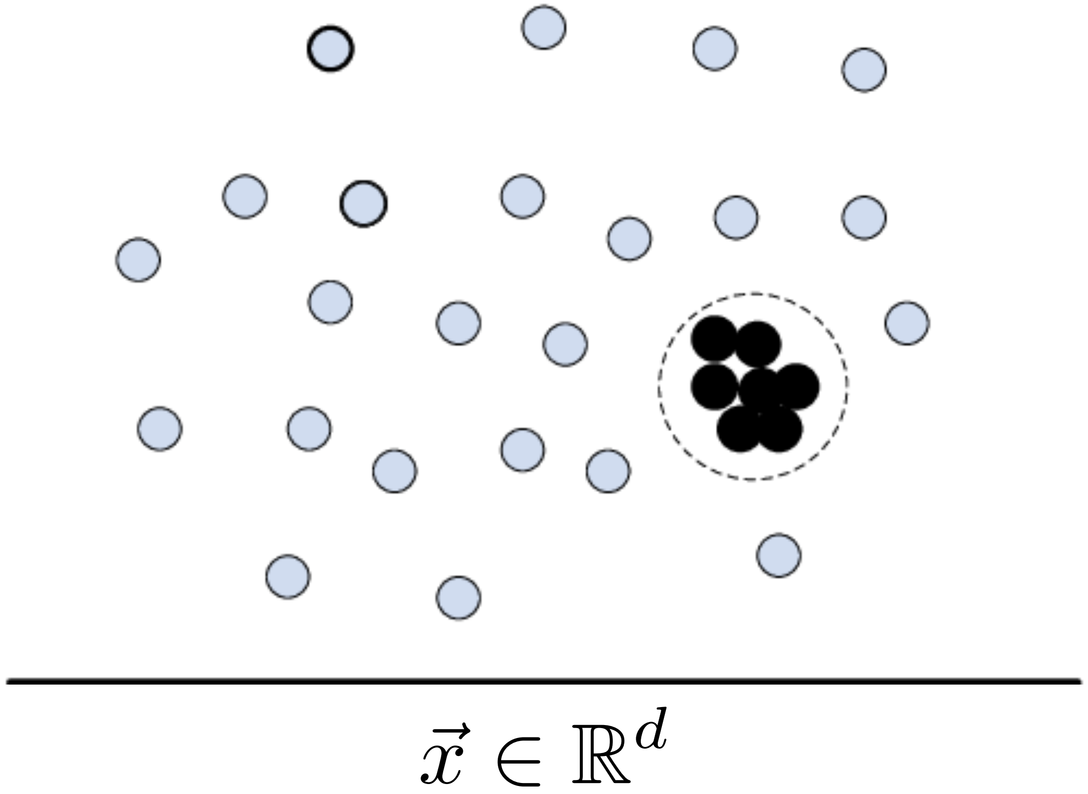
Enhancing the bump hunt

A growing number of methods aim to enhance the bump hunt using additional features:



primary resonant feature (m)

$\times$

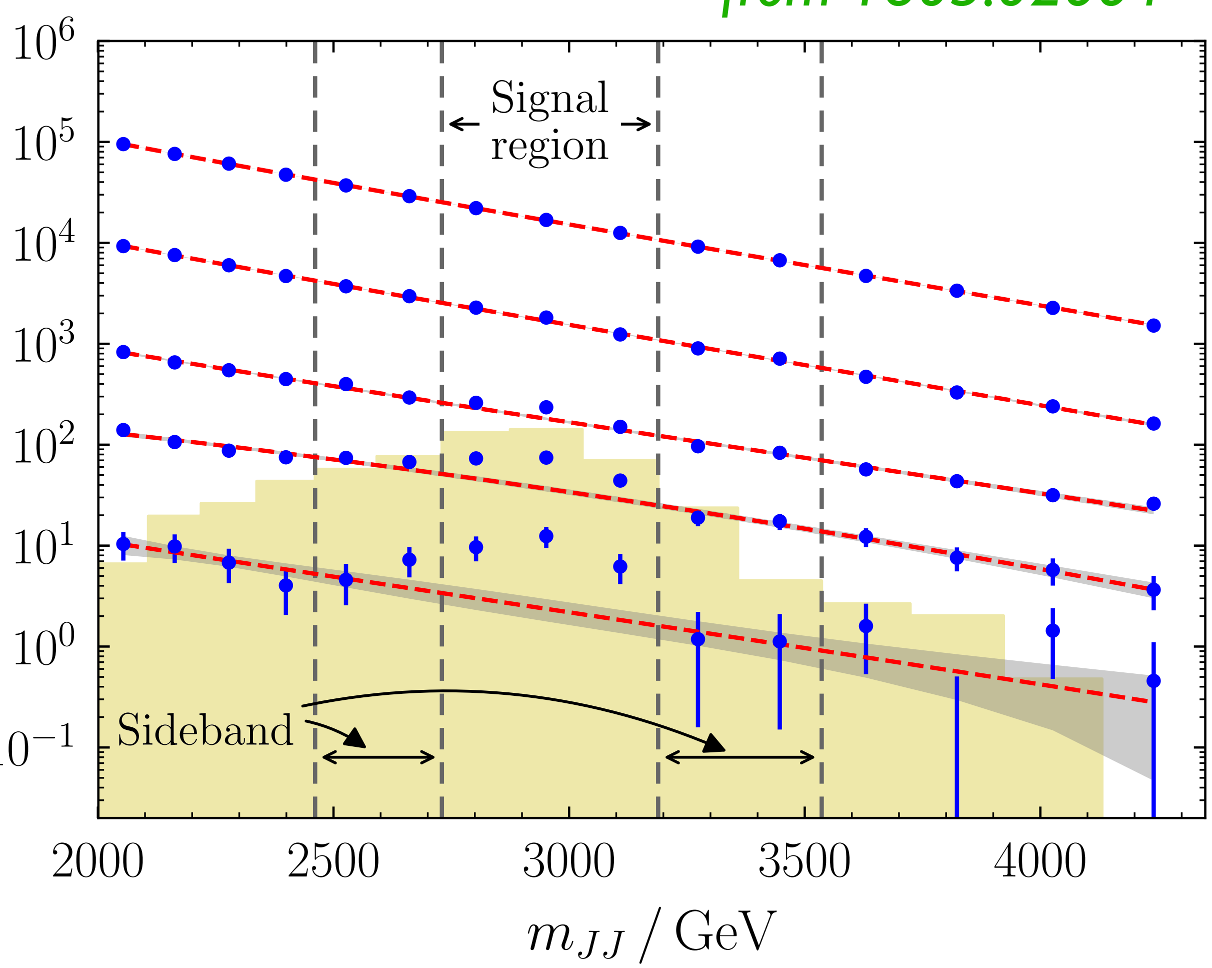


some additional features

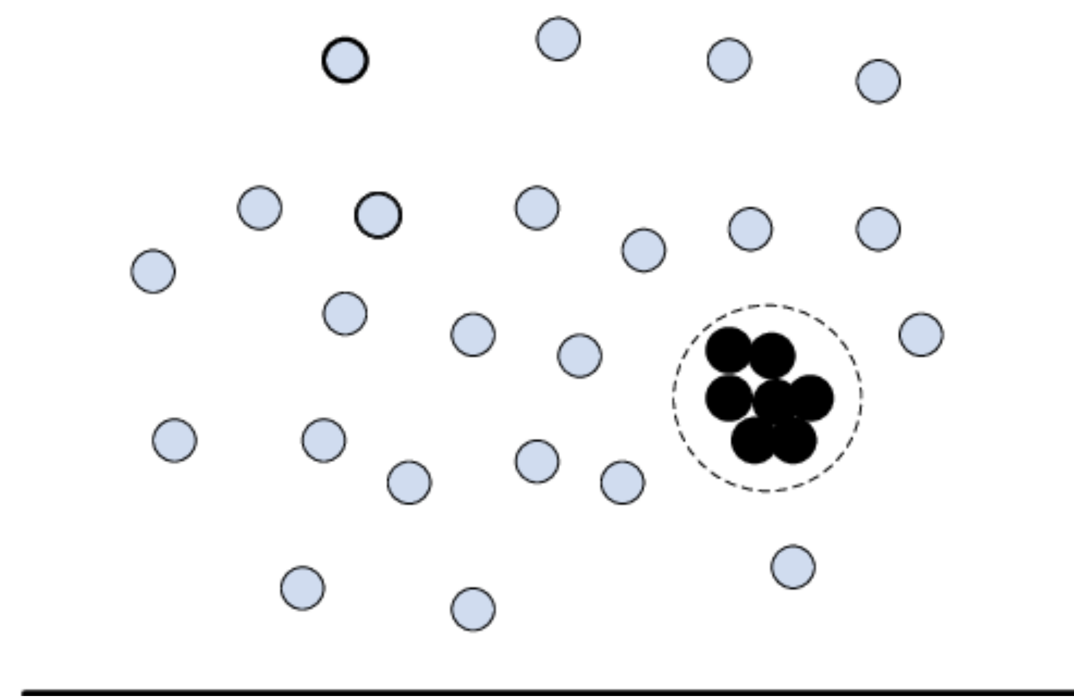
Enhancing the bump hunt

A growing number of methods aim to enhance the bump hunt using additional features:

from 1805.02664



$\times$



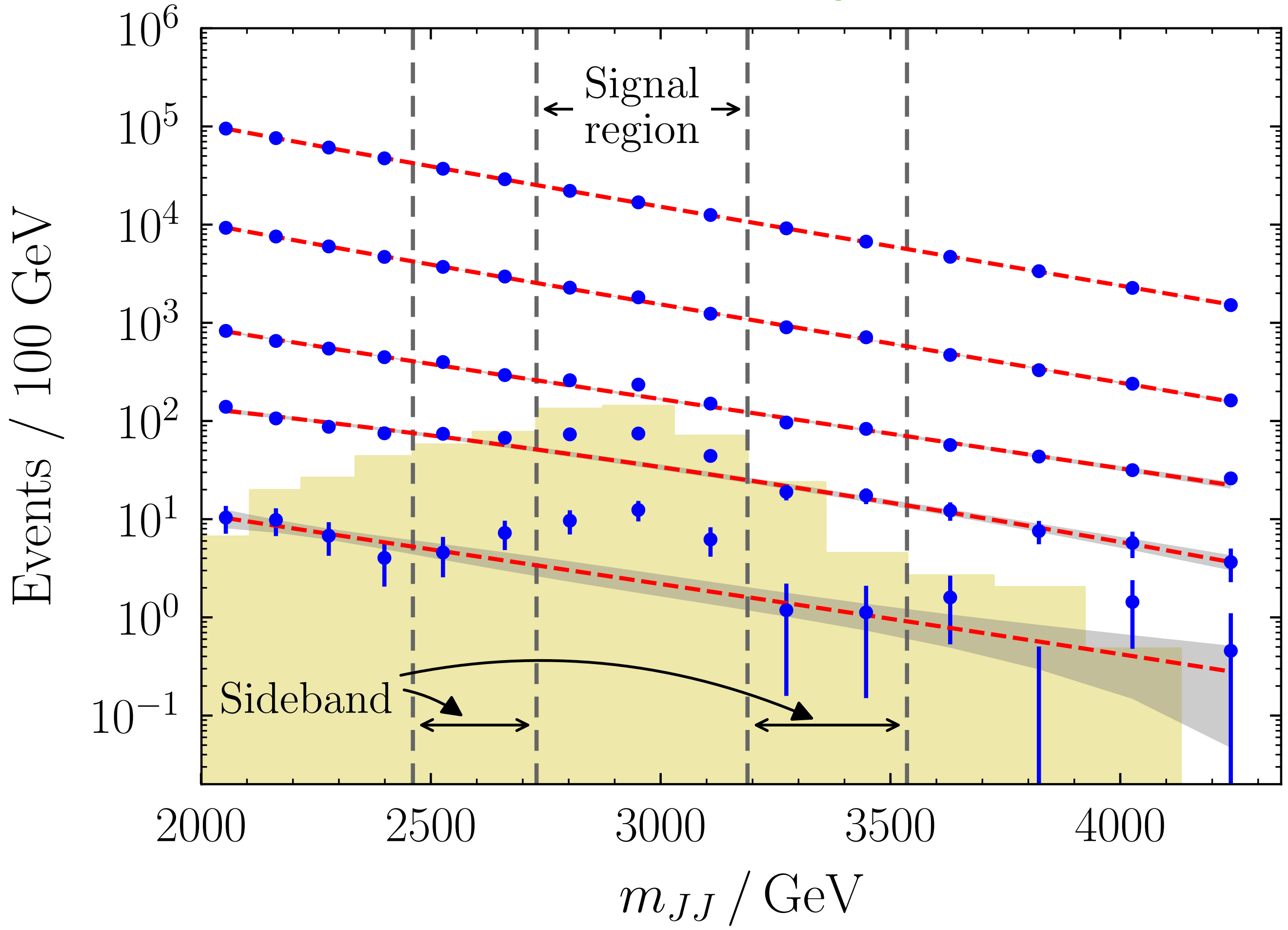
$\vec{x} \in \mathbb{R}^d$

primary resonant feature (m)

some additional features

Enhancing the bump hunt

from 1805.02664



Use deep learning to derive something approaching the **multidimensional** likelihood ratio, *directly from the data*

$$R(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \in SR)}$$

Cut on $R > R_c$, enhance the bump hunt!

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

1. CWoLa Hunting

Collins, Howe & Nachman 1805.02664, 1902.02634

Assume $\vec{x}$ and m are completely independent

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

1. CWoLa Hunting

Collins, Howe & Nachman 1805.02664, 1902.02634

Assume $\vec{x}$ and m are completely independent

Train a classifier on $\vec{x}$ to distinguish *signal region* and *sideband*. If the classifier is near-optimal, it will approach likelihood ratio

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

1. <u>CWoLa Hunting</u>

Collins, Howe & Nachman 1805.02664, 1902.02634

Assume $\vec{x}$ and m are completely independent

Train a classifier on $\vec{x}$ to distinguish *signal region* and *sideband*. If the classifier is near-optimal, it will approach likelihood ratio

$$R_{CWoLa}(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|data; m \notin SR)}$$

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

1. CWoLa Hunting

Collins, Howe & Nachman 1805.02664, 1902.02634

Assume $\vec{x}$ and m are completely independent

Train a classifier on $\vec{x}$ to distinguish *signal region* and *sideband*. If the classifier is near-optimal, it will approach likelihood ratio

$$R_{CWoLa}(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|data; m \notin SR)}$$

$$= \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \notin SR)}$$

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

1. <u>CWoLa Hunting</u>

Collins, Howe & Nachman 1805.02664, 1902.02634

Assume $\vec{x}$ and m are completely independent

Train a classifier on $\vec{x}$ to distinguish *signal region* and *sideband*. If the classifier is near-optimal, it will approach likelihood ratio

$$R_{CWoLa}(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|data; m \notin SR)}$$

$$= \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \notin SR)}$$

$$= \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \in SR)}$$

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

1. CWoLa Hunting

Collins, Howe & Nachman 1805.02664, 1902.02634

Assume $\vec{x}$ and m are completely independent

Train a classifier on $\vec{x}$ to distinguish *signal region* and *sideband*. If the classifier is near-optimal, it will approach likelihood ratio

$$R_{CWoLa}(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|data; m \notin SR)}$$

$$= \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \notin SR)}$$

"Classification Without Labels"

$$= \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \in SR)}$$

Enhancing the bump hunt

3 new ideas

2. AnoDE: Anomaly Detection with Density Estimation

Nachman & DS 2001.04990

Directly learn the conditional probability densities from the data

$$\mathcal{L}(\vec{x}|data; m \in SR) \qquad \mathcal{L}(\vec{x}|data; m \notin SR) = \mathcal{L}(\vec{x}|B_{data}; m \notin SR)$$

Enhancing the bump hunt
3 new ideas

2. AnoDE: Anomaly Detection with Density Estimation

Nachman & DS 2001.04990

Directly learn the conditional probability densities from the data

$$\mathcal{L}(\vec{x}|data; m \in SR) \qquad \mathcal{L}(\vec{x}|data; m \notin SR) = \mathcal{L}(\vec{x}|B_{data}; m \notin SR)$$

interpolate in (x,m)

Enhancing the bump hunt

3 new ideas

2. <u>AnoDE: Anomaly Detection with Density Estimation</u>

Nachman & DS 2001.04990

Directly learn the conditional probability densities from the data

$$\mathcal{L}(\vec{x}|data; m \in SR) \qquad \mathcal{L}(\vec{x}|data; m \notin SR) = \mathcal{L}(\vec{x}|B_{data}; m \notin SR)$$

interpolate in (x,m)

$$\mathcal{L}(\vec{x}|B_{data}; m \in SR)$$

Enhancing the bump hunt

3 new ideas

2. AnoDE: Anomaly Detection with Density Estimation

Nachman & DS 2001.04990

Directly learn the conditional probability densities from the data

$$\mathcal{L}(\vec{x}|data; m \in SR) \qquad \mathcal{L}(\vec{x}|data; m \notin SR) = \mathcal{L}(\vec{x}|B_{data}; m \notin SR)$$

interpolate in (x,m)

$$\mathcal{L}(\vec{x}|B_{data}; m \in SR)$$

Construct the likelihood ratio: $R(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \in SR)}$

Enhancing the bump hunt

3 new ideas

2. AnoDE: Anomaly Detection with Density Estimation

Nachman & DS 2001.04990

Directly learn the conditional probability densities from the data

$$\mathcal{L}(\vec{x}|data; m \in SR) \qquad \mathcal{L}(\vec{x}|data; m \notin SR) = \mathcal{L}(\vec{x}|B_{data}; m \notin SR)$$

interpolate in (x,m)

$$\mathcal{L}(\vec{x}|B_{data}; m \in SR)$$

Construct the likelihood ratio: $$R(\vec{x}) = \frac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \in SR)}$$

Recent breakthroughs in (neural) density estimation make this possible in high dimensions. We used conditional MAF (Papamakarios et al 1705.07057) but many other density estimators are possible

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

3. <u>SALAD: Simulation Assisted Likelihood-free Anomaly Detection</u>

Andreassen, Nachman & DS 2001.05001

Simulations in HEP are very good but not good enough to directly compare against data. If we could reweight simulations to match data they would be much more useful!

- Train a classifier on data vs simulation in sidebands. If this classifier is near-optimal, it will approach the likelihood ratio (1907.08209)

$$w(\vec{x}) = \frac{\mathcal{L}(\vec{x}|B_{data}, m \notin SR)}{\mathcal{L}(\vec{x}|B_{sim}, m \notin SR)}$$

- Interpolate into SR
- Using reweighted simulation, generate a sample that follows $\mathcal{L}(\vec{x}|B_{data}, m \in SR)$
- Train a classifier to distinguish data from this sample
- Obtain a discriminant that approaches

See Ben's talk for more on this

Enhancing the bump hunt

3 new ideas

3. SALAD: Simulation Assisted Likelihood-free Anomaly Detection
Andreassen, Nachman & DS 2001.05001

Simulations in HEP are very good but not good enough to directly compare against data. If we could reweight simulations to match data they would be much more useful!

- Train a classifier on data vs simulation in sidebands. If this classifier is near-optimal, it will approach the likelihood ratio (1907.08209)

$$w(\vec{x}) = \frac{\mathcal{L}(\vec{x}|B_{data}, m \notin SR)}{\mathcal{L}(\vec{x}|B_{sim}, m \notin SR)}$$

- Interpolate into SR
- Using reweighted simulation, generate a sample that follows $\mathcal{L}(\vec{x}|B_{data}, m \in SR)$
- Train a classifier to distinguish data from this sample
- Obtain a discriminant that approaches $R(\vec{x}) = \dfrac{\mathcal{L}(\vec{x}|data; m \in SR)}{\mathcal{L}(\vec{x}|B_{data}; m \in SR)}$

LHC Olympics 2020

To facilitate a meaningful comparison between different approaches and to spur the development of new ones, in April 2019, Gregor Kasieczka, Ben Nachman and I initiated an anomaly detection data challenge:

The LHC Olympics 2020

https://indico.cern.ch/event/809820/page/19002-lhcolympics2020

LHC Olympics 2020: Black Boxes

We prepared three black boxes of simulated data:

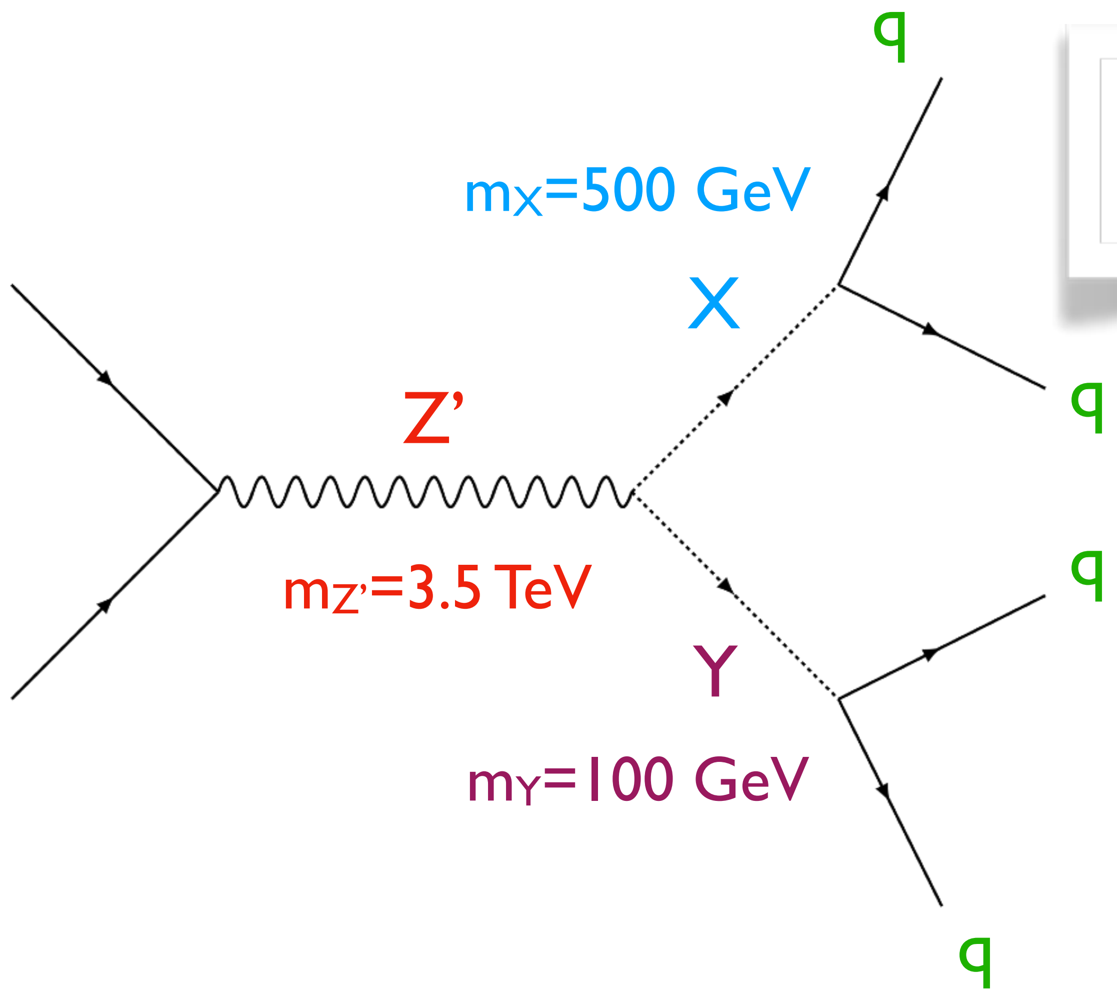
- 1 million events each
- 4-vectors of every reconstructed particle (hadron) in the event
- Particle ID, charge, etc not included
- Single R=1 jet trigger pT>1.2 TeV
- Black boxes are meant to be representative of actual data, meaning they are mostly background and may contain signals of new physics

In addition, a sample of 1M QCD dijet events (produced with Pythia8 and Delphes3.4.1) was provided as a background sample.

https://doi.org/10.5281/zenodo.3547721

LHC Olympics 2020: R&D Dataset

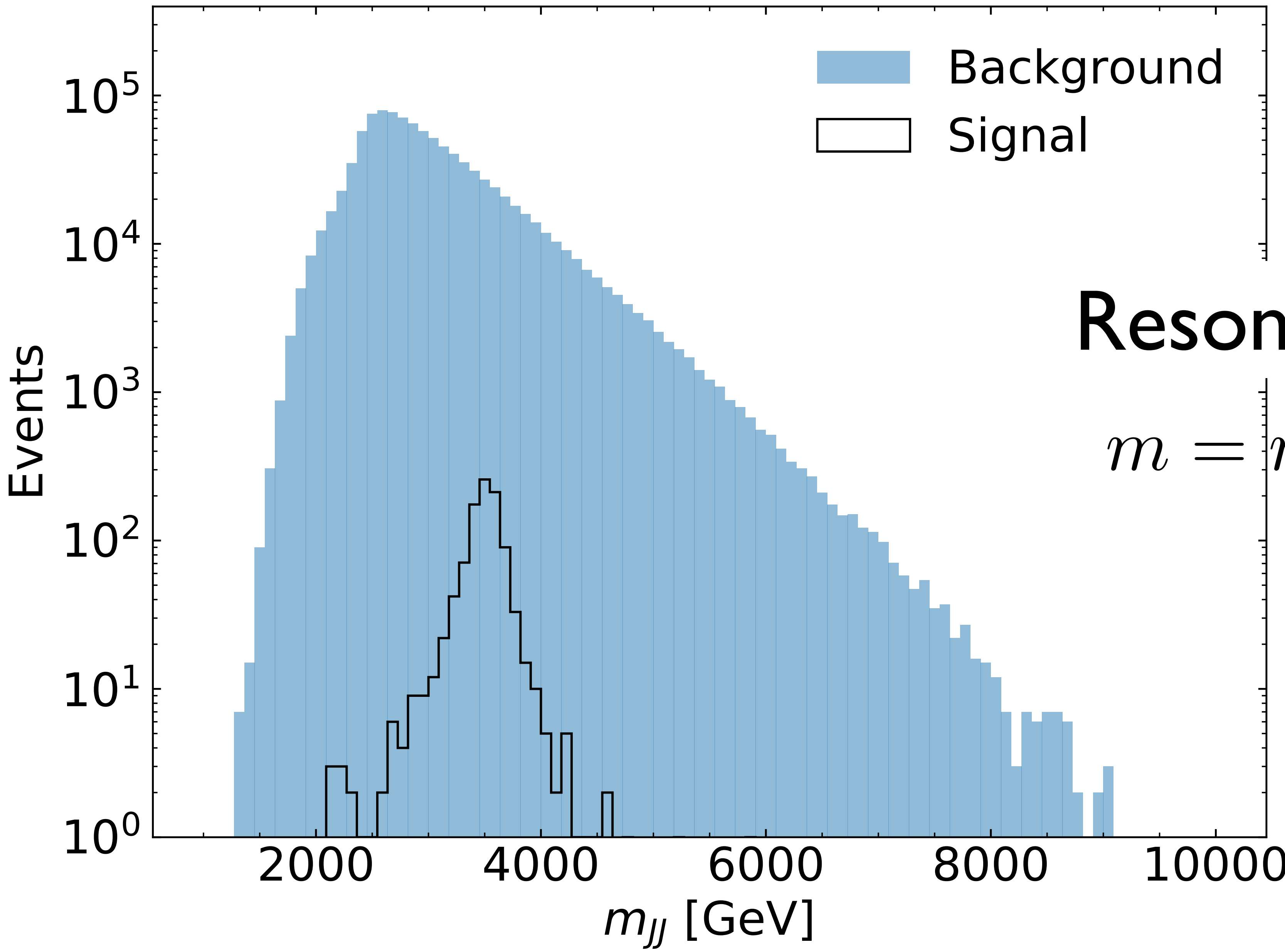
Prior to the challenge, we also released a labeled R&D dataset consisting of 1M QCD dijet events and 100k signal events



No explicit search at the LHC for this scenario!

https://doi.org/10.5281/zenodo.2629072

LHC Olympics 2020: R&D Dataset



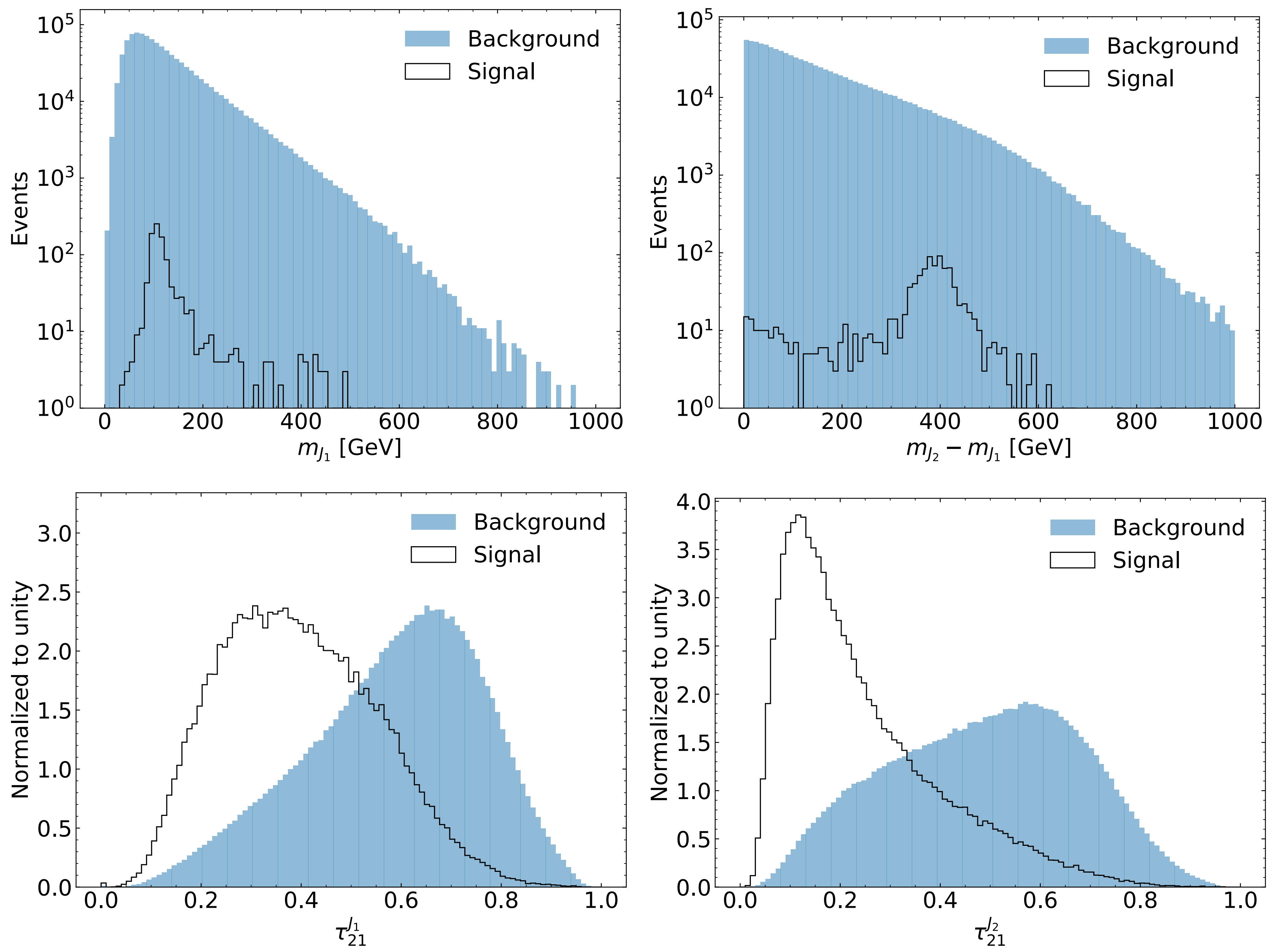
Resonant feature

$$m = m_{Z'} = m_{JJ}$$

S=500, B=500,000, B_{SR}=61,000

$S/B_{SR} \sim 6\times10^{-3}$, $S/\sqrt{B_{SR}} \sim 1.5$

LHC Olympics 2020: R&D Dataset



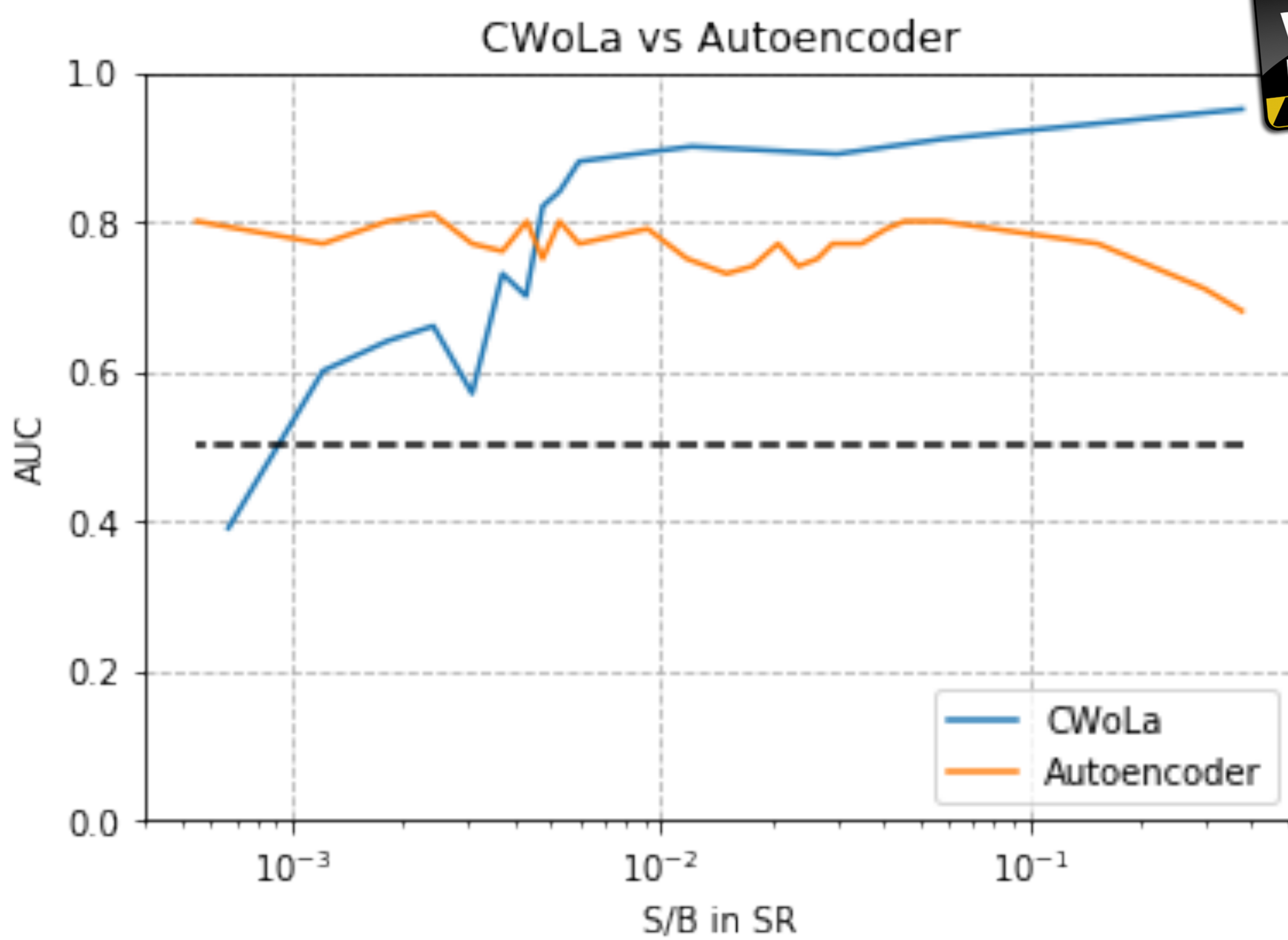
Additional features: $x = (m_{J_1}, m_{J_2}, \tau_{21}^{J_1}, \tau_{21}^{J_2})$

slide credit: Pablo Martin

Comparing CWoLa vs Autoencoders with LHCO R&D Dataset

Pablo Martin, Ben Nachman & DS work in progress

- Test performance of both methods for different S/B ratios



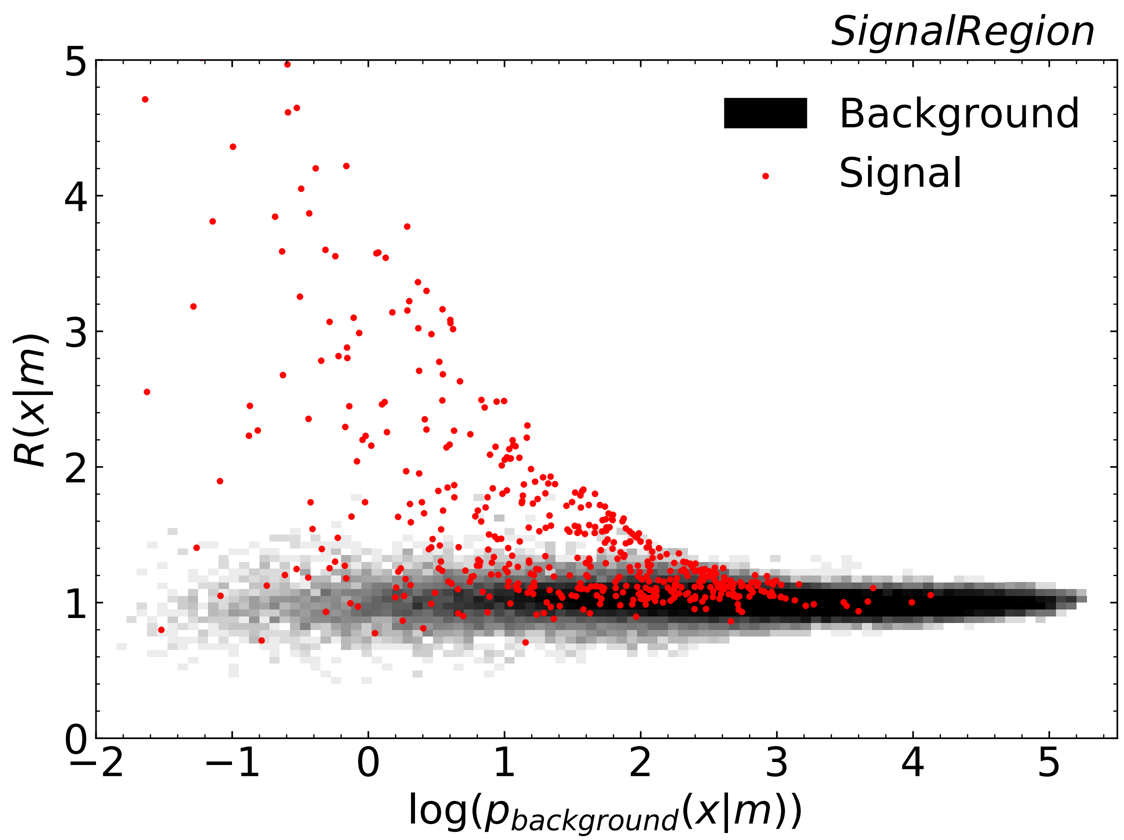
⇒ CWoLa performs better at large cross sections

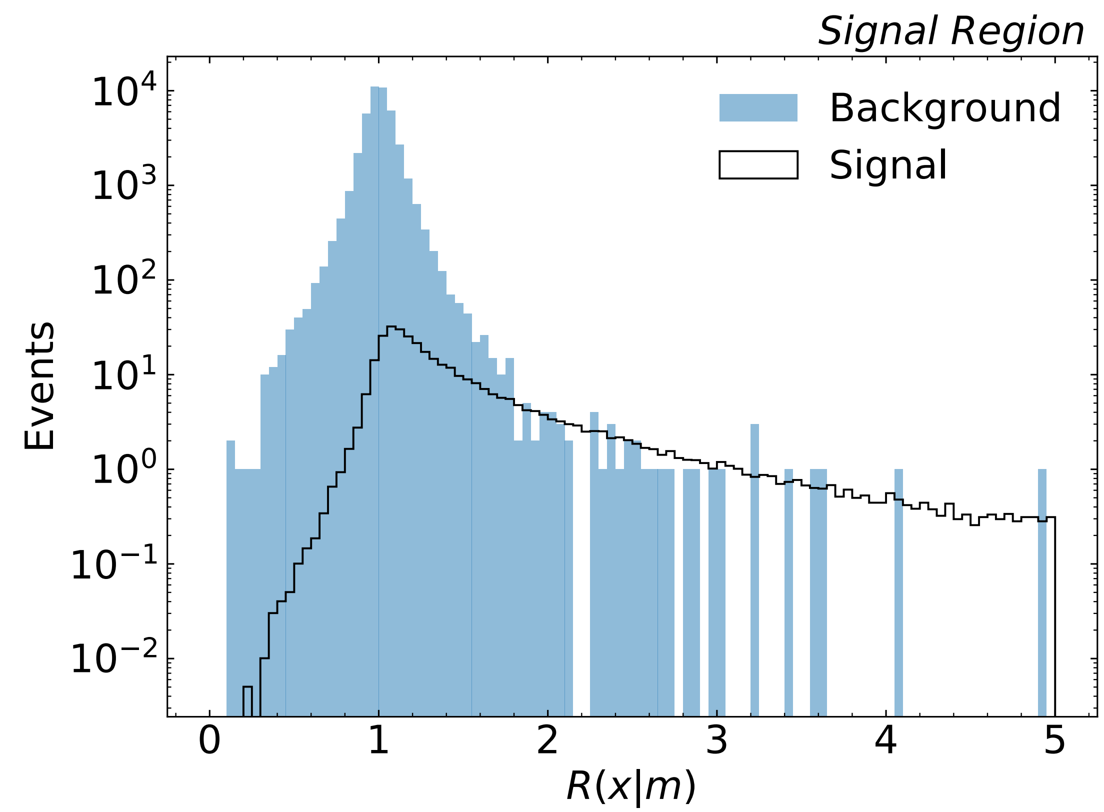
⇒ Autoencoder solid at very low cross sections

Complementary techniques!

ANODE: Results on LHCO R&D Dataset

Ben Nachman & DS 2001.04990

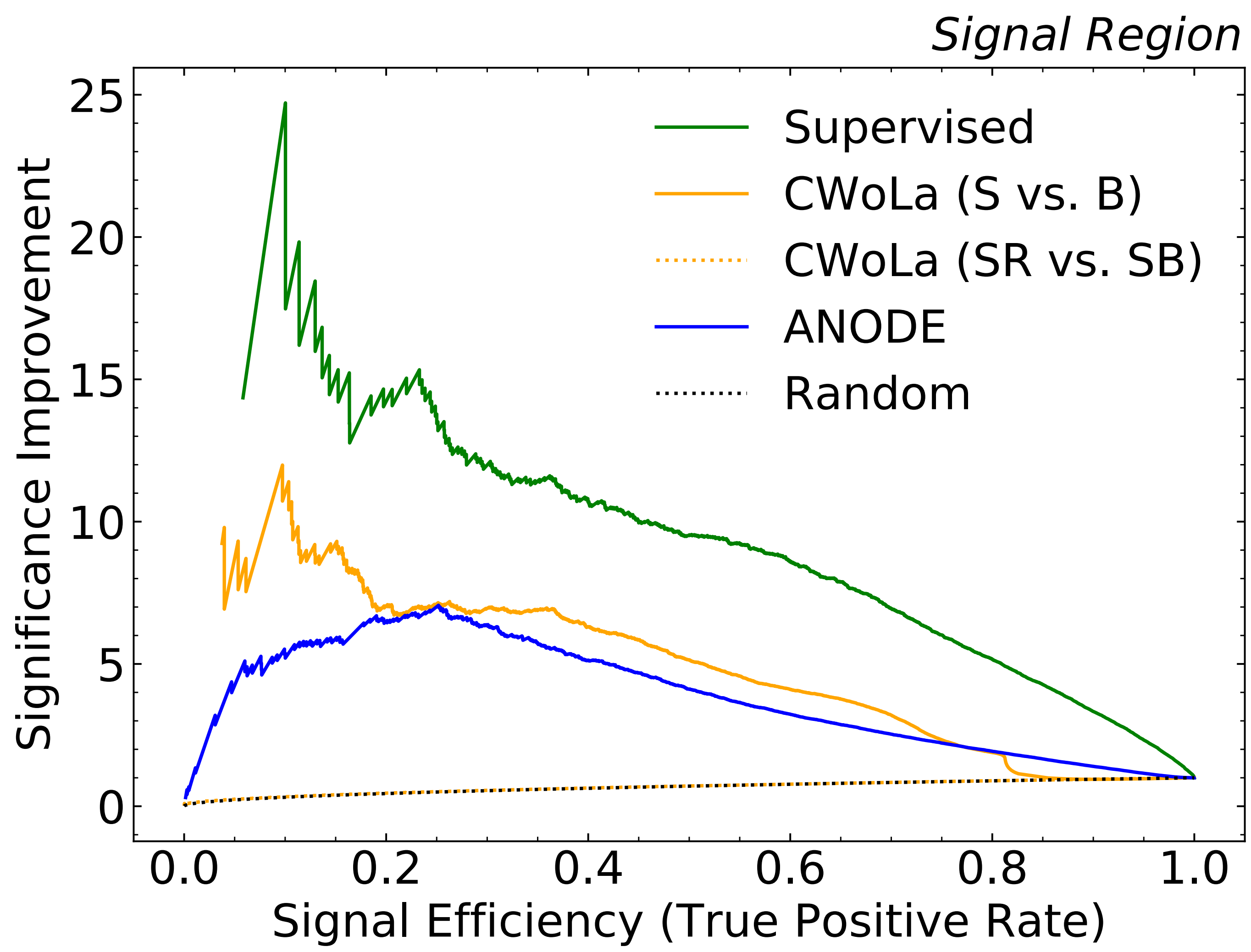




The method works! ANODE is sensitive to the signal!

ANODE: Results on LHCO R&D Dataset

Ben Nachman & DS 2001.04990



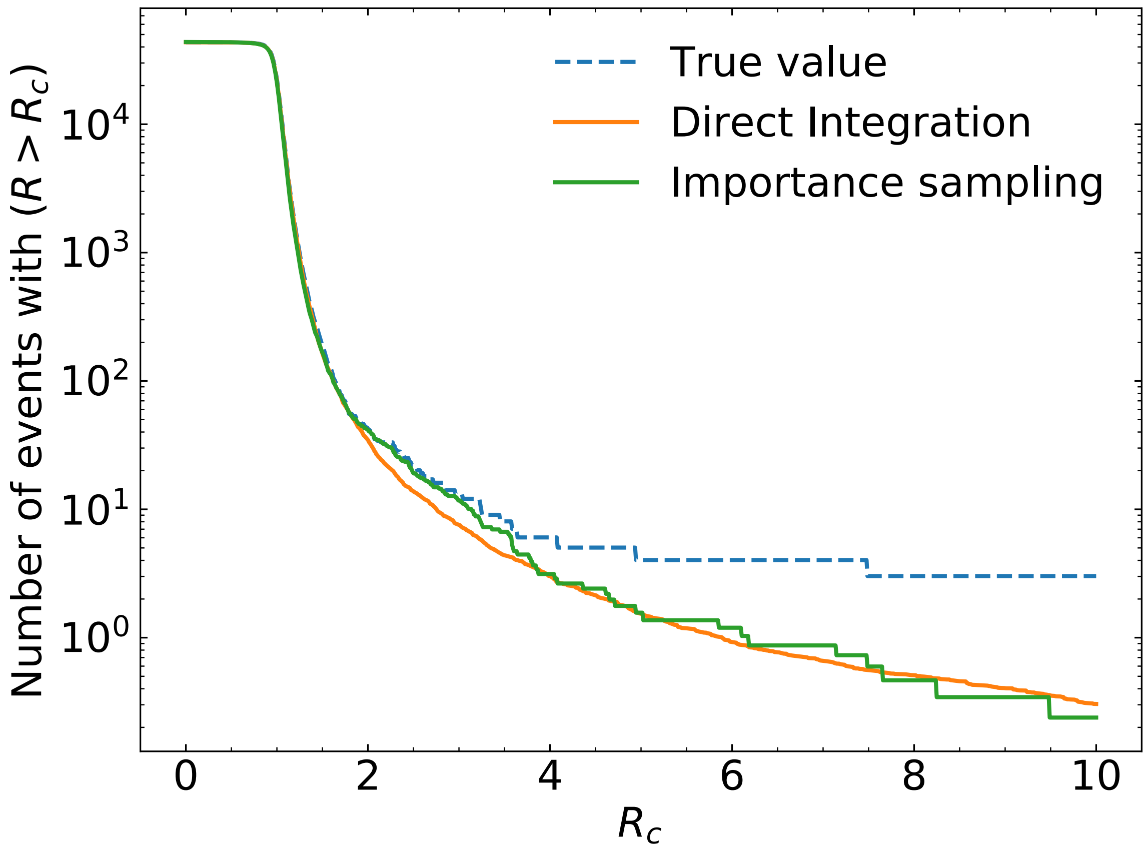
Can enhance the significance of the bump hunt by a factor of up to 7!

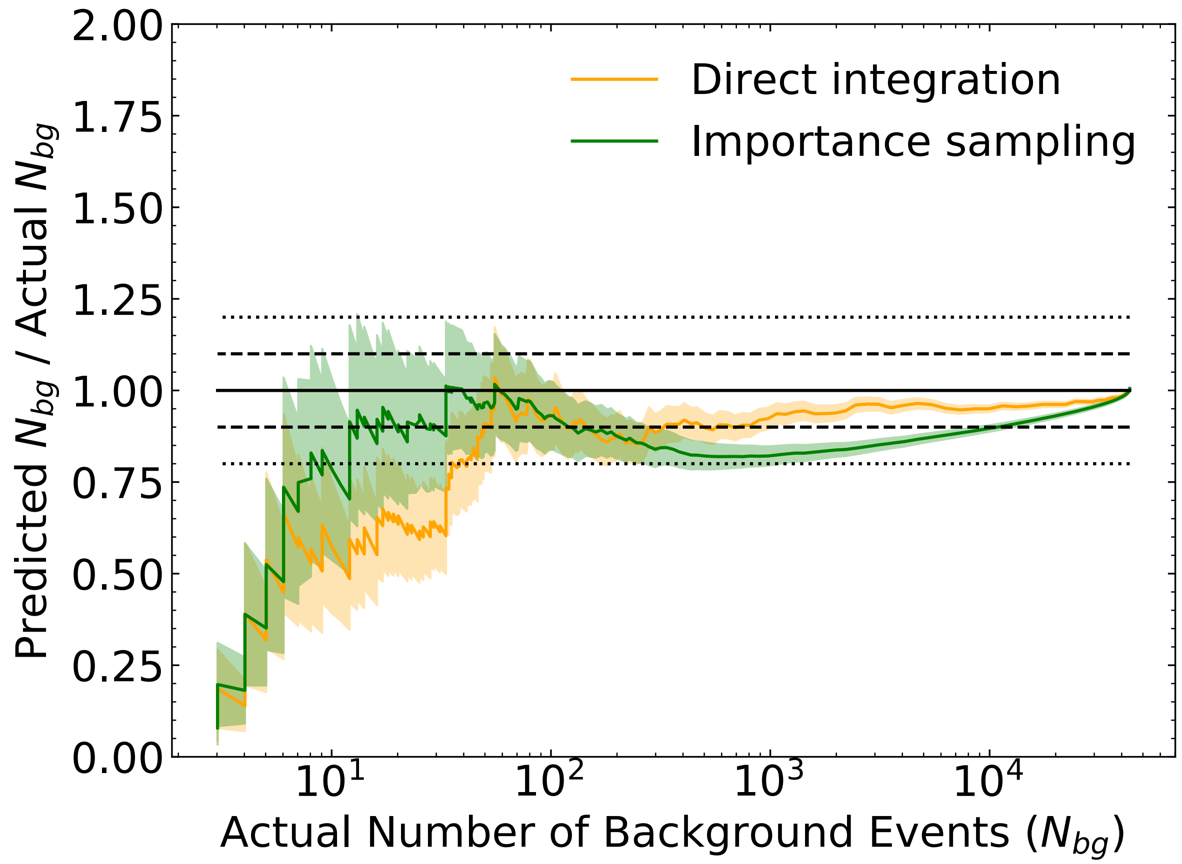
(For this feature set, the CWoLa independence assumptions are satisfied, and it outperforms ANODE. Shows the power of likelihood free methods.)

ANODE: Results on LHCO R&D Dataset

Ben Nachman & DS 2001.04990

Novel aspect of ANODE: can estimate backgrounds directly with $L(x|B_{data;}\, m \in SR)$

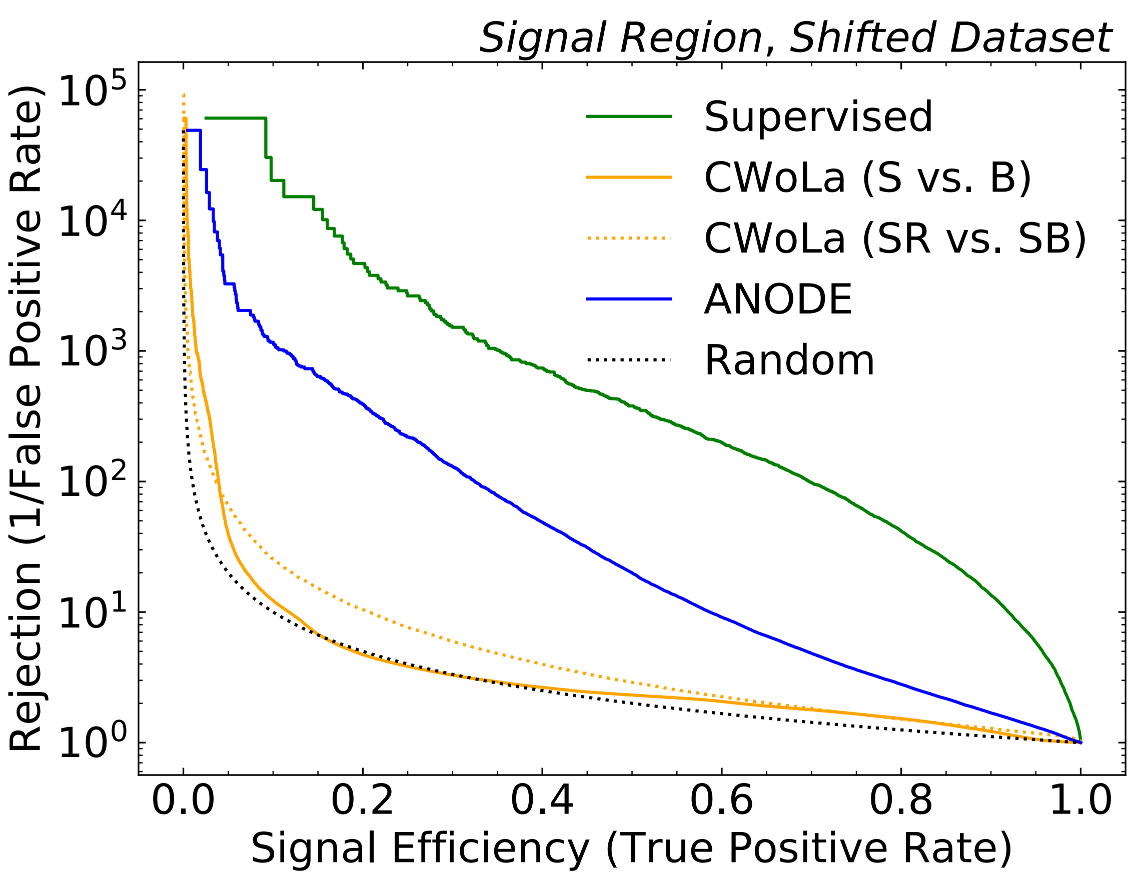


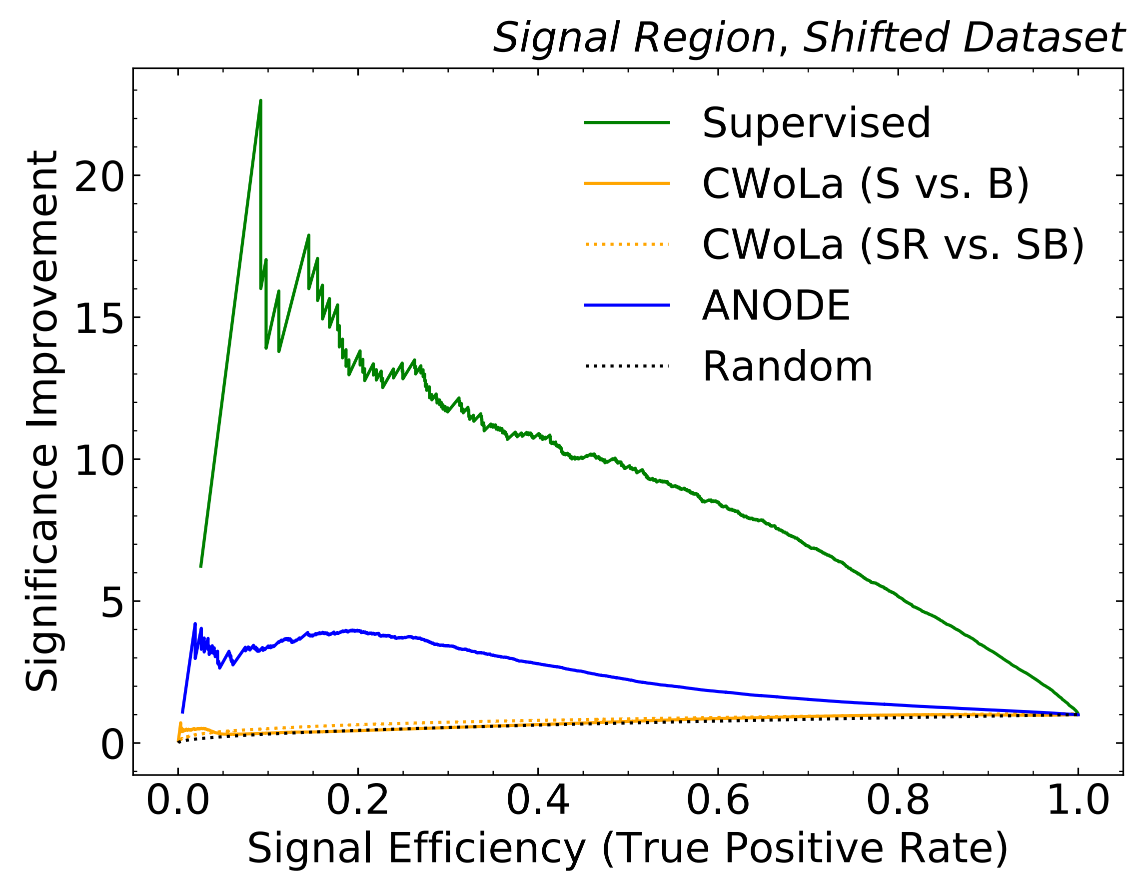


ANODE: Results on LHCO R&D Dataset

Ben Nachman & DS 2001.04990

Can also consider performance on a feature set which is not independent of m. We introduced artificial correlations just as proof of concept: $m_{J_{1,2}} \to m_{J_{1,2}} + c\, m_{JJ}$





ANODE is robust while CWoLa completely fails!

LHC Olympics 2020: Submission format

A p-value associated with the dataset having no new particles (null hypothesis).

Short answer text

As complete a description of the new physics as possible. For example: the masses and decay modes of all new particles (and uncertainties on those parameters).

Short answer text

How many signal events (+uncertainty) are in the dataset (before any selection criteria).

Short answer text

Please consider submitting plots or a Jupyter notebook! (these will be private and used only for the presentation / documentation at the end)

Add file

https://docs.google.com/forms/d/e/1FAIpQLScw323fa9qpLbdMvGtr2YeqcGTjE5Zm18-umIDiPldi_cWxVA/viewform

Overview of submissions

- 10 groups submitted results on box 1
- 4 of these groups also submitted results on boxes 2 & 3
- A number of additional groups could not finish the challenge in time but got results on the R&D dataset
- 7 of these groups gave talks about their methods and results at the ML4Jets2020 conference

Overview of submissions

People tried both supervised and unsupervised methods.

Methods used included

- Autoencoders
- CWoLa hunting
- PCA outlier detection
- LSTM
- CNN+BDT
- variational RNNs for anti-QCD tagging
- density estimation
- biological neural network
- …

Box 1

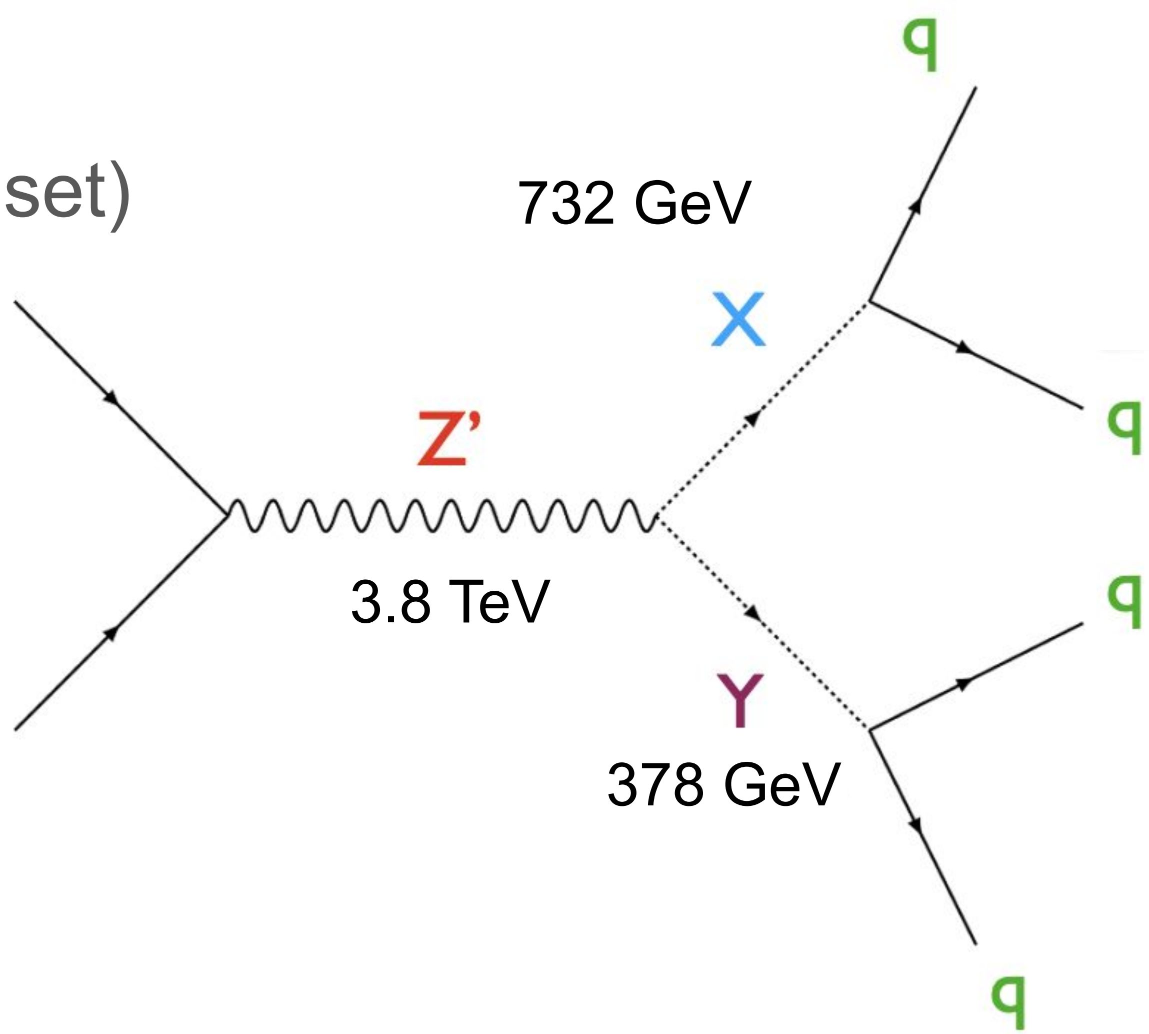
Signal: 834 events

Z’->XY; X,Y->qq
(same topology as R&D dataset)

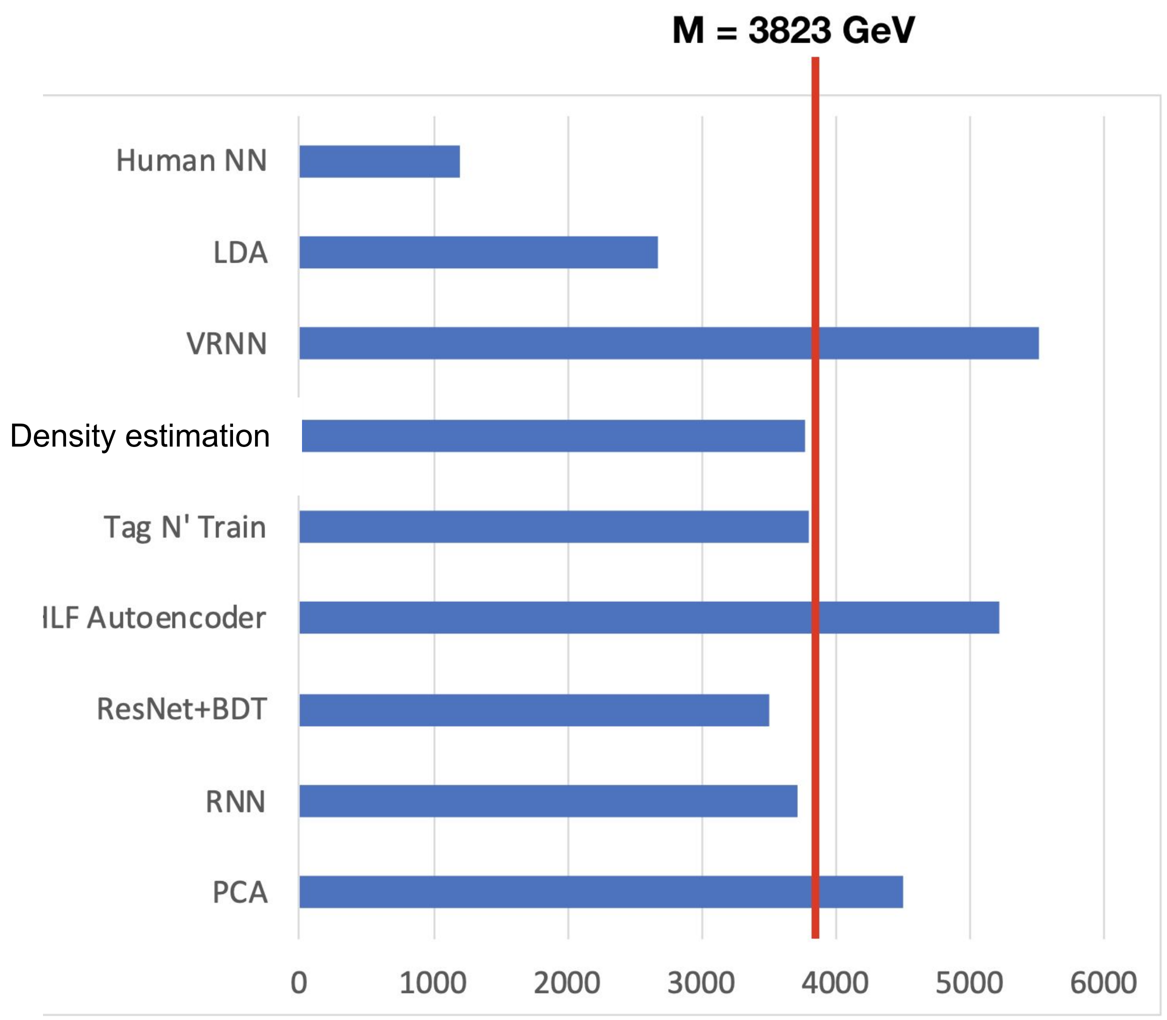
mZ’ = 3823 GeV

mX = 732 GeV

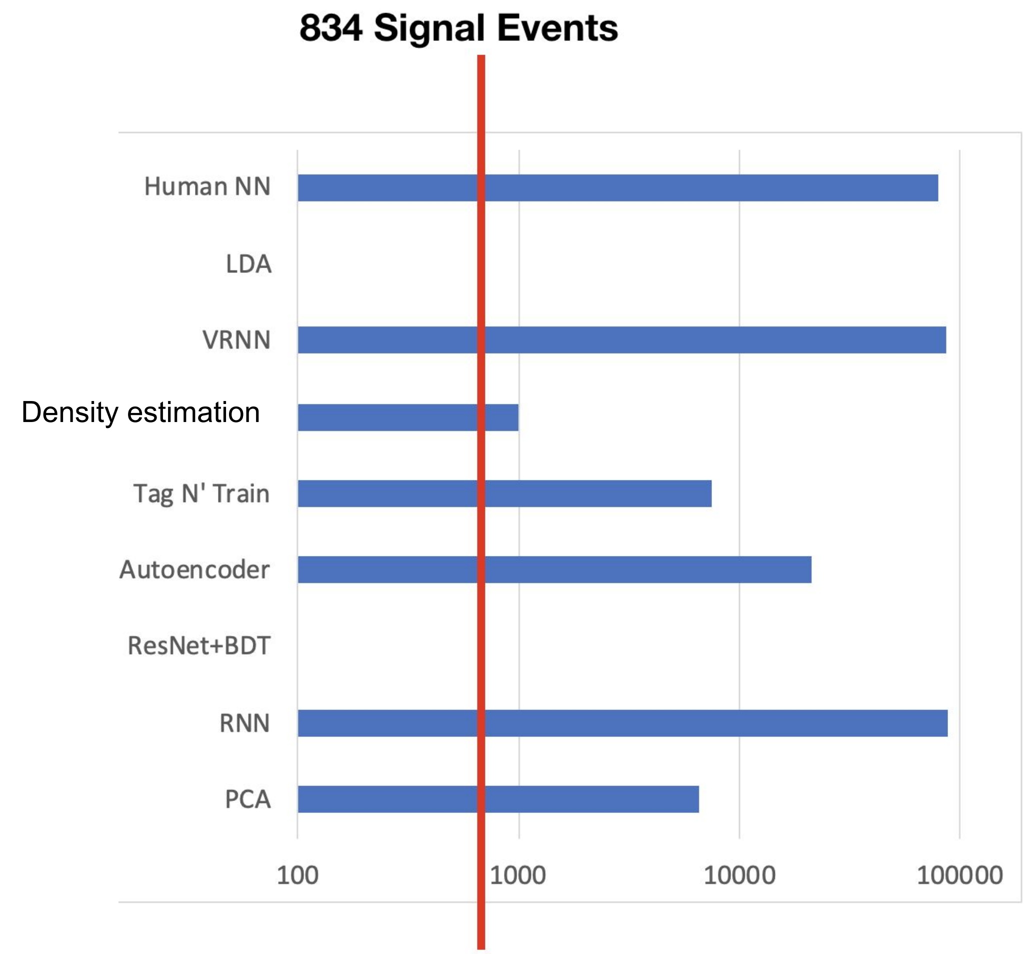
mY = 378 GeV



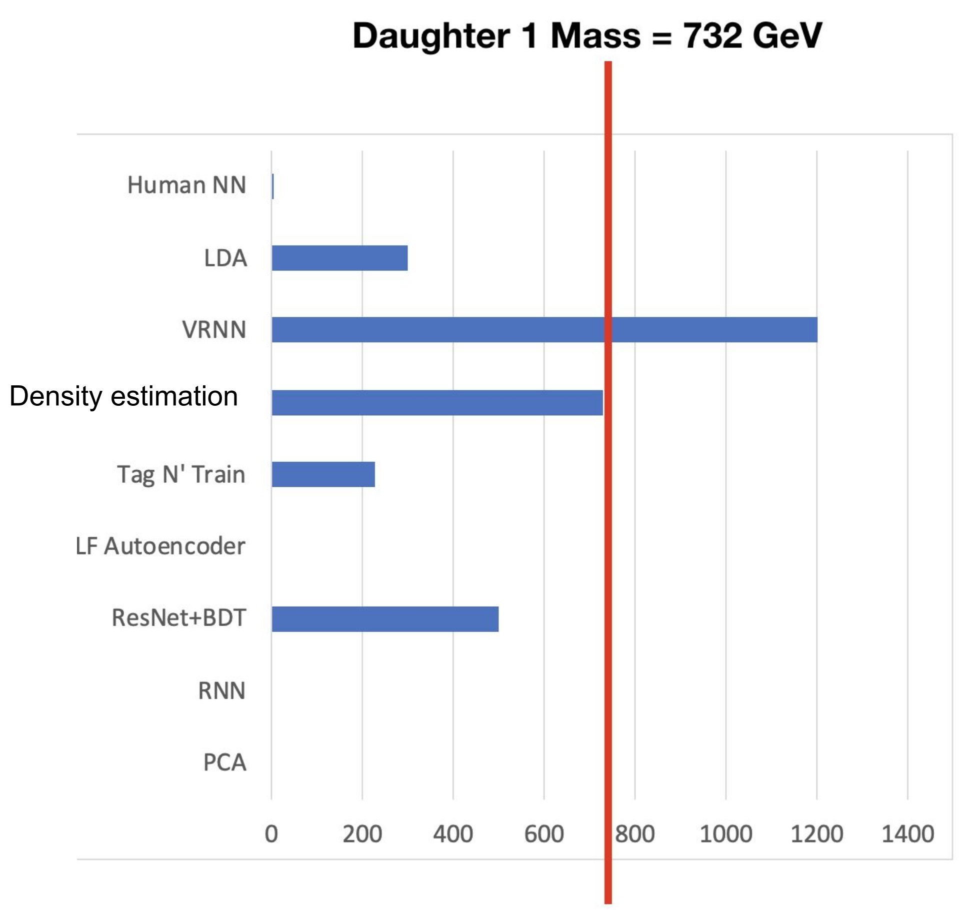
We revealed the answer at the ML4Jets2020 conference in early January



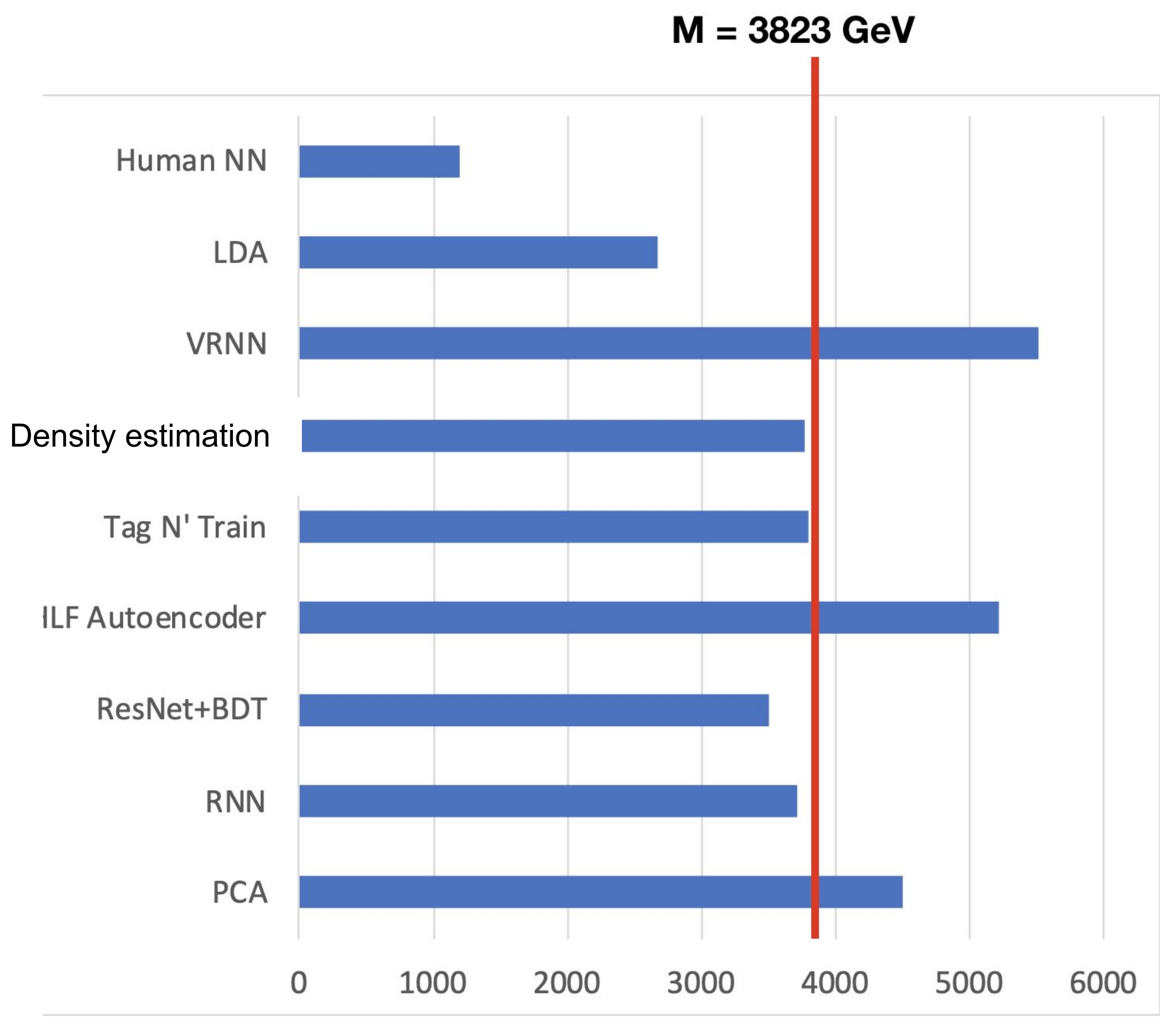
M = 3823 GeV
Human NN
LDA
VRNN
Density estimation
Tag N' Train
ILF Autoencoder
ResNet+BDT
RNN
PCA
0
1000
2000
3000
4000
5000
6000

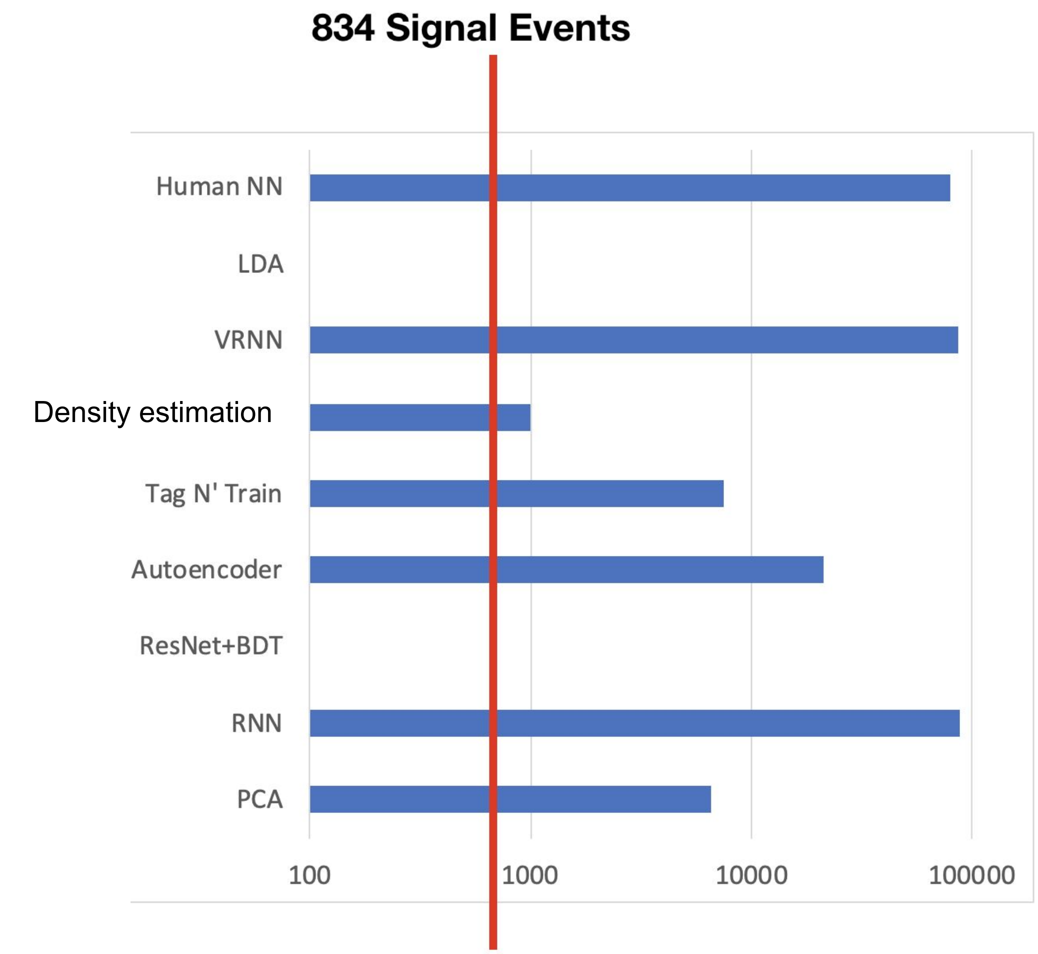


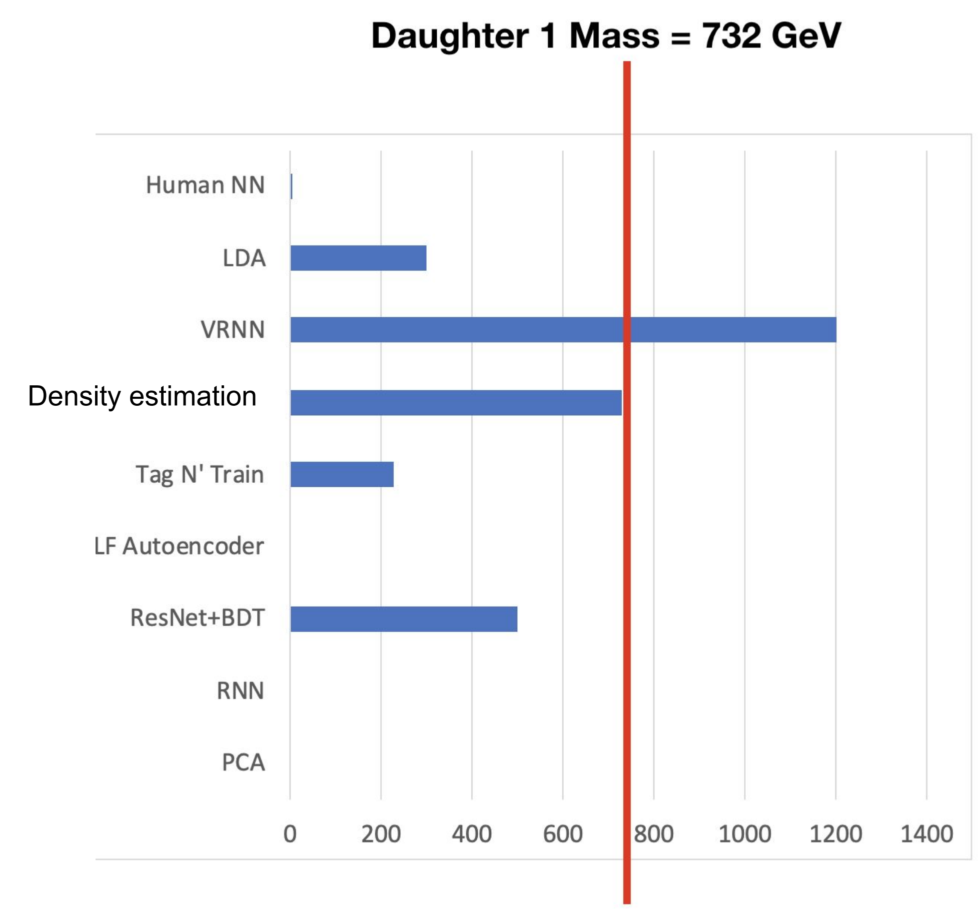
834 Signal Events
Human NN
LDA
VRNN
Density estimation
Tag N' Train
Autoencoder
ResNet+BDT
RNN
PCA
100
1000
10000
100000



Daughter 1 Mass = 732 GeV
Human NN
LDA
VRNN
Density estimation
Tag N' Train
LF Autoencoder
ResNet+BDT
RNN
PCA
0
200
400
600
800
1000
1200
1400







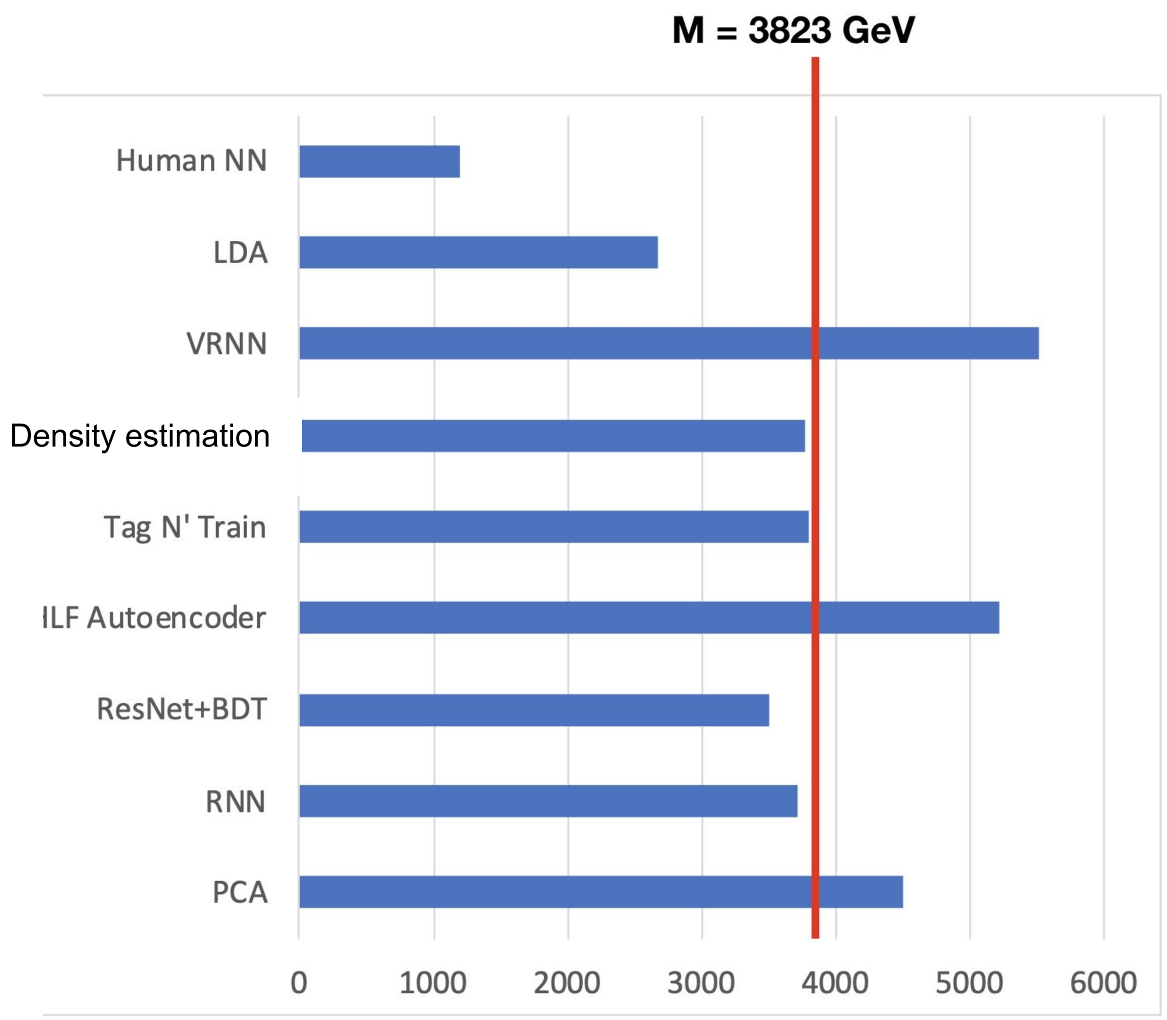
A clear winner emerged:

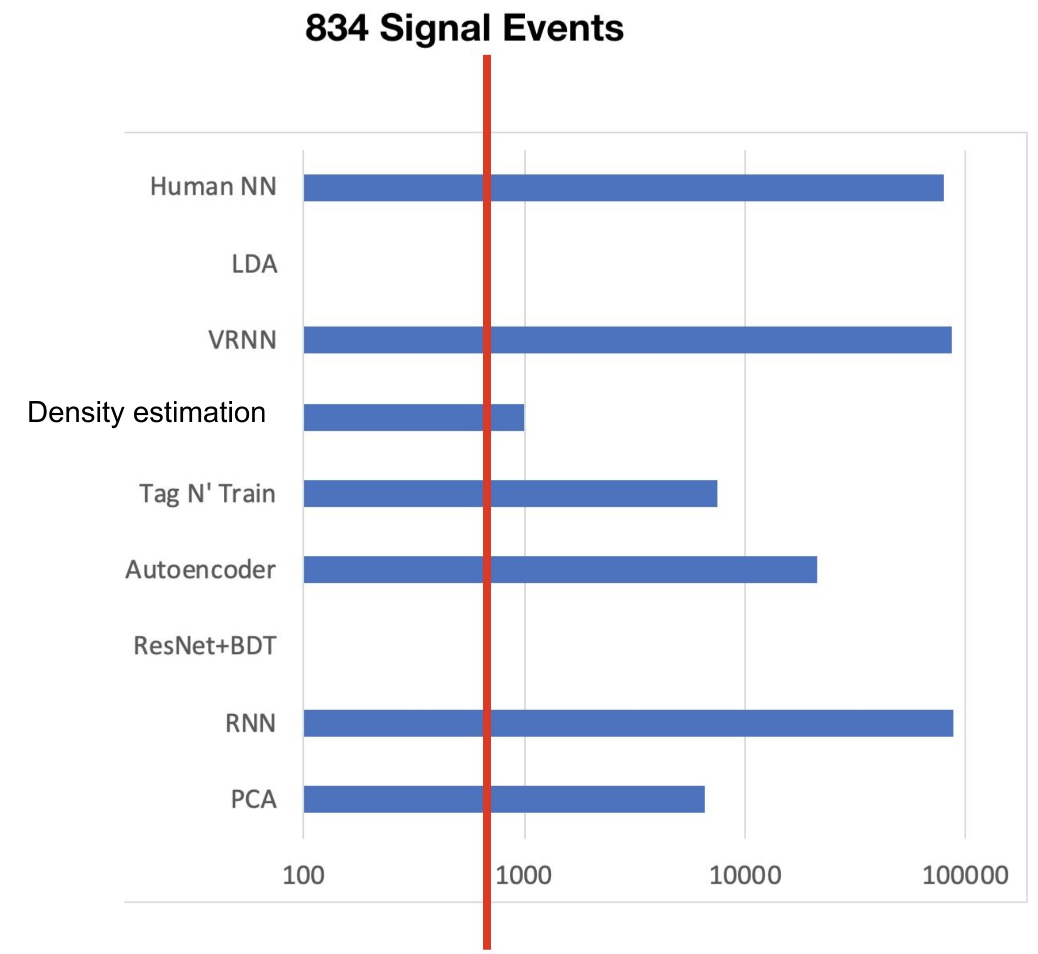
Conditional density estimation for anomaly detection

George Stein, Uros Seljak, Biwei Dai, He Jia

BERKELEY CENTER *for* COSMOLOGICAL PHYSICS

Used the ANODE method with a novel density estimator!





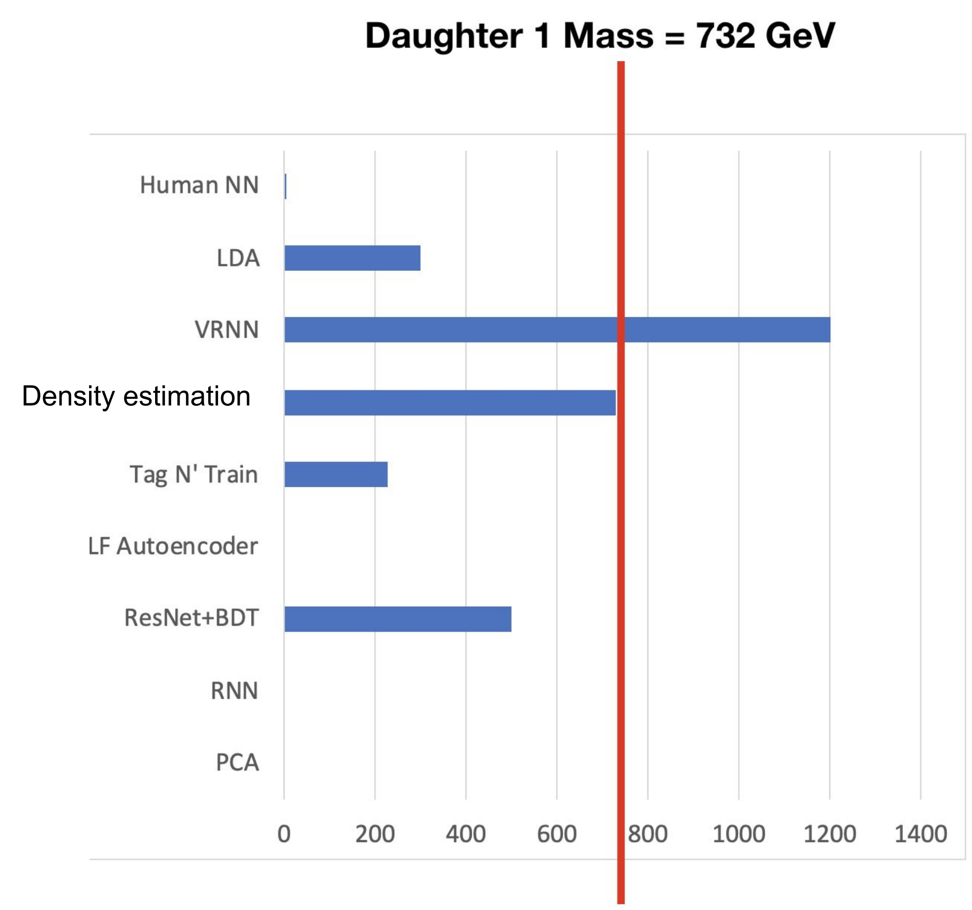
In second place:

Tag N' Train

Oz Amram & Cristina Mantilla Suarez (Johns Hopkins)

JOHNS HOPKINS UNIVERSITY

Used a combination of autoencoders and CWoLa hunting



LHCO2020: Summer Games

https://indico.desy.de/indico/event/25341/

Stay tuned for more on the LHCO 2020…

We will be organizing a 1-day mini-workshop on anomaly detection in Hamburg the Saturday before BOOST (July 18).

There the answers for Boxes 2 and 3 will be revealed.

We will also discuss plans for a community paper on new methods for anomaly detection and the LHCO2020.

Please come and join us!

Conclusions

These are exciting times for anomaly detection in HEP.

Many new approaches making use of unsupervised ML are being developed by theorists and experimentalists.

Model independent searches have a bright future at the LHC. Maybe this is how we will finally discover the new physics!

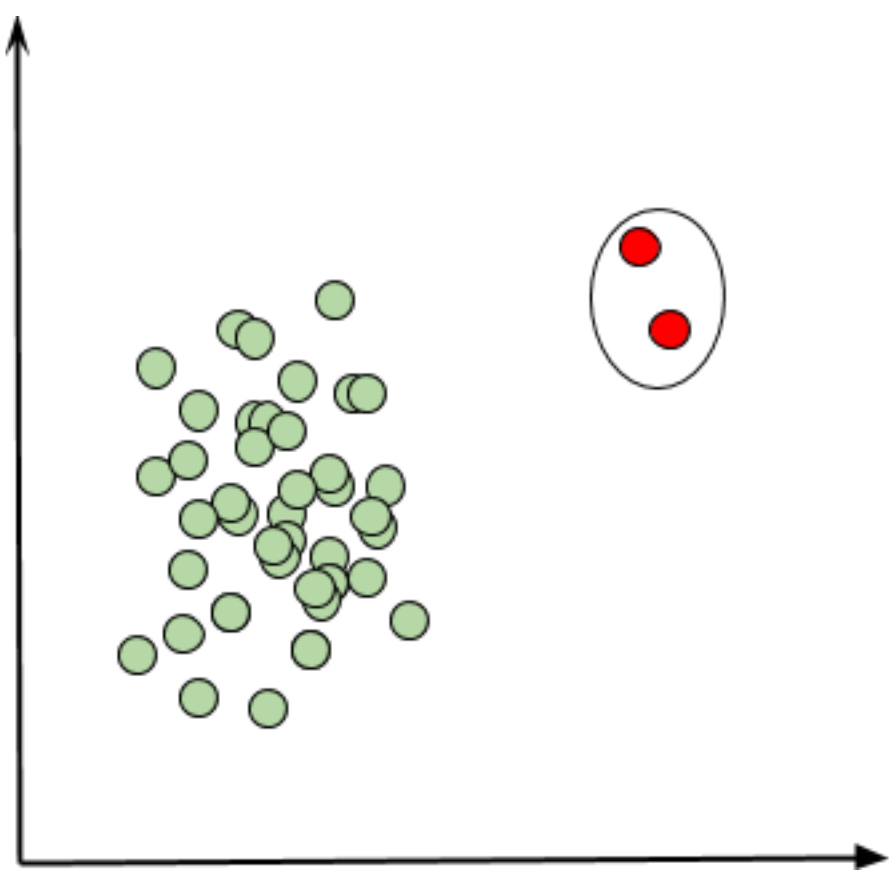
These methods also have potential applications beyond HEP. For example, ANODE is a completely general method for finding localized overdensities in high dimensional datasets. One can imagine many uses for such a method!

Thanks for your attention!

Anomaly Detection at the LHC

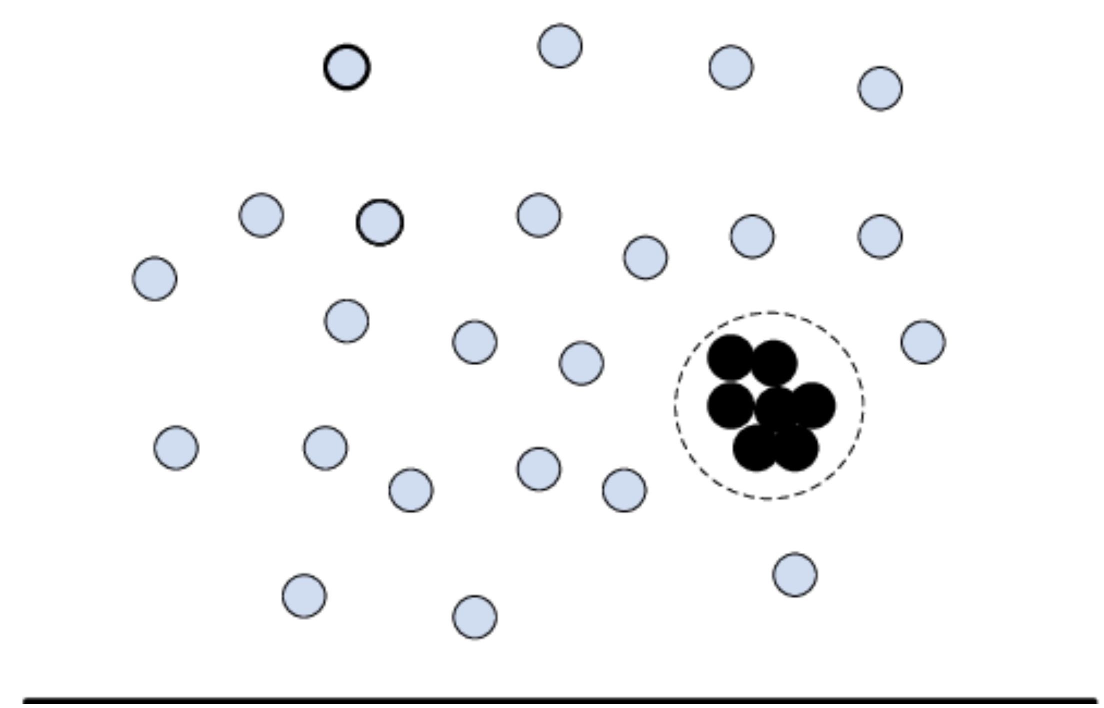
In the broader ML field, there are two types of anomaly detection

point outlier, out of sample anomaly,
"zero-background search"



easy

collective outlier, population anomaly
"bump hunt"

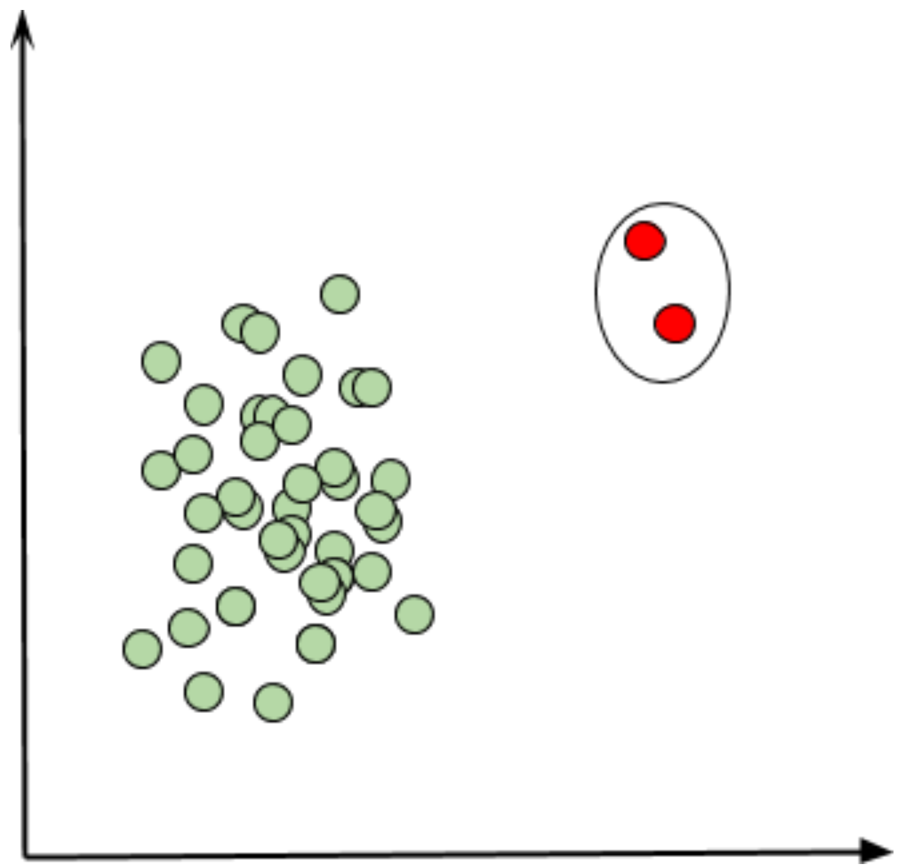


hard

Anomaly Detection at the LHC

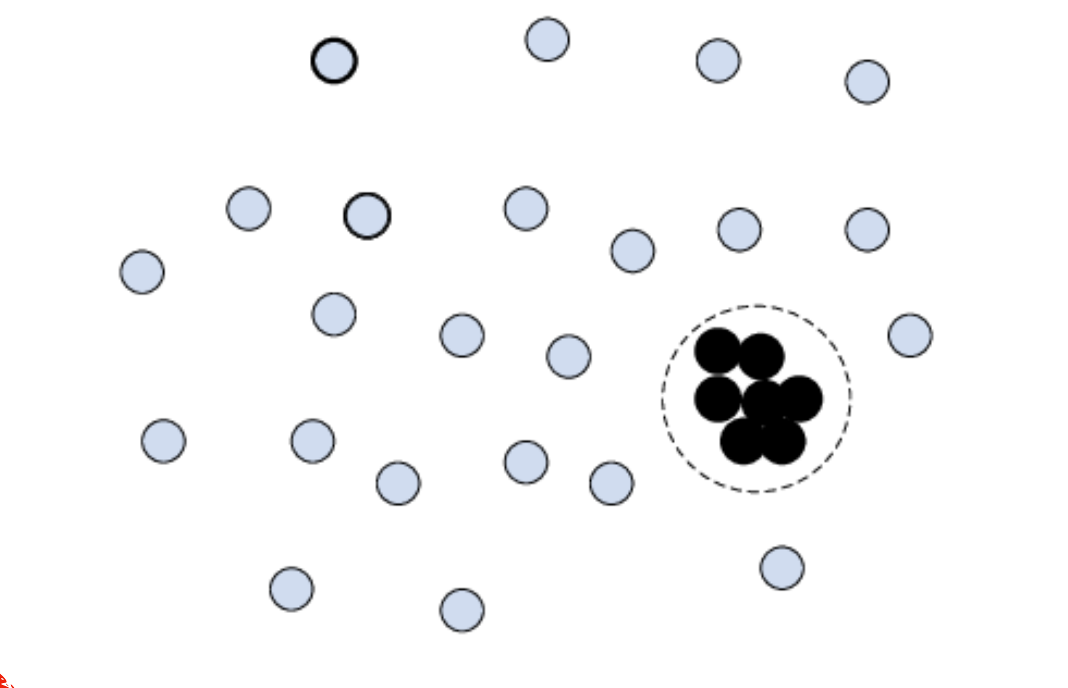
In the broader ML field, there are two types of anomaly detection

point outlier, out of sample anomaly,
"zero-background search"



easy

collective outlier, population anomaly
"bump hunt"

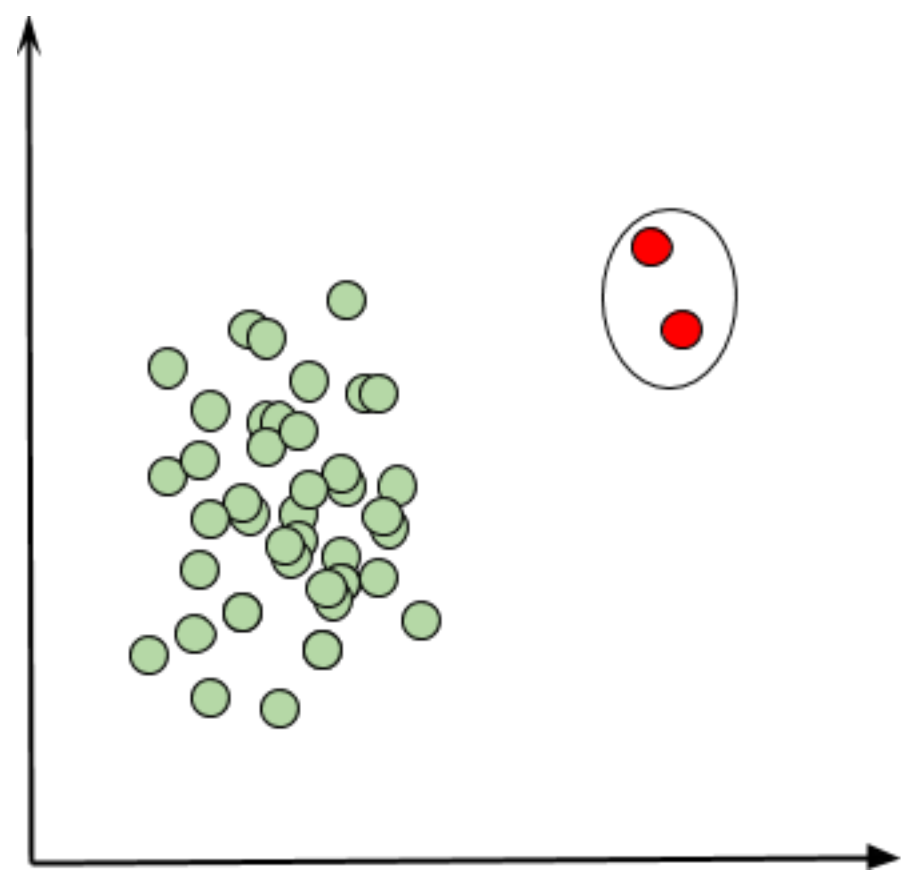


hard

Anomaly Detection at the LHC

In the broader ML field, there are two types of anomaly detection

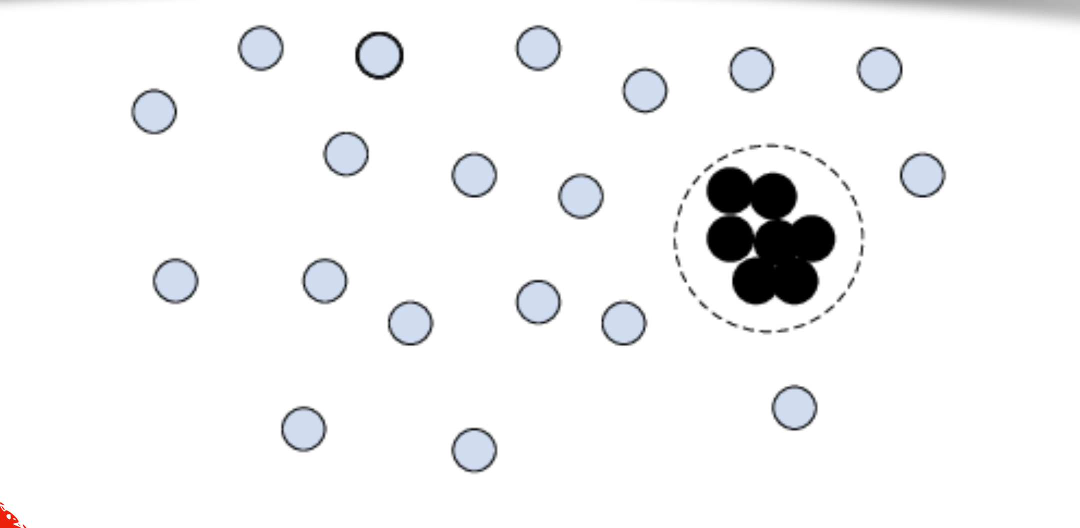
point outlier, out of sample anomaly,
“zero-background search”



easy

collective outlier, population anomaly
“bump hunt”

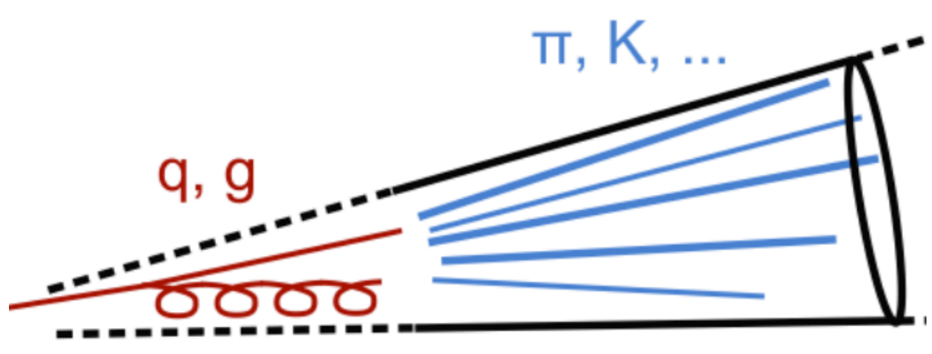
Background estimation is a key component of anomaly detection in HEP



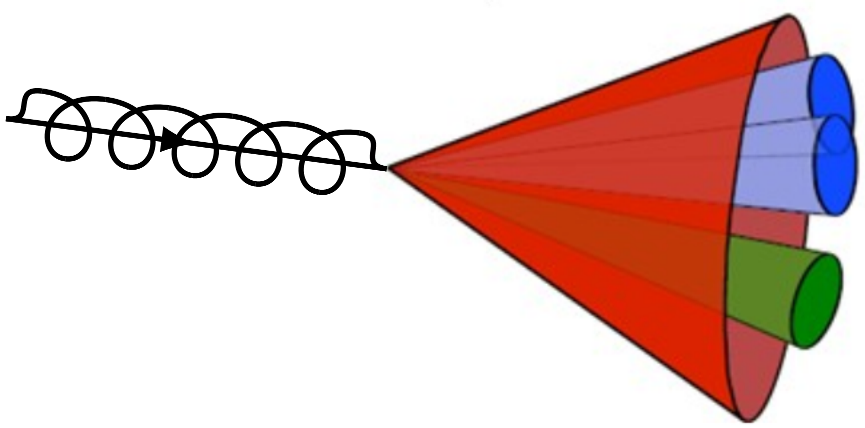
hard

Sample definitions

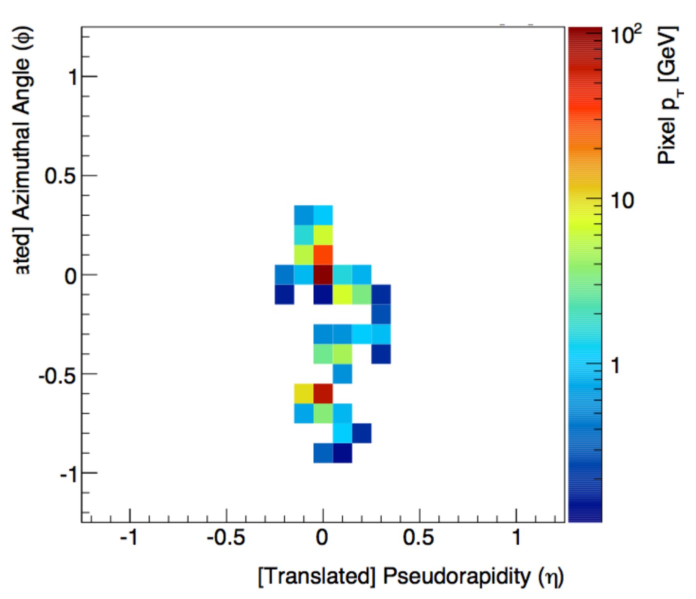
- Background: QCD jets
  (p_T: 800-900 GeV, $|\eta|<1$, anti-kt R=1)



- Signals:
  - All-hadronic tops
  - 400 GeV gluinos decaying via RPV



- We formed jet images in η and ϕ with a pixel resolution and intensity given by the calorimeter towers.



Searching for NP with deep autoencoders

Heimel et al 1808.08979;   Farina, Nakai & DS 1808.08992

Loss function for autoencoder:

"reconstruction error"

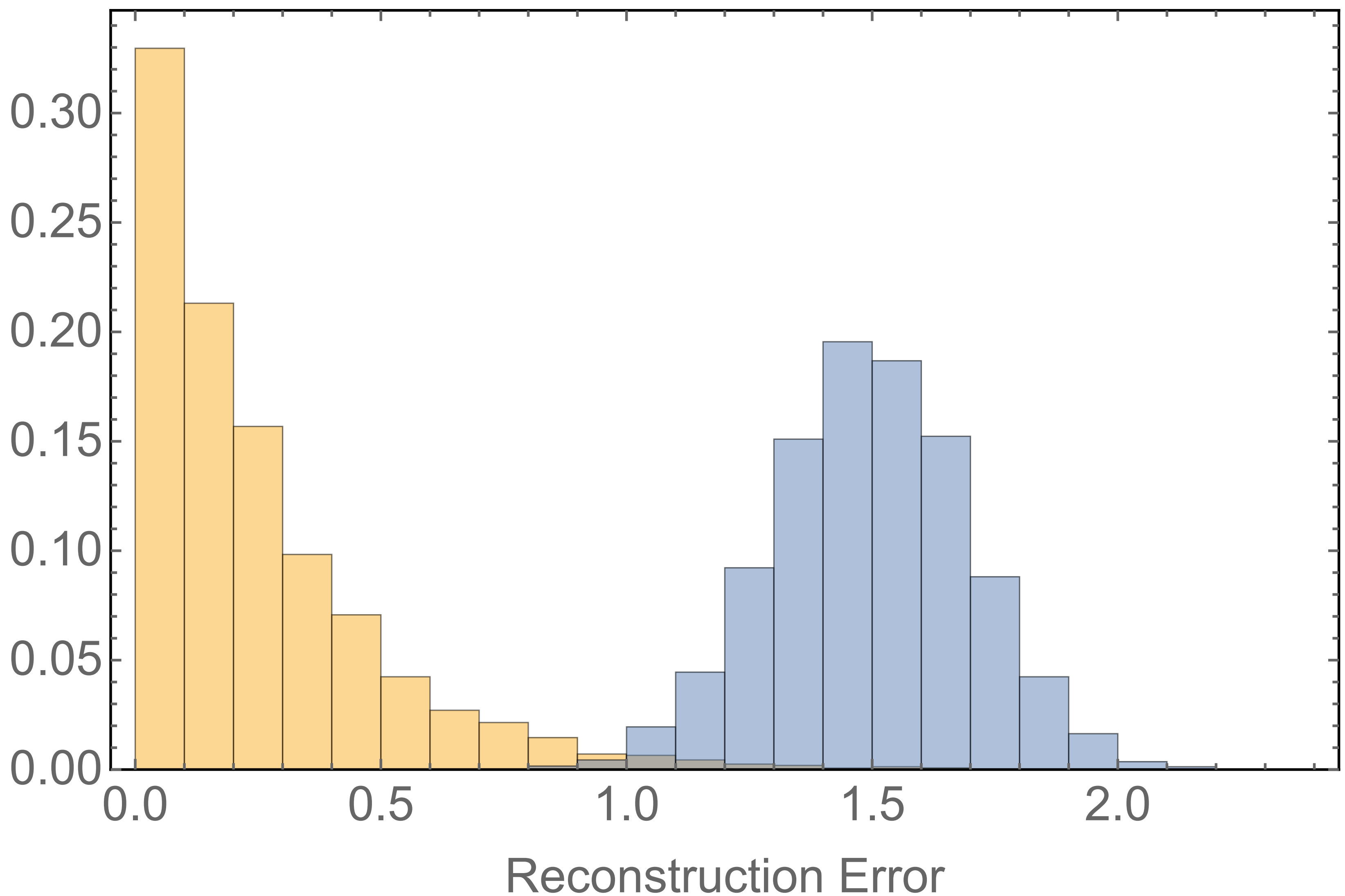
$$L = \frac{1}{N}\sum_{i=1}^{N}(x_i^{in} - x_i^{out})^2$$

Searching for NP with deep autoencoders

Heimel et al 1808.08979;   Farina, Nakai & DS 1808.08992

Loss function for autoencoder: $$L = \frac{1}{N}\sum_{i=1}^{N}(x_i^{in} - x_i^{out})^2$$

"reconstruction error"



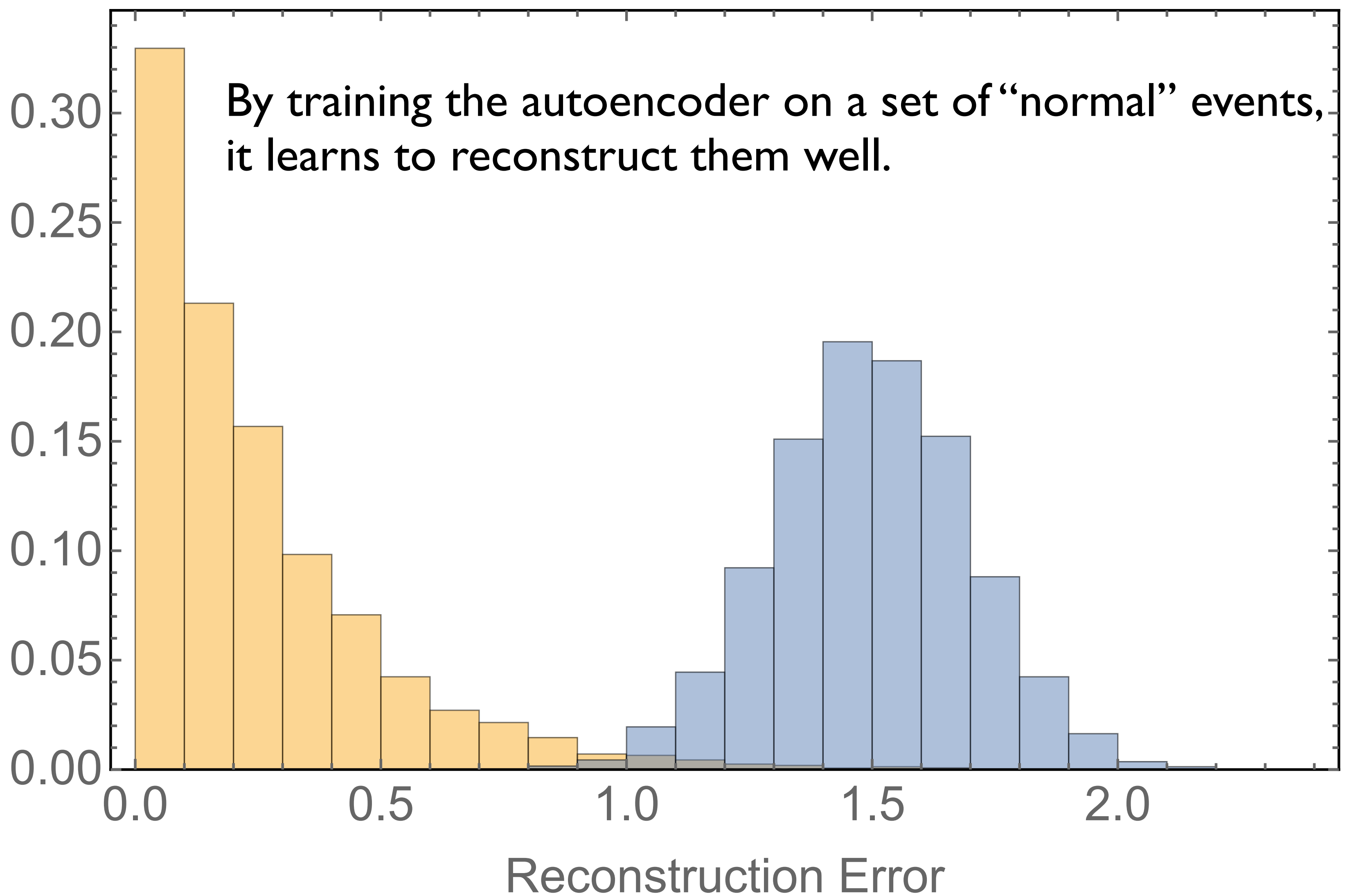
Searching for NP with deep autoencoders

Heimel et al 1808.08979;   Farina, Nakai & DS 1808.08992

Loss function for autoencoder:

"reconstruction error"

$$L = \frac{1}{N}\sum_{i=1}^{N}(x_i^{in} - x_i^{out})^2$$

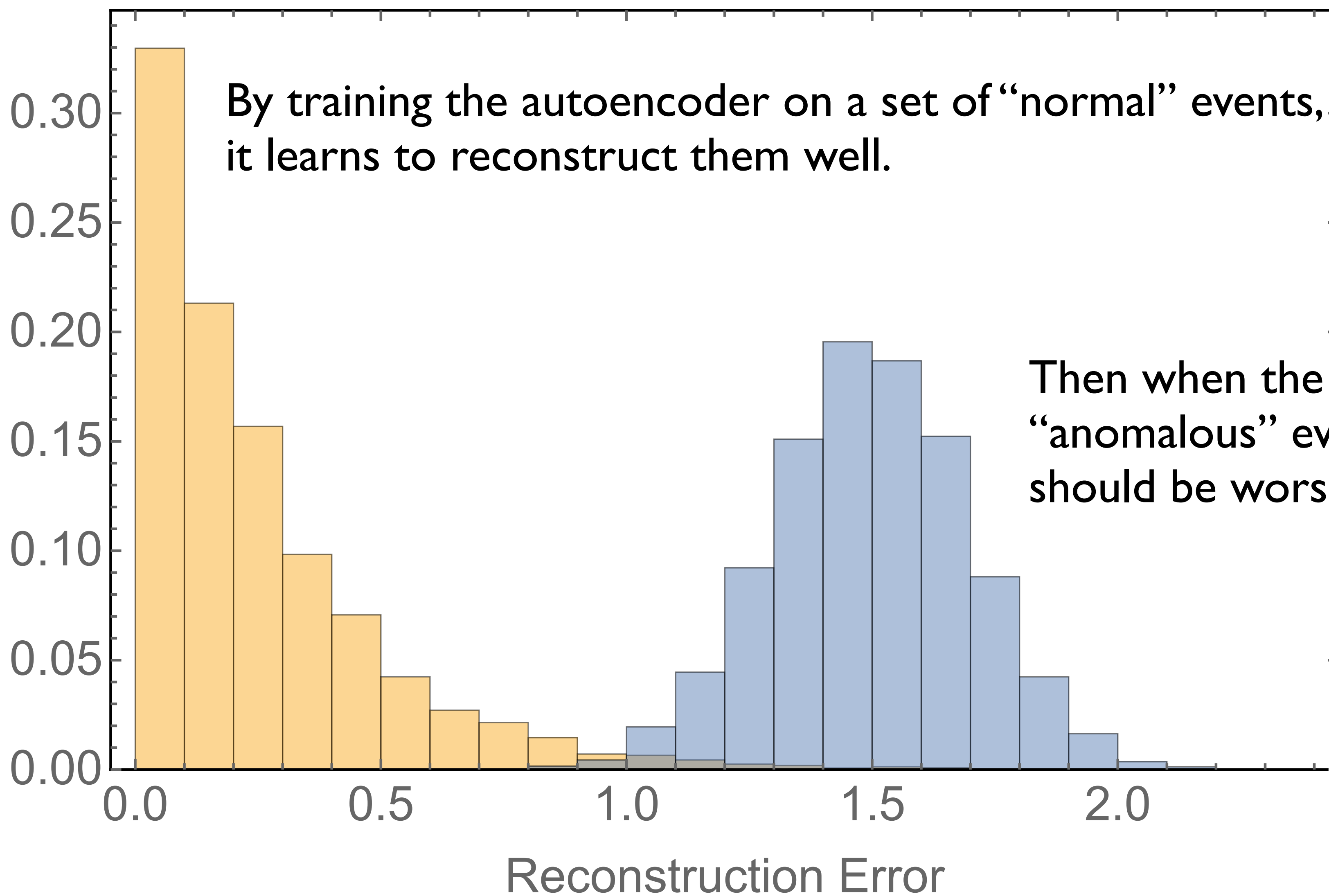


Searching for NP with deep autoencoders

Heimel et al 1808.08979; Farina, Nakai & DS 1808.08992

Loss function for autoencoder: $$L = \frac{1}{N}\sum_{i=1}^{N}(x_i^{in} - x_i^{out})^2$$

"reconstruction error"



By training the autoencoder on a set of "normal" events, it learns to reconstruct them well.

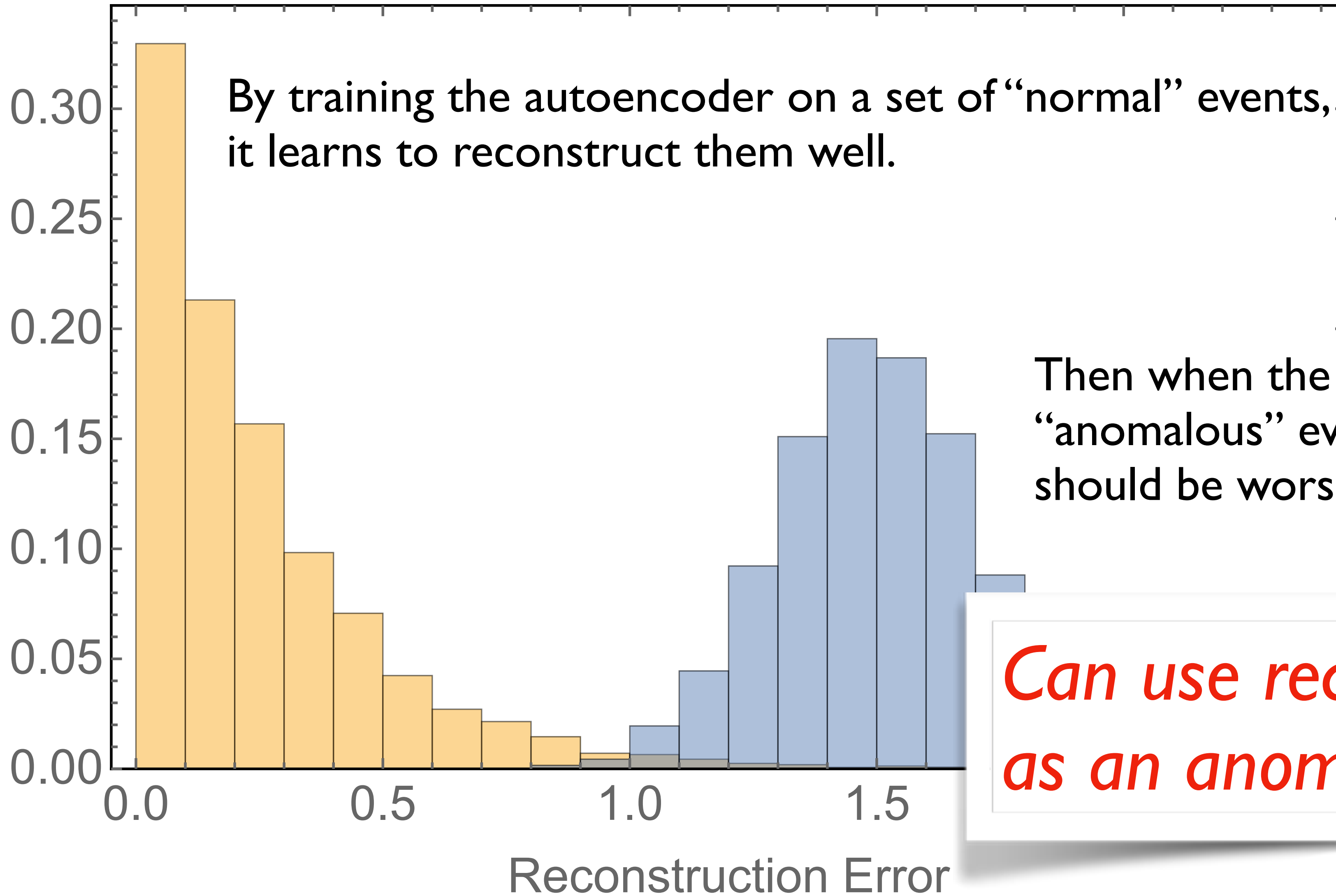
Then when the autoencoder encounters "anomalous" events, its performance should be worse.

Searching for NP with deep autoencoders

Heimel et al 1808.08979;   Farina, Nakai & DS 1808.08992

Loss function for autoencoder: $$L = \frac{1}{N}\sum_{i=1}^{N}(x_i^{in} - x_i^{out})^2$$

"reconstruction error"

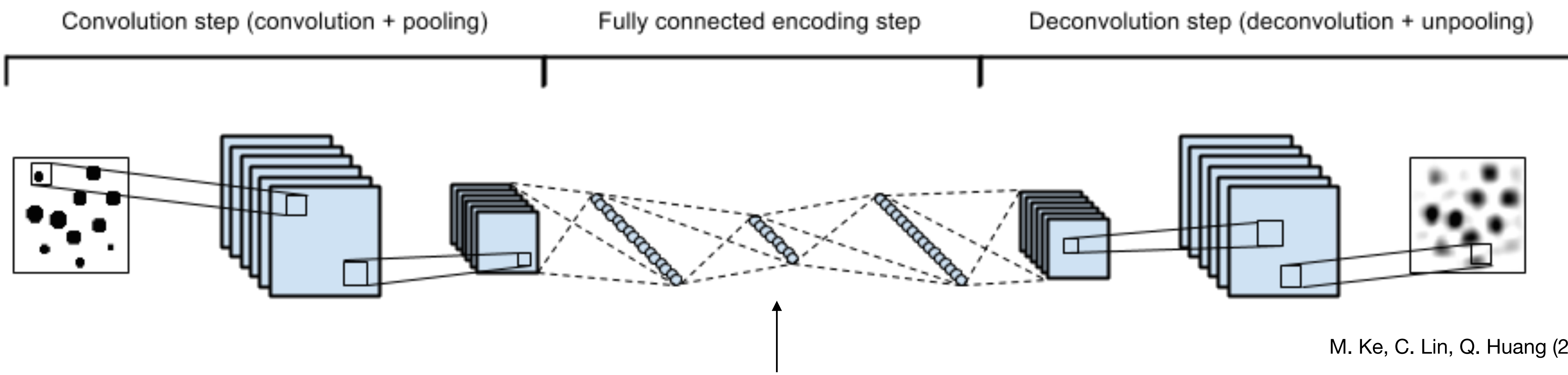


By training the autoencoder on a set of "normal" events, it learns to reconstruct them well.

Then when the autoencoder encounters "anomalous" events, its performance should be worse.

Can use reconstruction error as an anomaly threshold!

Autoencoder architecture



M. Ke, C. Lin, Q. Huang (2017)

Encoder **Latent space** **Decoder**

128C3-MP2-128C3-MP2-128C3-32N-6N-32N-12800N-128C3-US2-128C3-US2-1C3

Our primary autoencoder used convolutional neural networks (CNNs) for encoding and decoding the jet images.

We also considered autoencoders based on PCA and simple DNNs.

Many more architectures are possible.

Choosing the latent dimension

Should choose the latent dimension in an unsupervised manner (ie without optimizing on a specific signal)

- d too large → autoencoder becomes identity transform
- d too small → autoencoder cannot learn all the features

Choosing the latent dimension

Should choose the latent dimension in an unsupervised manner (ie without optimizing on a specific signal)

- d too large → autoencoder becomes identity transform
- d too small → autoencoder cannot learn all the features

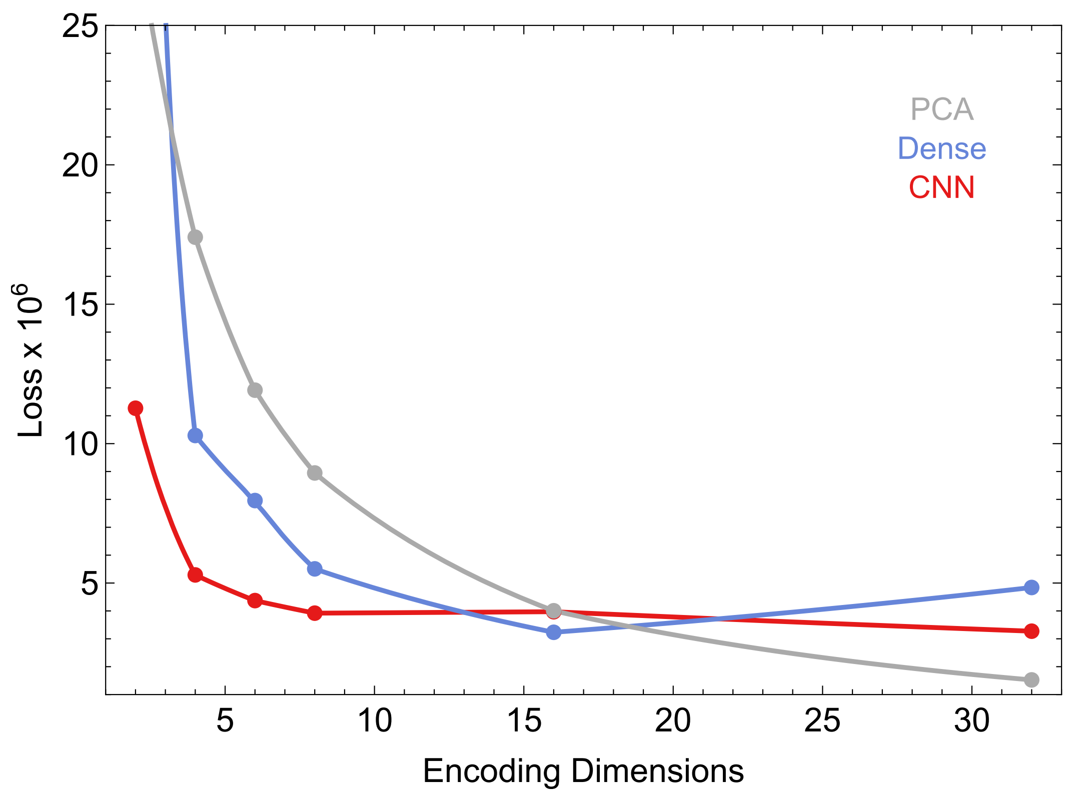
➡ *idea: look for saturation point as d is increased*

Choosing the latent dimension

Should choose the latent dimension in an unsupervised manner (ie without optimizing on a specific signal)

- d too large → autoencoder becomes identity transform
- d too small → autoencoder cannot learn all the features

idea: look for saturation point as d is increased

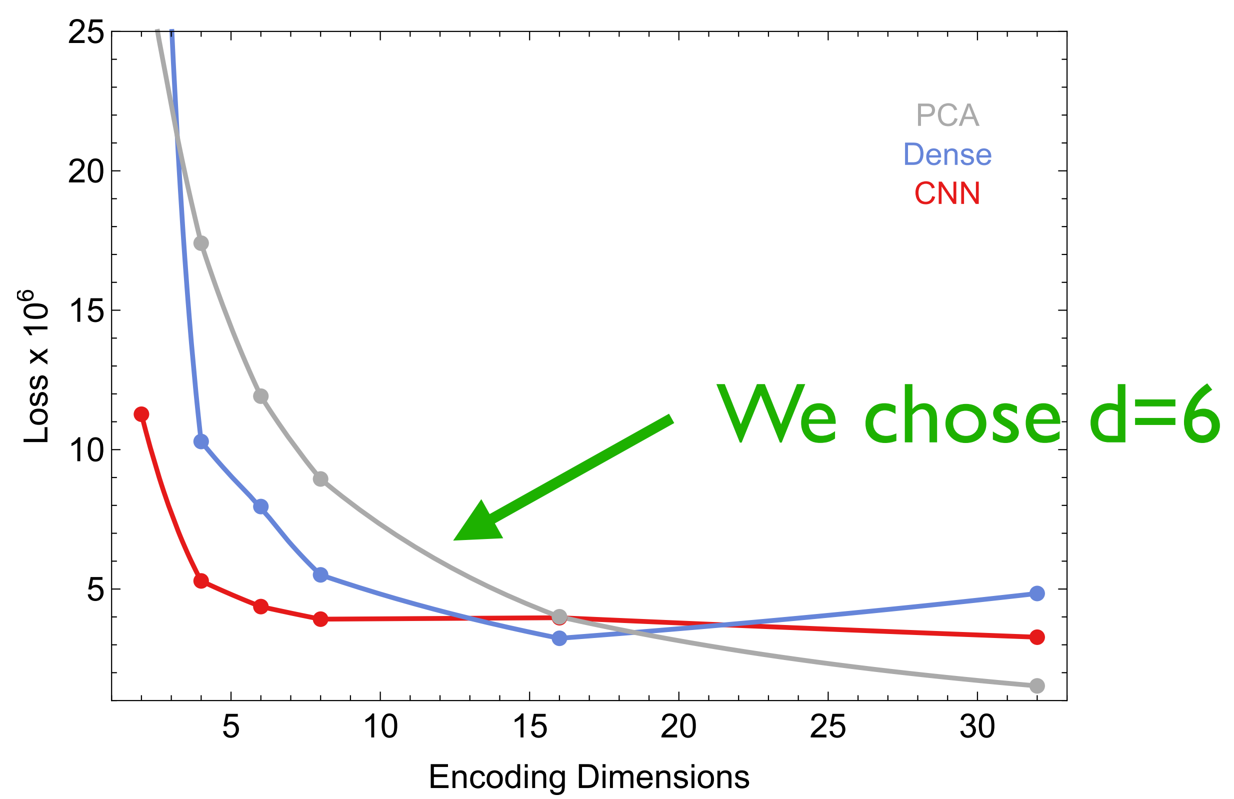


Choosing the latent dimension

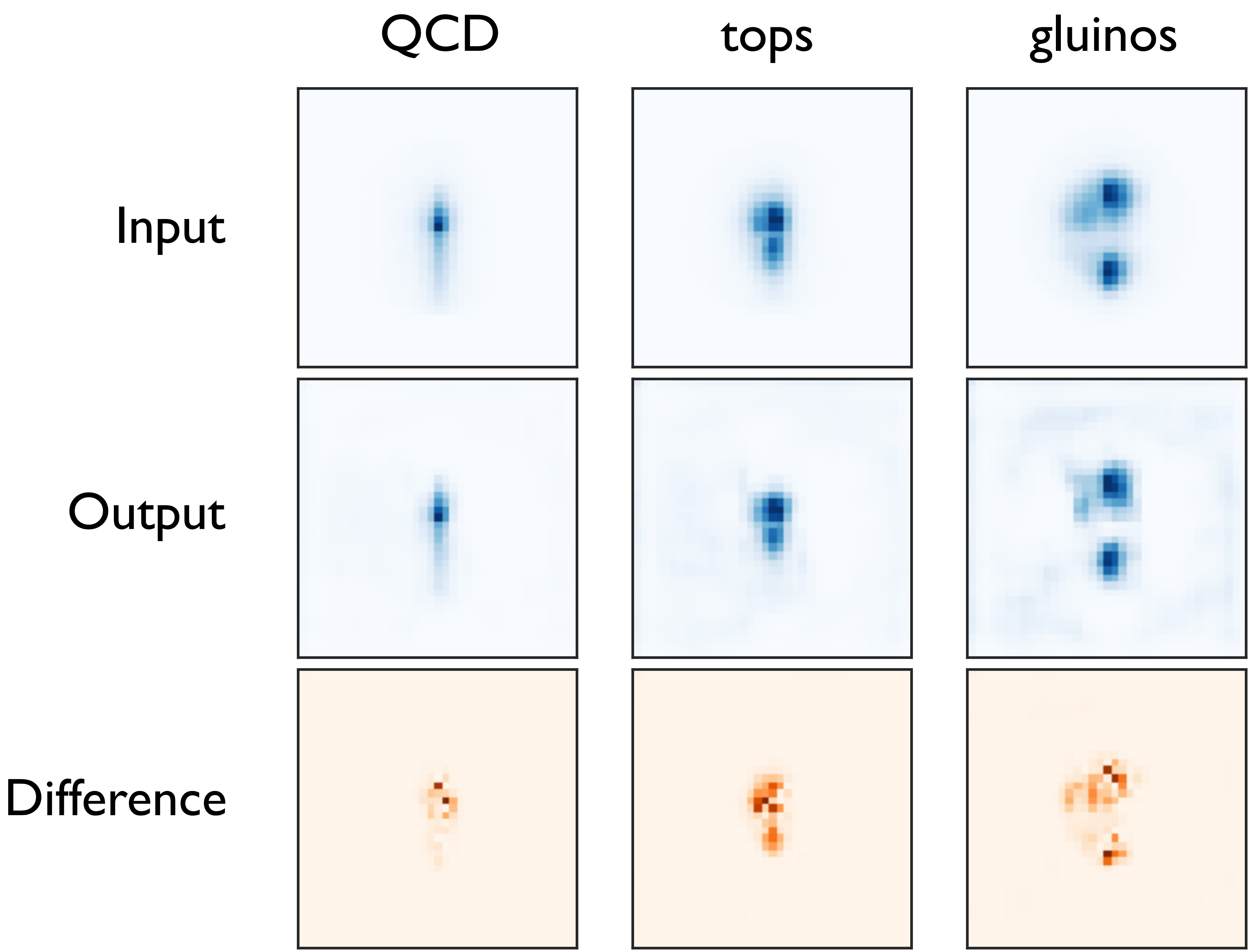
Should choose the latent dimension in an unsupervised manner (ie without optimizing on a specific signal)

- d too large → autoencoder becomes identity transform
- d too small → autoencoder cannot learn all the features

idea: look for saturation point as d is increased



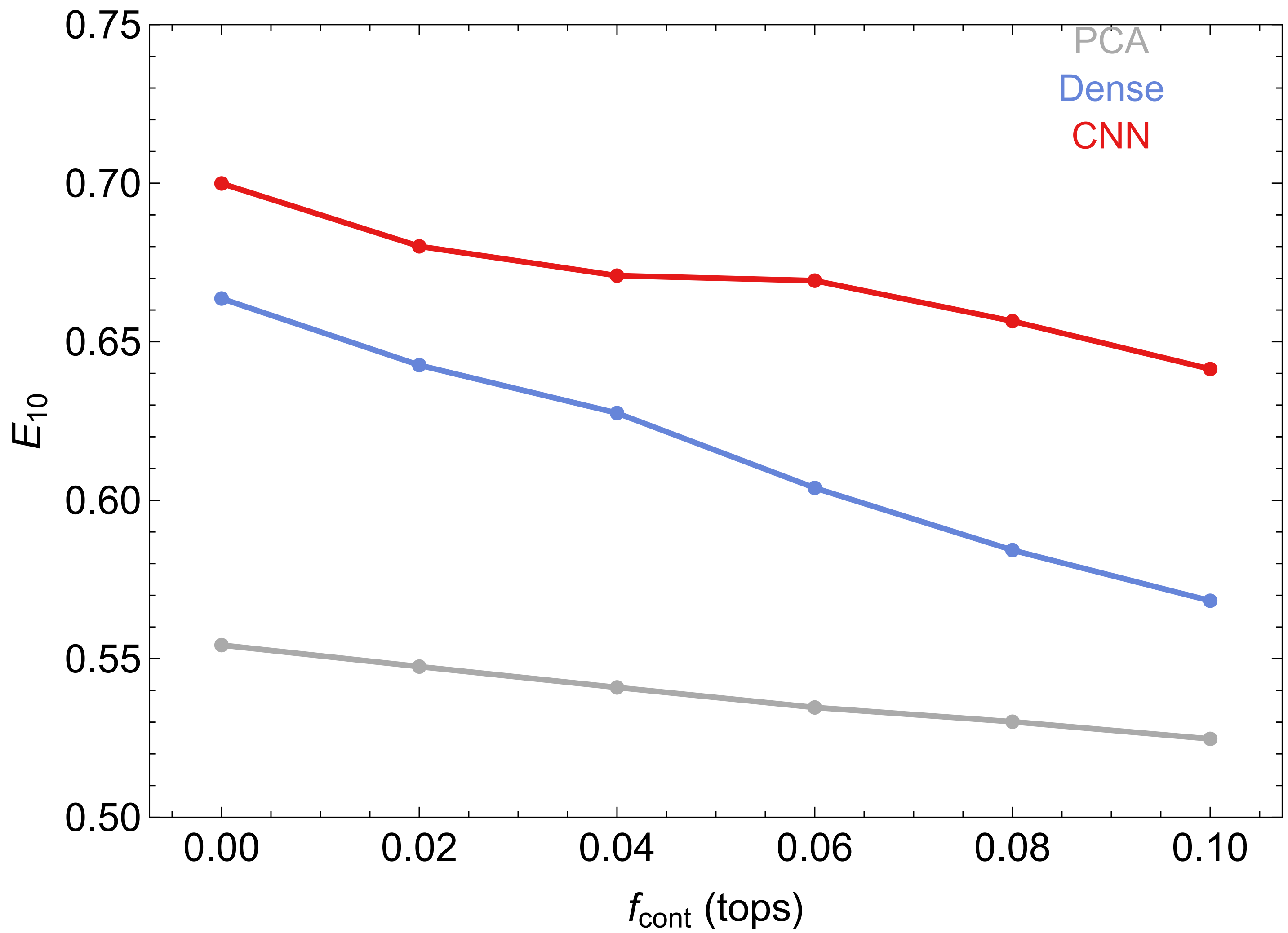
Performance: weakly supervised mode



Fully unsupervised mode

Can also train on QCD background "contaminated" with a small fraction of signal. This could be representative of actual data.

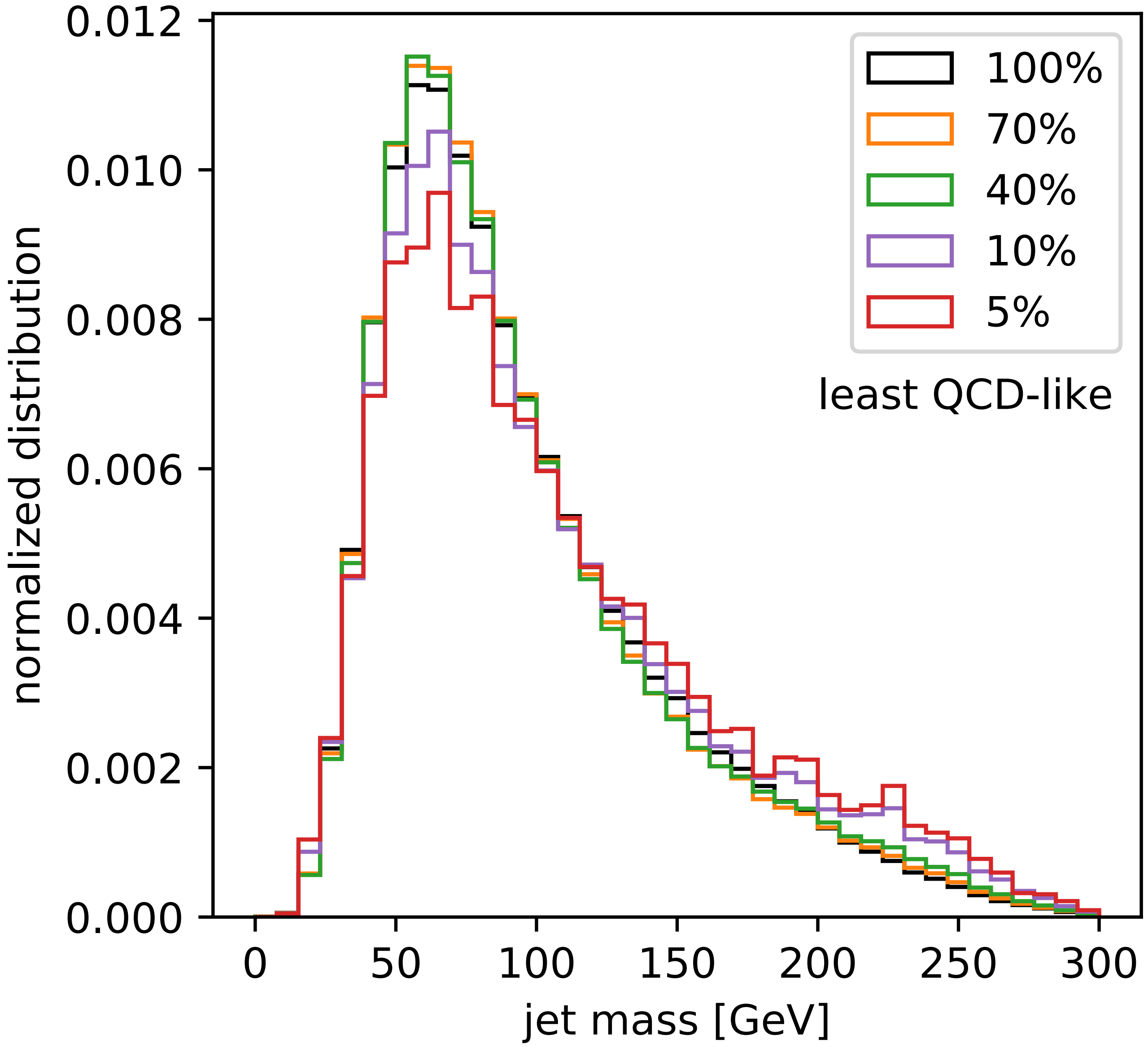
(E_{10} = signal efficiency at 10% bg efficiency)



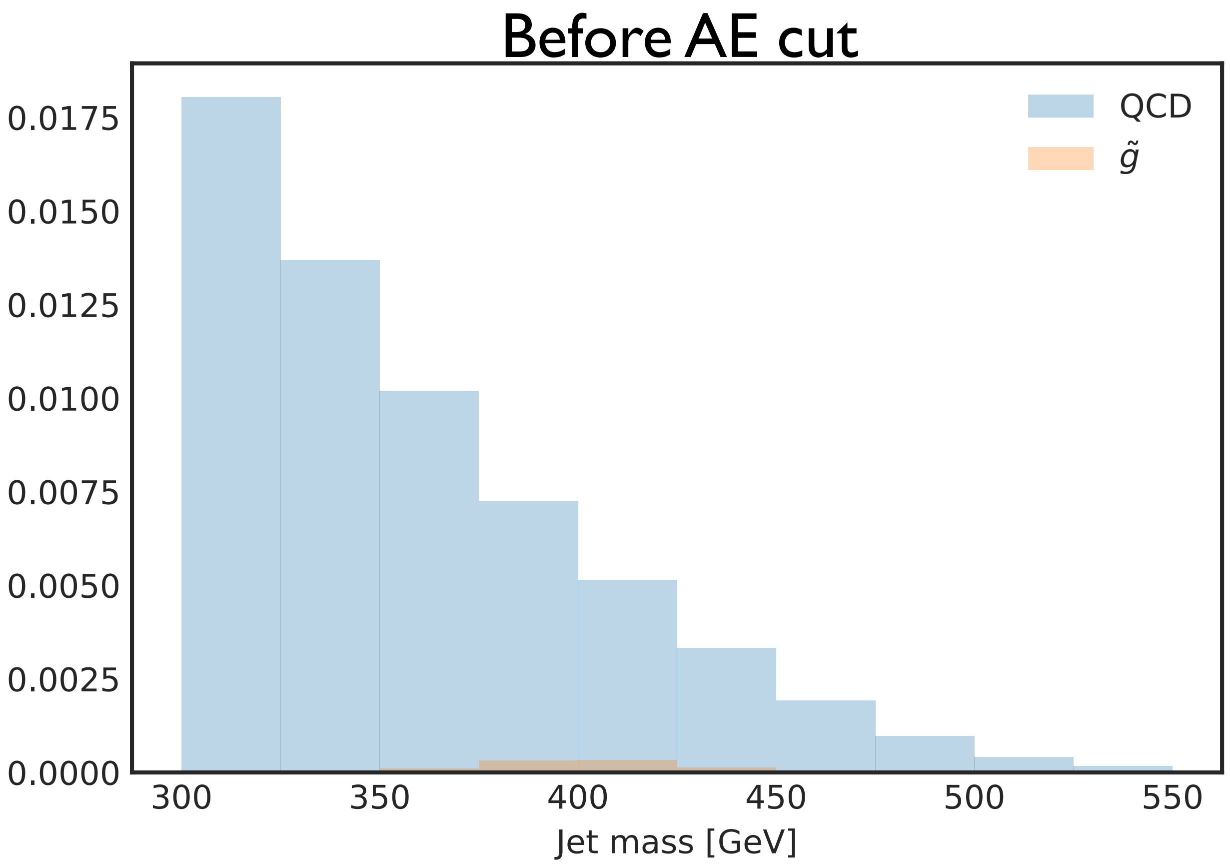
Performance of AE robust even up to 10% contamination!

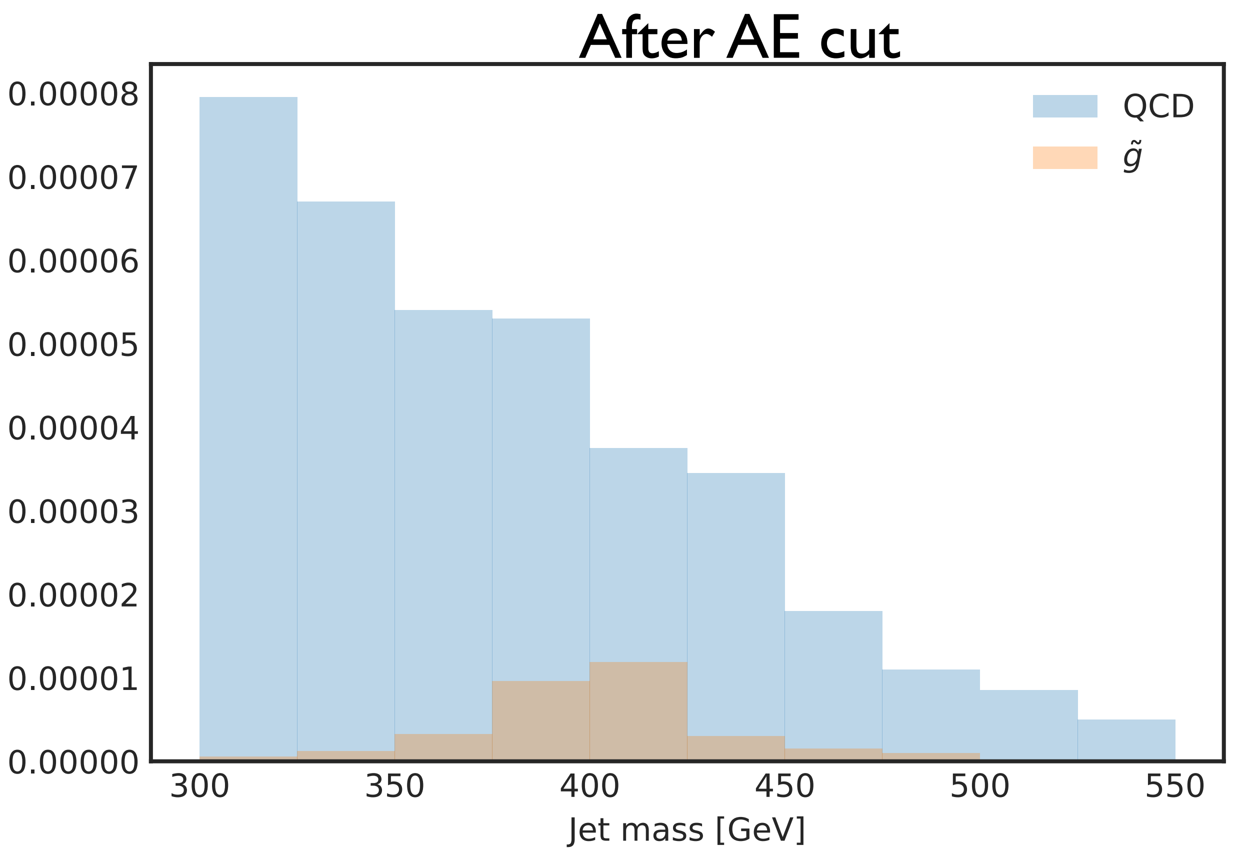
Autoencoder with explicit decorrelation

Heimel et al 1808.08979



Bump hunt with autoencoder





Train directly on data that contains 400 GeV gluinos.

Use the AE to clean away QCD jets.

Enhance the significance of the bump hunt! (improve S/B by factor of ~6)

Could really discover new physics this way!