

FYS9555 Final project

«Higgs boson production mechanism classification from ATLAS Open Data»

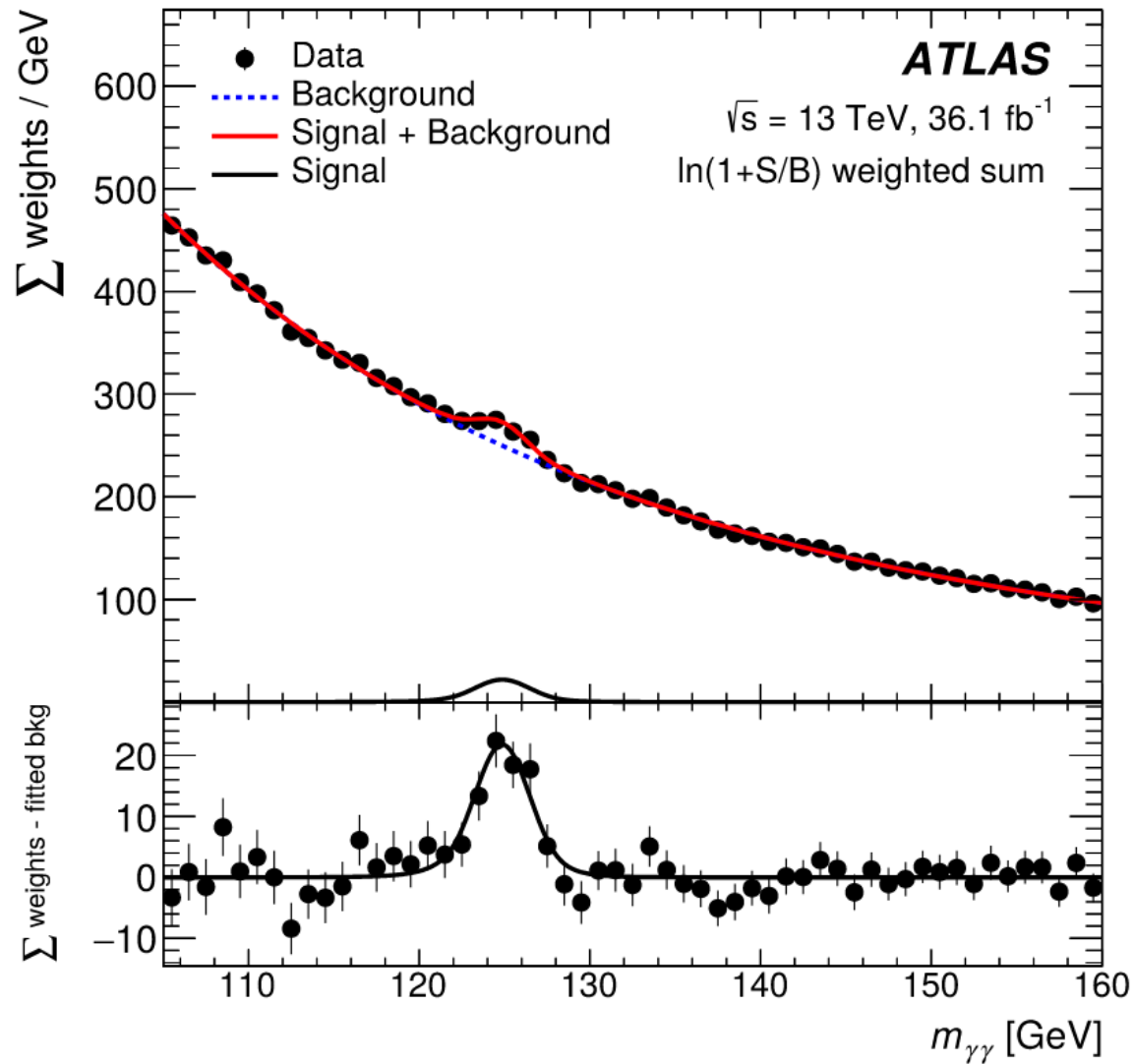
10 June 2020

Victor Ananyev

Outline

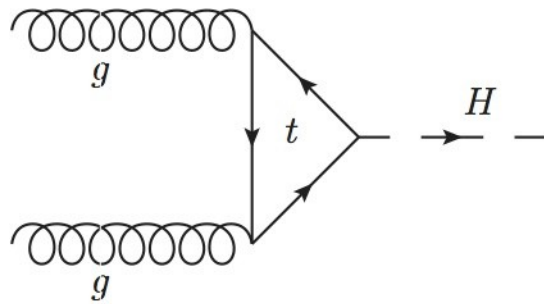
- Di-photon process
- Data acquisition
- What's inside?
- Dealing with jagged data in ML
- Fighting imbalanced data sets
- Fitting baseline ML models

Di-photon process

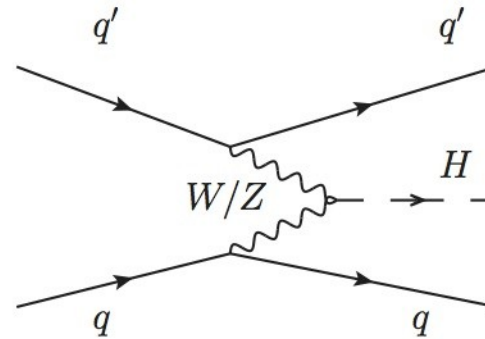


(borrowed from <https://arxiv.org/abs/1806.00242>)

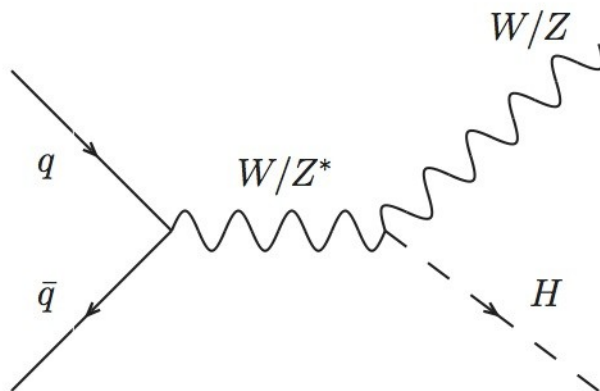
Major production mechanisms



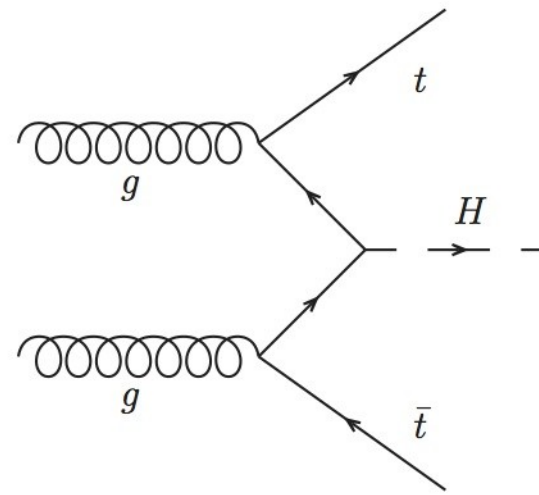
a)



b)



c)

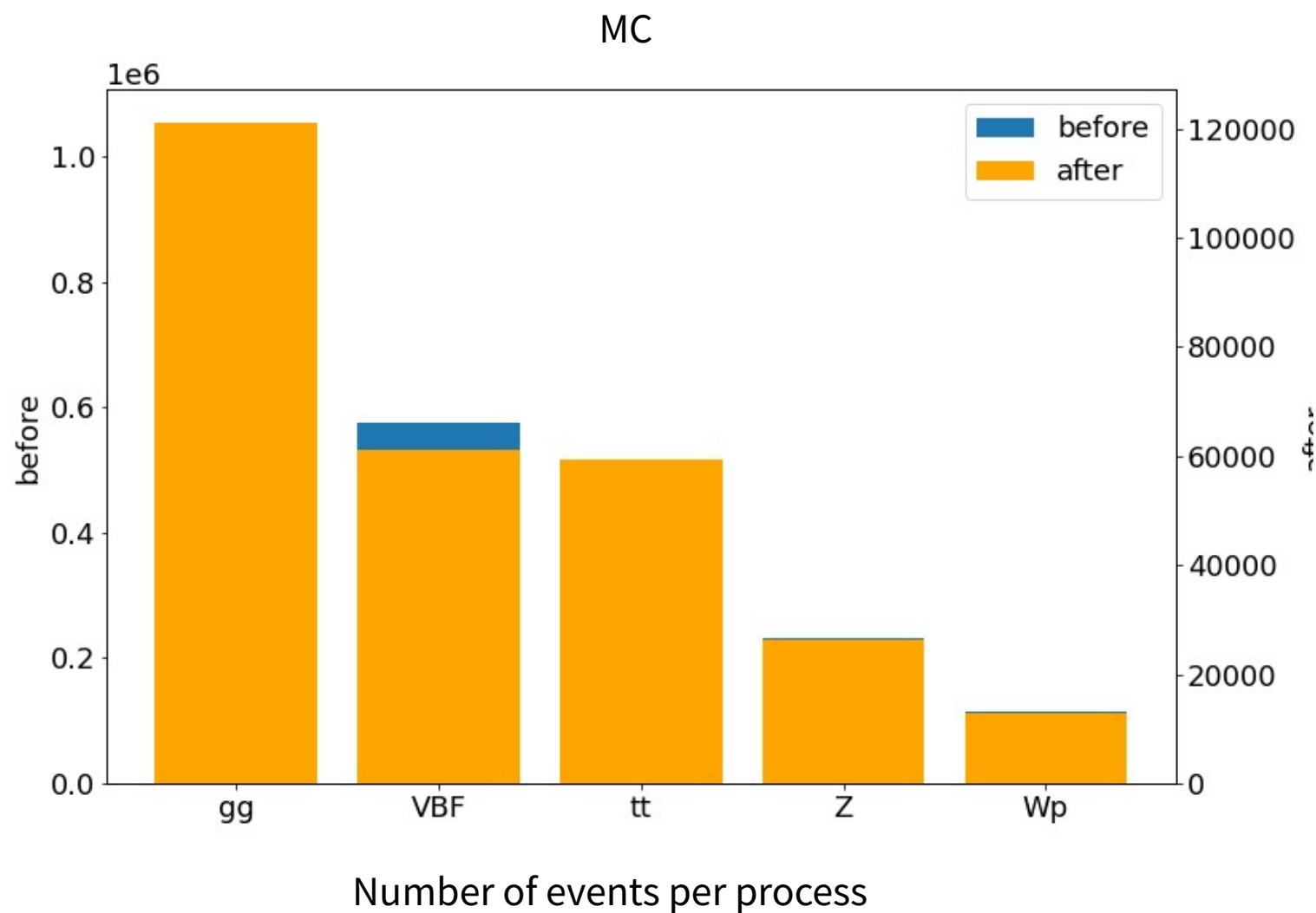


d)

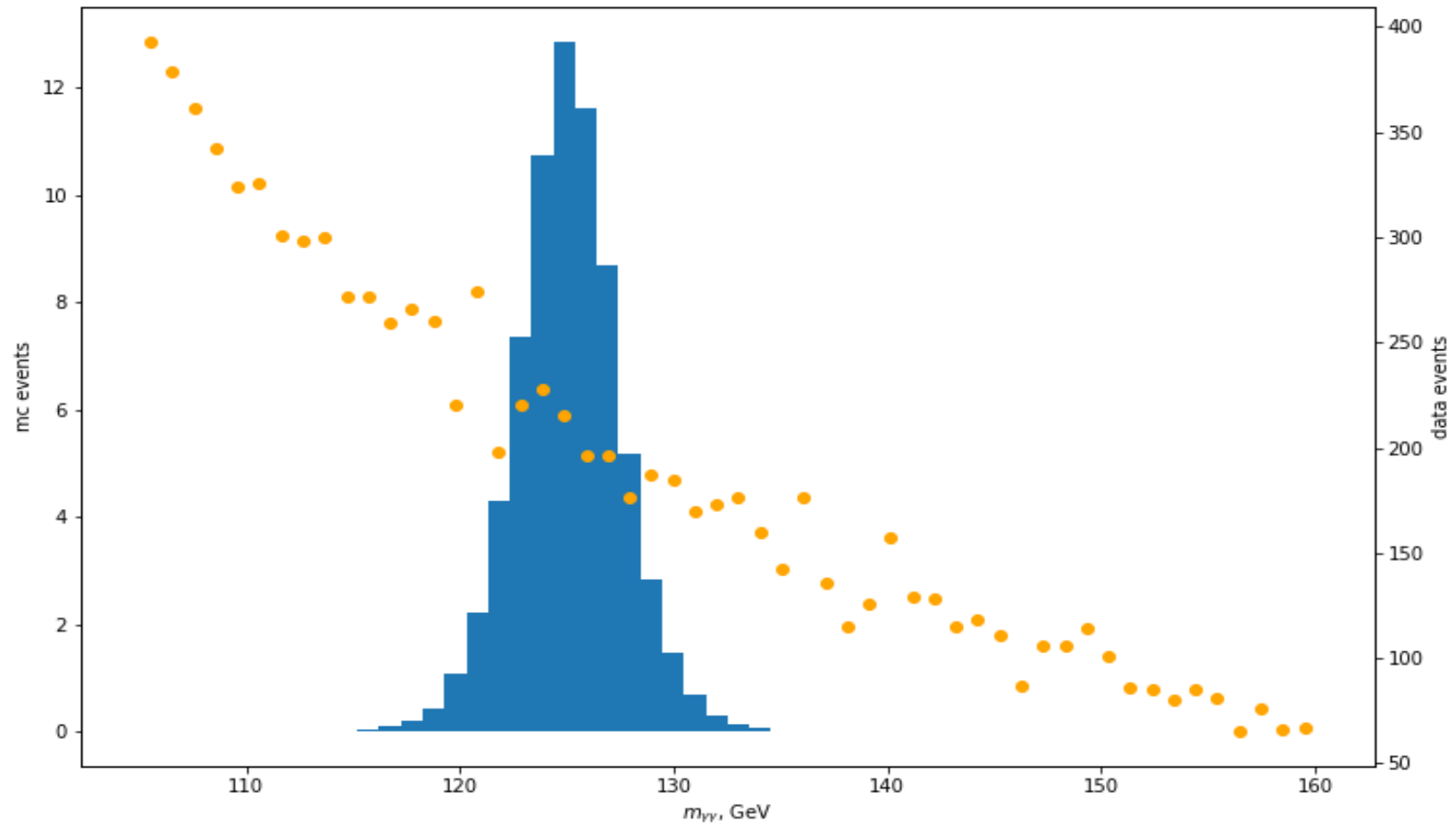
Event selection for $H\gamma\gamma$ peak

- Diphoton trigger is satisfied;
- Exactly two photons with $E_T > 35$ and 25 GeV, respectively;
- Leading and subleading photon candidates are respectively required to have $E_T/m_{\gamma\gamma} > 0.35$ and 0.25;
- Diphoton invariant mass $m_{\gamma\gamma}$ between 105 GeV and 160 GeV.

Selection efficiency ~ 10%

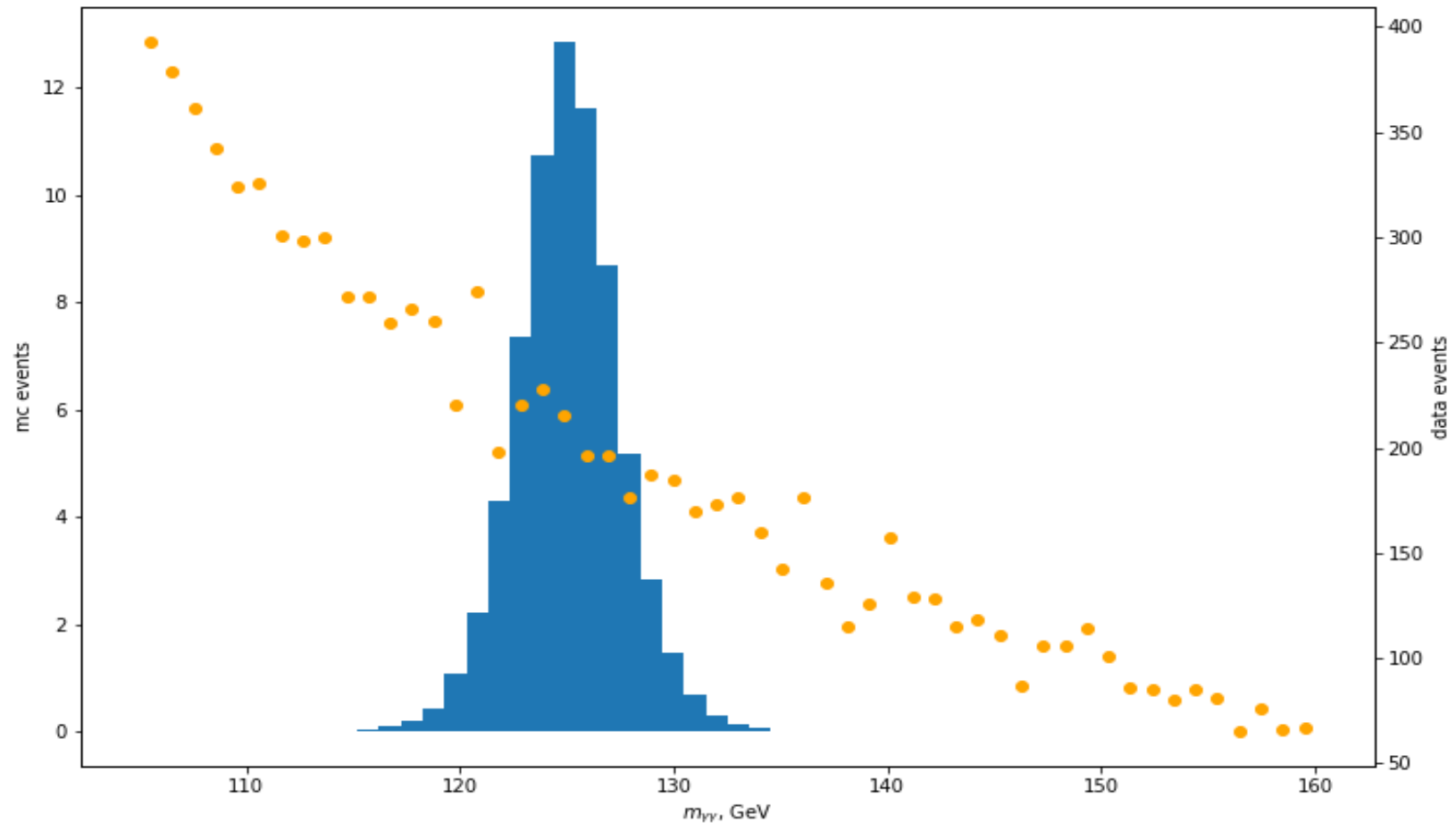


Higgs mass from ATLAS Open Data



8M events in 7 minutes

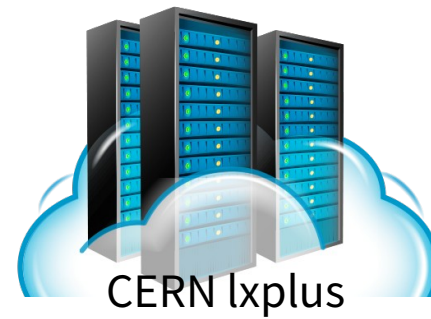
Acquire data efficiently



8M events in 7 minutes **with Python!**

Data pipeline

Processing power and data traffic



Data pipeline



1TB Root files storage



Processing power and data traffic



Data pipeline



1TB Root files storage



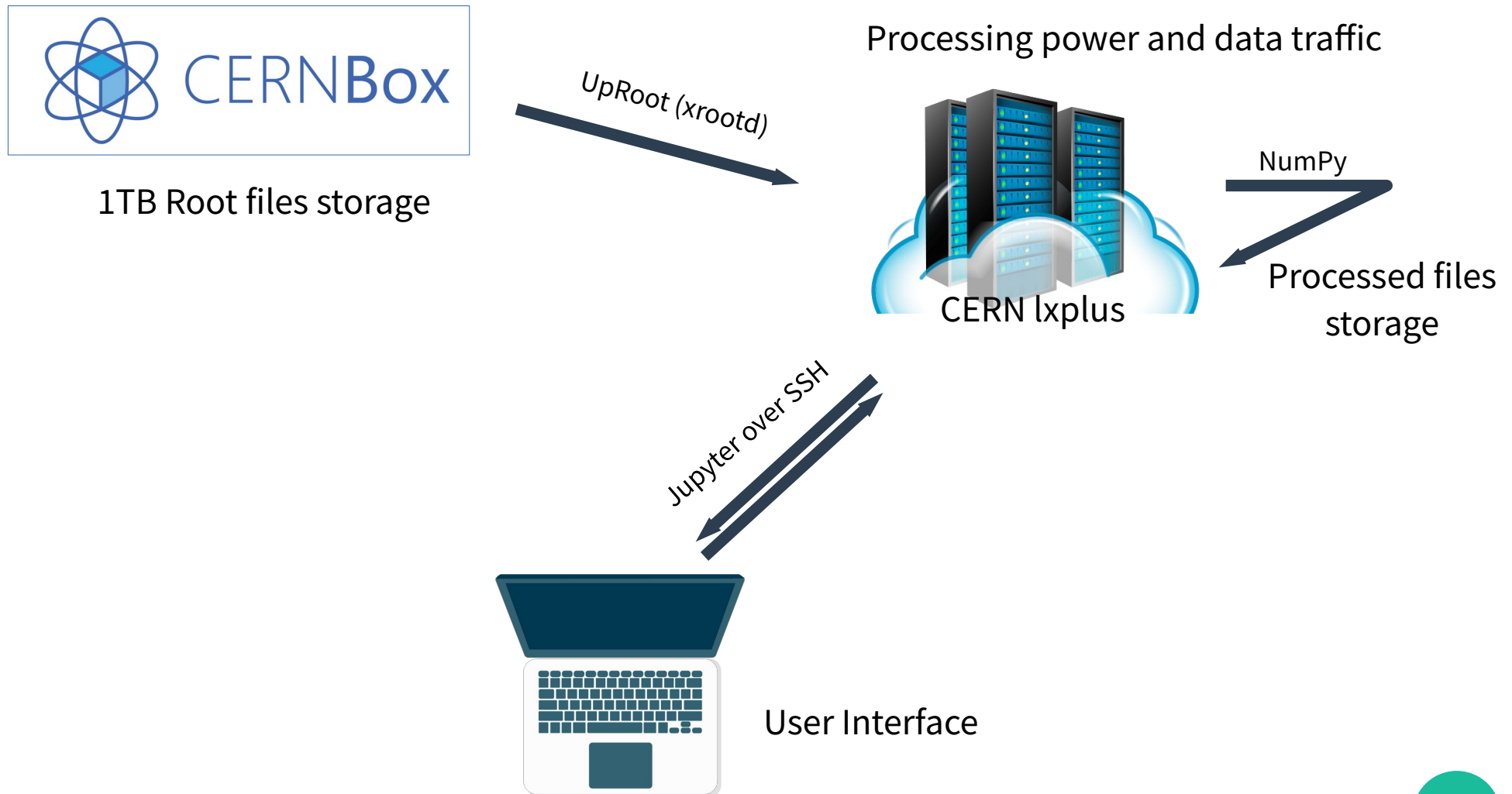
Processing power and data traffic



NumPy

Processed files
storage

Data pipeline



Data pipeline

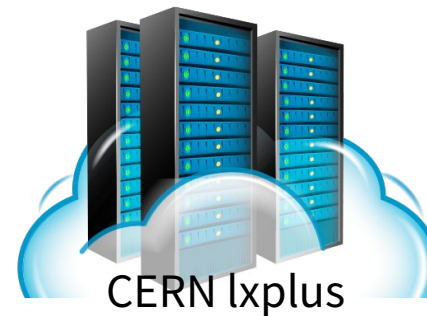
Processing power and data traffic

1TB Root files storage

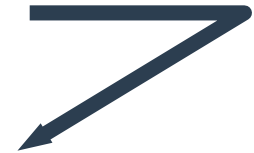


2GB – 100 GB

UpRoot (xrootd)



NumPy



Batches 200MB
~ 100 000 events

Store processed
~ 2MB
~1000 events/batch

What's inside?

- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

Macro features
1 number per event

- | | | | | |
|-----------------------|----------------------------|--------------------|--------------------|--------------------------|
| • photon_n | • lep_n | • tau_n | • jet_n | • largeRjet_n |
| • photon_truthMatched | • lep_truthMatched | • tau_pt | • jet_pt | • largeRjet_pt |
| • photon_trigMatched | • lep_trigMatched | • tau_eta | • jet_eta | • largeRjet_eta |
| • photon_pt | • lep_pt | • tau_phi | • jet_phi | • largeRjet_phi |
| • photon_eta | • lep_eta | • tau_E | • jet_E | • largeRjet_E |
| • photon_phi | • lep_phi | • tau_isTightID | • jet_jvt | • largeRjet_m |
| • photon_E | • lep_E | • tau_truthMatched | • jet_trueflav | • largeRjet_truthMatched |
| • photon_isTightID | • lep_z0 | • tau_trigMatched | • jet_truthMatched | • largeRjet_D2 |
| • photon_ptcone30 | • lep_charge | • tau_nTracks | • jet_MV2c10 | • largeRjet_tau32 |
| • photon_etcone20 | • lep_type | • tau_BDTid | • jet_pt_syst | • largeRjet_pt_syst |
| • photon_convType | • lep_isTightID | • tau_pt_syst | | |
| • photon_pt_syst | • lep_ptcone30 | • tau_charge | | |
| | • lep_etcone20 | | | |
| | • lep_trackd0pvunbiased | | | |
| | • lep_tracksigd0pvunbiased | | | |
| | • lep_pt_syst | | | |

Micro features
*Jagged. List of numbers
per event*

Mask bubbling for feature extraction

- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

Macro features

[0, 0, 1, 1, 0, 1 ...]

MASK

1000 events

800

**Photon
Features**

- | | | | | |
|-----------------------|----------------------------|--------------------|--------------------|--------------------------|
| • photon_n | • lep_n | • tau_n | • jet_n | • largeRjet_n |
| • photon_truthMatched | • lep_truthMatched | • tau_pt | • jet_pt | • largeRjet_pt |
| • photon_trigMatched | • lep_trigMatched | • tau_eta | • jet_eta | • largeRjet_eta |
| • photon_pt | • lep_pt | • tau_phi | • jet_phi | • largeRjet_phi |
| • photon_eta | • lep_eta | • tau_E | • jet_E | • largeRjet_E |
| • photon_phi | • lep_phi | • tau_isTightID | • jet_jvt | • largeRjet_m |
| • photon_E | • lep_E | • tau_truthMatched | • jet_trueflav | • largeRjet_truthMatched |
| • photon_isTightID | • lep_z0 | • tau_trigMatched | • jet_truthMatched | • largeRjet_D2 |
| • photon_cone30 | • lep_charge | • tau_nTracks | • jet_MV2c10 | • largeRjet_tau32 |
| • photon_cone20 | • lep_type | • tau_BDTid | • jet_pt_syst | • largeRjet_pt_syst |
| • photon_Type | • lep_isTightID | • tau_pt_syst | | |
| • photon_syst | • lep_ptcone30 | • tau_charge | | |
| | • lep_etcone20 | | | |
| | • lep_trackd0pvunbiased | | | |
| | • lep_tracksigd0pvunbiased | | | |
| | • lep_pt_syst | | | |

Micro features

Mask bubbling for feature extraction

- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

Macro features

[0, 0, 1, 1, 0, 1 ...] > [0, 0, 0, 1, 0, 1 ...]

MASK

MASK

1000 events

800

600

**Photon
Features**

**Lepton
Features**

unbiased
pvunbiased

Micro features

- photon_n
- photon_truthMatched
- photon_trigMatched
- photon_pt
- photon_eta
- photon_phi
- photon_E
- photon_isTightID
- photon_tau30
- photon_tau20
- photon_Type
- photon_syst
- lep_n
- lep_truthMatched
- lep_trigMatched
- lep_pt
- lep_eta
- lep_phi
- lep_E
- lep_z0
- lep_c
- lep_t
- lep_i
- lep_p
- lep_syst
- tau_n
- tau_pt
- tau_eta
- tau_phi
- tau_E
- tau_isTightID
- tau_truthMatched
- tau_trigMatched
- tau_nTracks
- tau_BDTid
- tau_pt_syst
- tau_charge
- jet_n
- jet_pt
- jet_eta
- jet_phi
- jet_E
- jet_jvt
- jet_trueflav
- jet_truthMatched
- jet_MV2c10
- jet_pt_syst
- largeRjet_n
- largeRjet_pt
- largeRjet_eta
- largeRjet_phi
- largeRjet_E
- largeRjet_m
- largeRjet_truthMatched
- largeRjet_D2
- largeRjet_tau32
- largeRjet_pt_syst

Mask bubbling for feature extraction

- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

Macro features

$[0, 0, 1, 1, 0, 1 \dots] > [0, 0, 0, 1, 0, 1 \dots] > [0, 0, 0, 1, 0, 0 \dots]$

MASK

MASK

MASK

1000 events

250 events

800

600

300

Photon Features

Lepton Features

Tau Features

Micro features

Mask bubbling for feature extraction

- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

Macro features

$[0, 0, 1, 1, 0, 1 \dots] > [0, 0, 0, 1, 0, 1 \dots] > [0, 0, 0, 1, 0, 0 \dots]$

MASK

MASK

MASK

800

600

250

Photon Features

Lepton Features

Tau Feature

largeRjet_n
largeRjet_pt
largeRjet_eta
largeRjet_phi
largeRjet_E
largeRjet_m
largeRjet_truthMatched
largeRjet_D2
largeRjet_tau32
largeRjet_pt_syst

Micro features

250 events

MASK

Mask bubbling for feature extraction

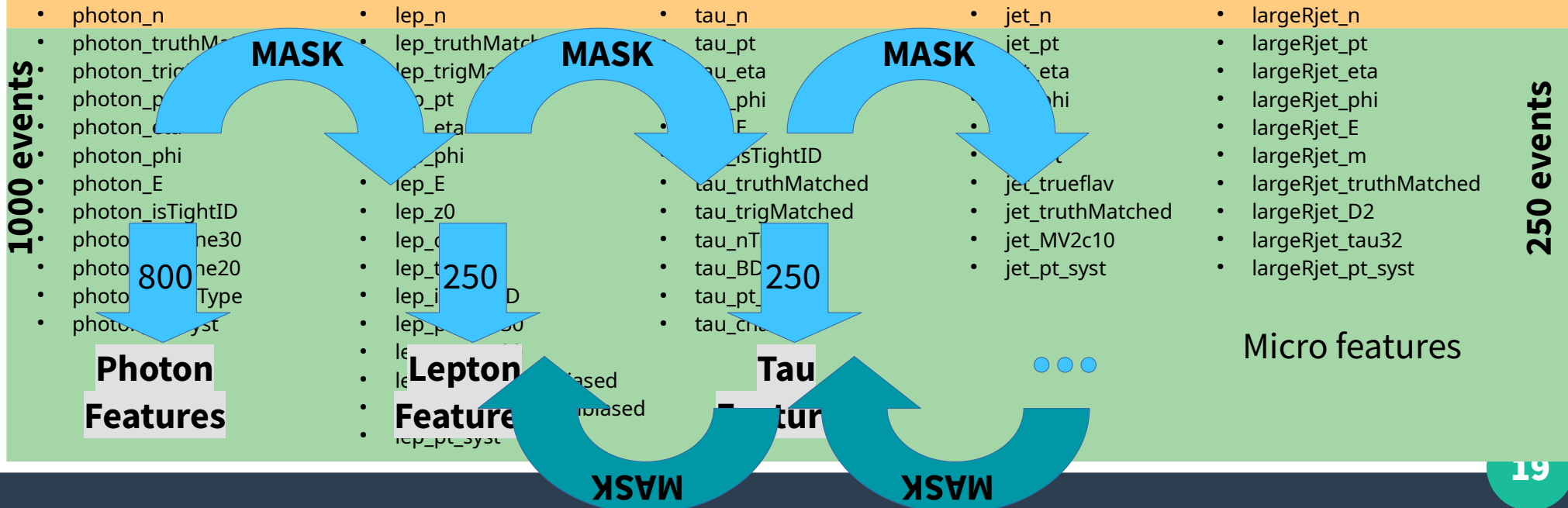
- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

Macro features

[0, 0, 1, 1, 0, 1 ...] > [0, 0, 0, 1, 0, 1 ...] > [0, 0, 0, 1, 0, 0 ...]



Mask bubbling for feature extraction

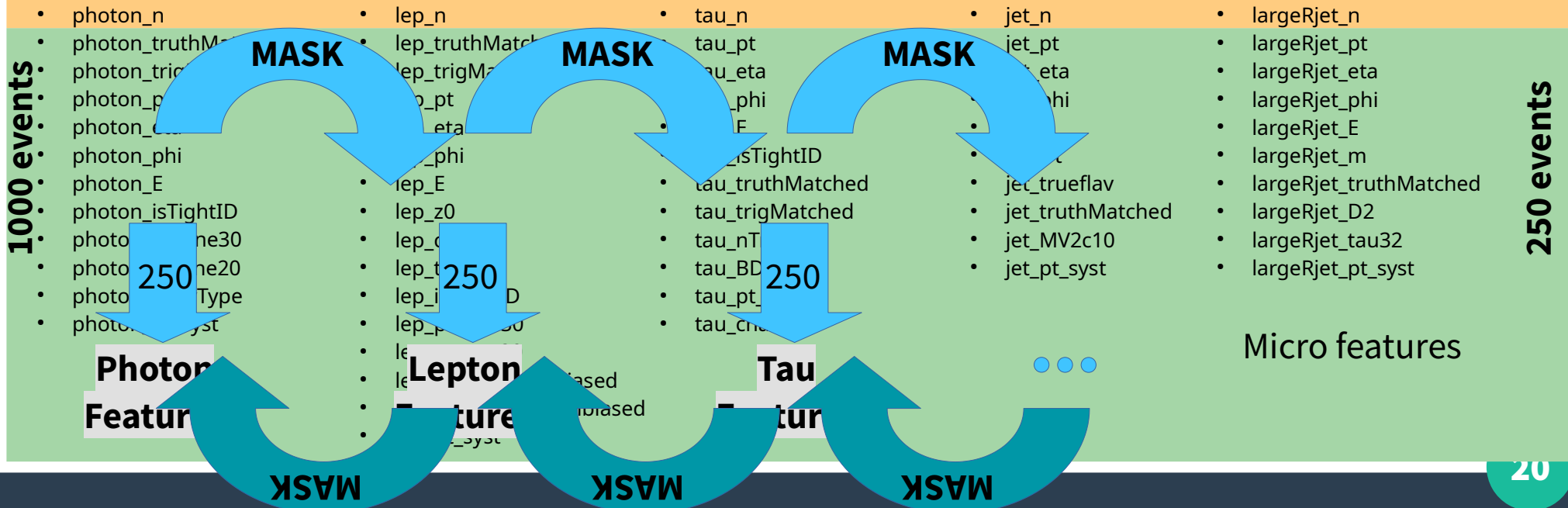
- XSection
- SumWeights

Simulation wide

- runNumber
- eventNumber
- channelNumber
- mcWeight
- scaleFactor_*
- trig*
- met_et
- met_et_syst
- met_phi
- ditau_m

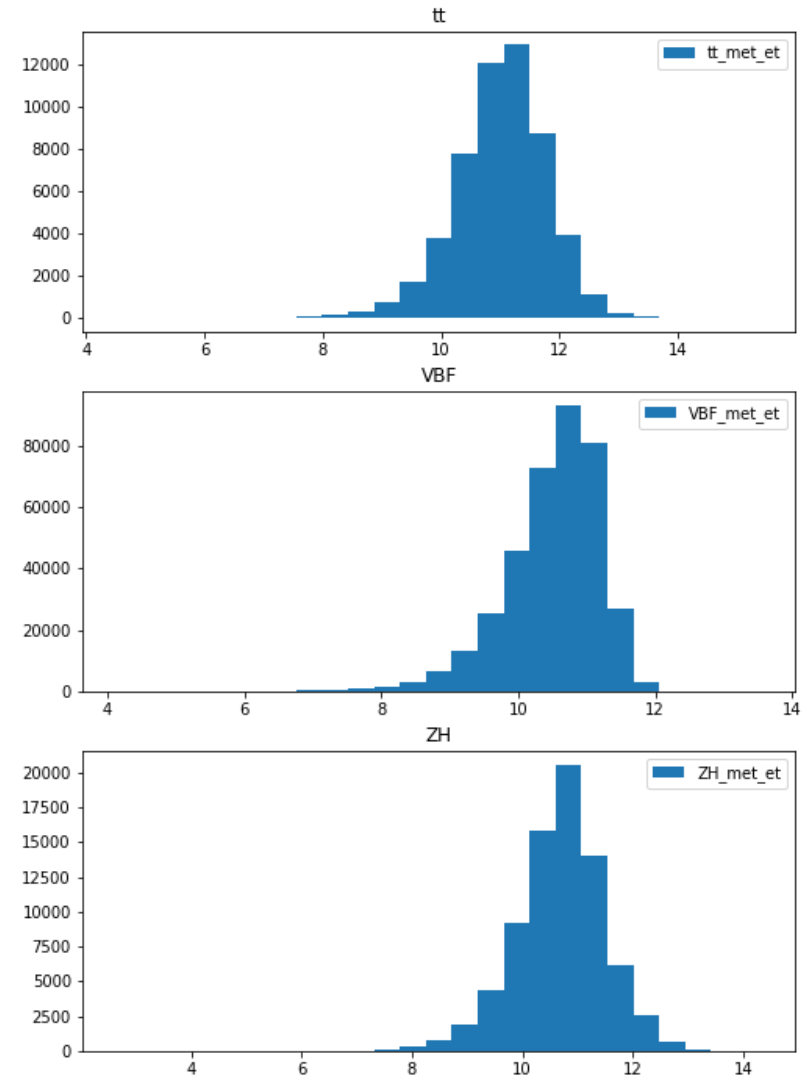
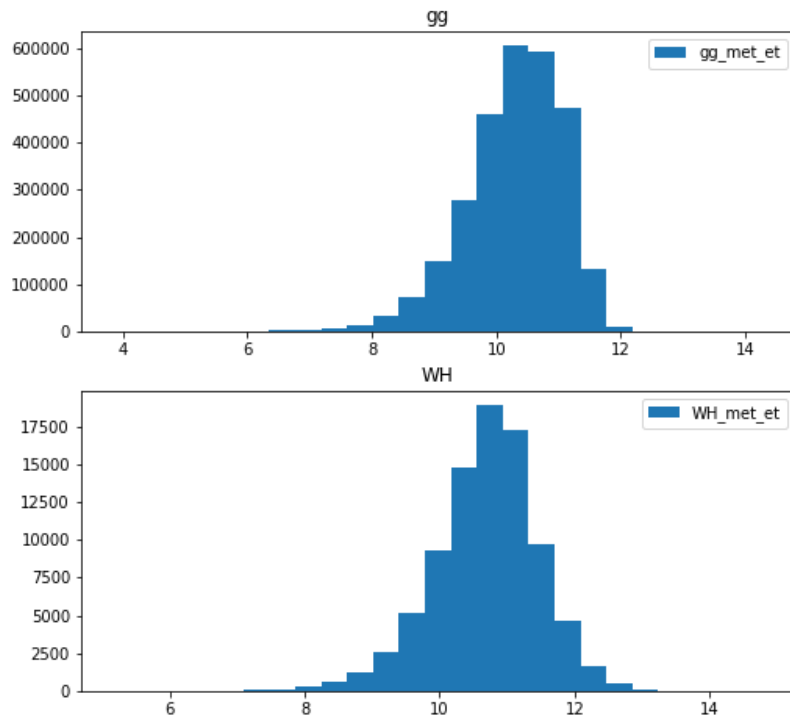
Macro features

[0, 0, 1, 1, 0, 1 ...] > [0, 0, 0, 1, 0, 1 ...] > [0, 0, 0, 1, 0, 0 ...]



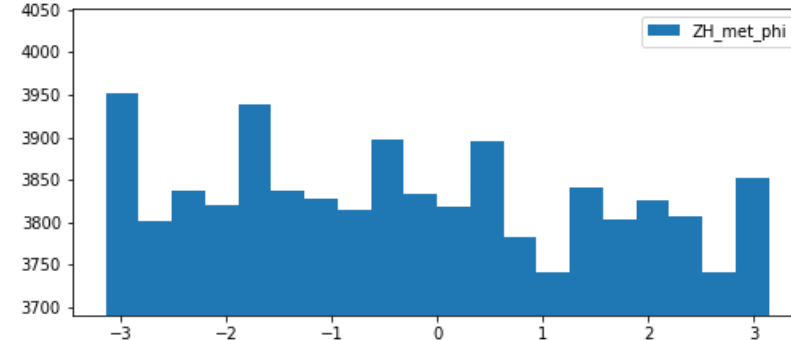
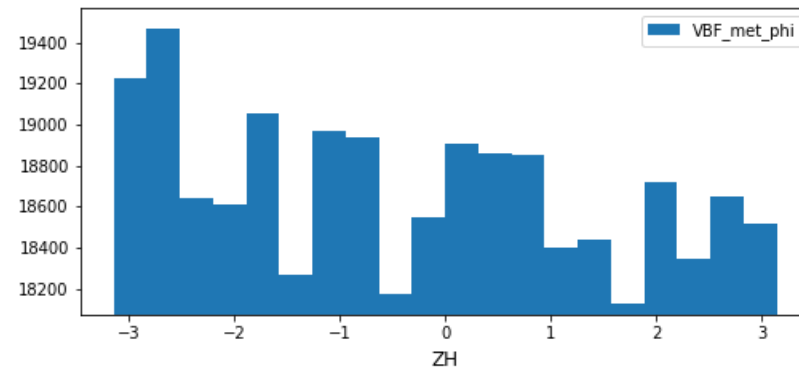
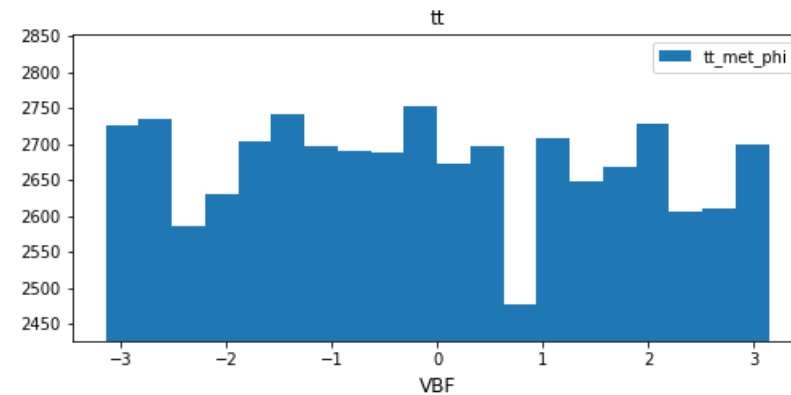
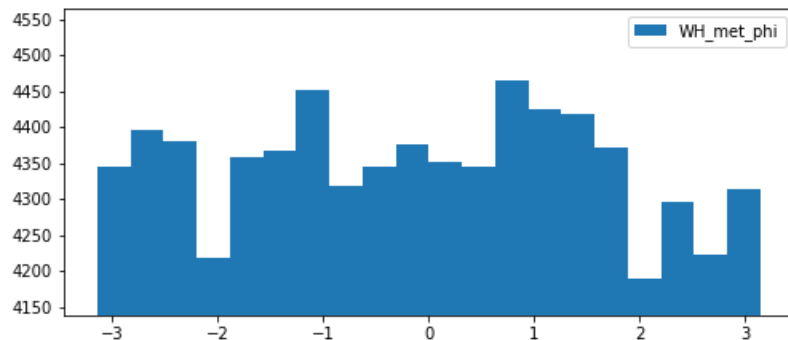
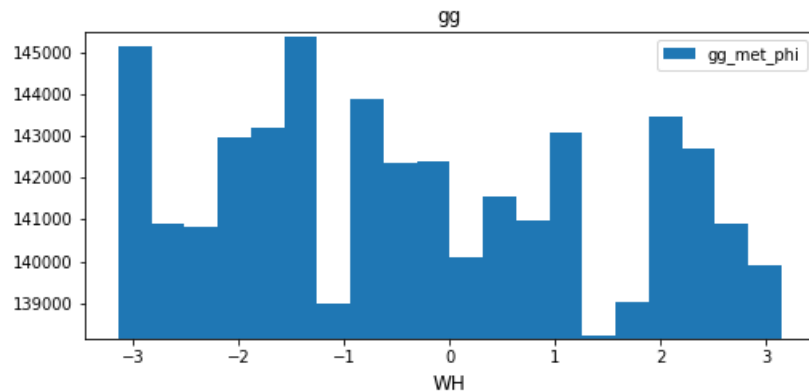
Missing energy E_t (log x scale)

Within 100 000 events batch



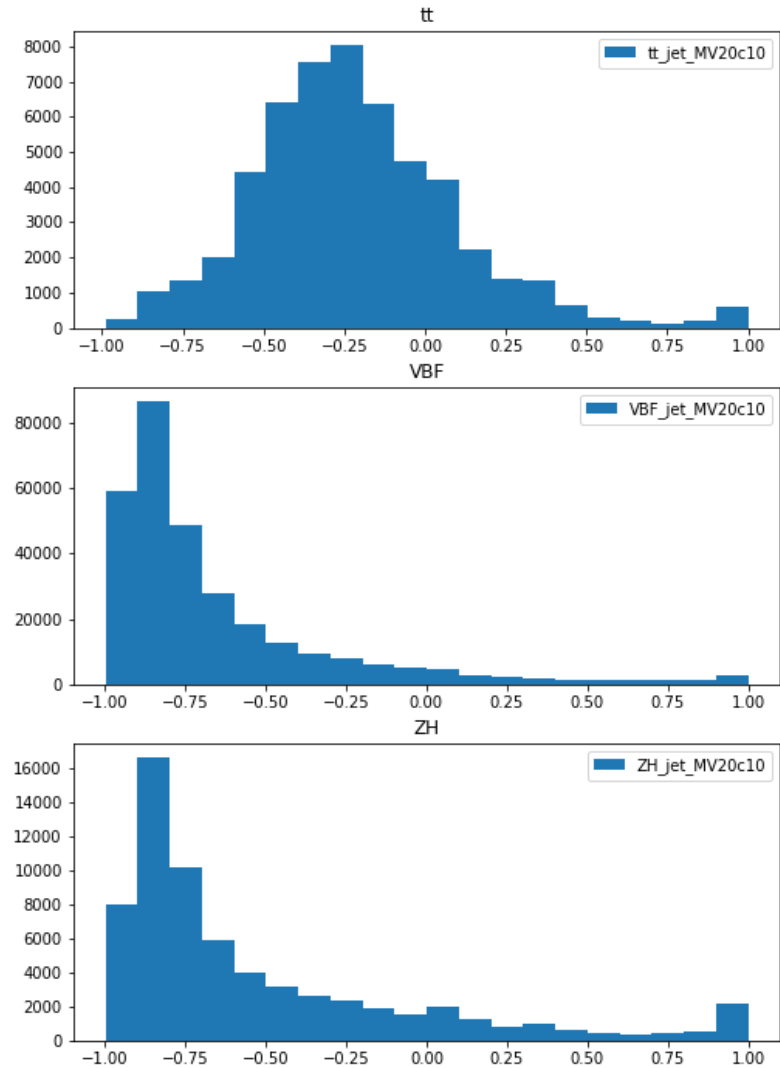
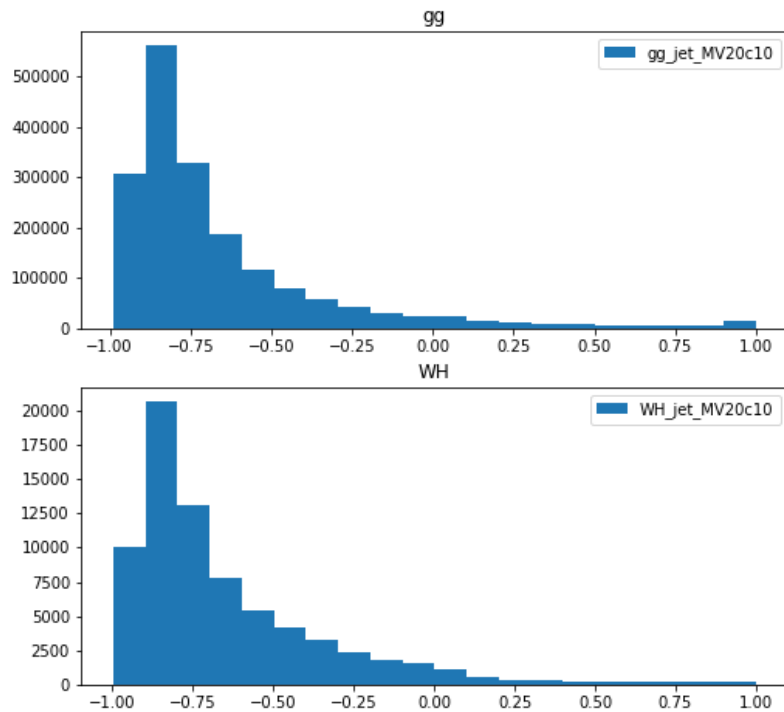
Missing energy Phi

Within 100 000 events batch



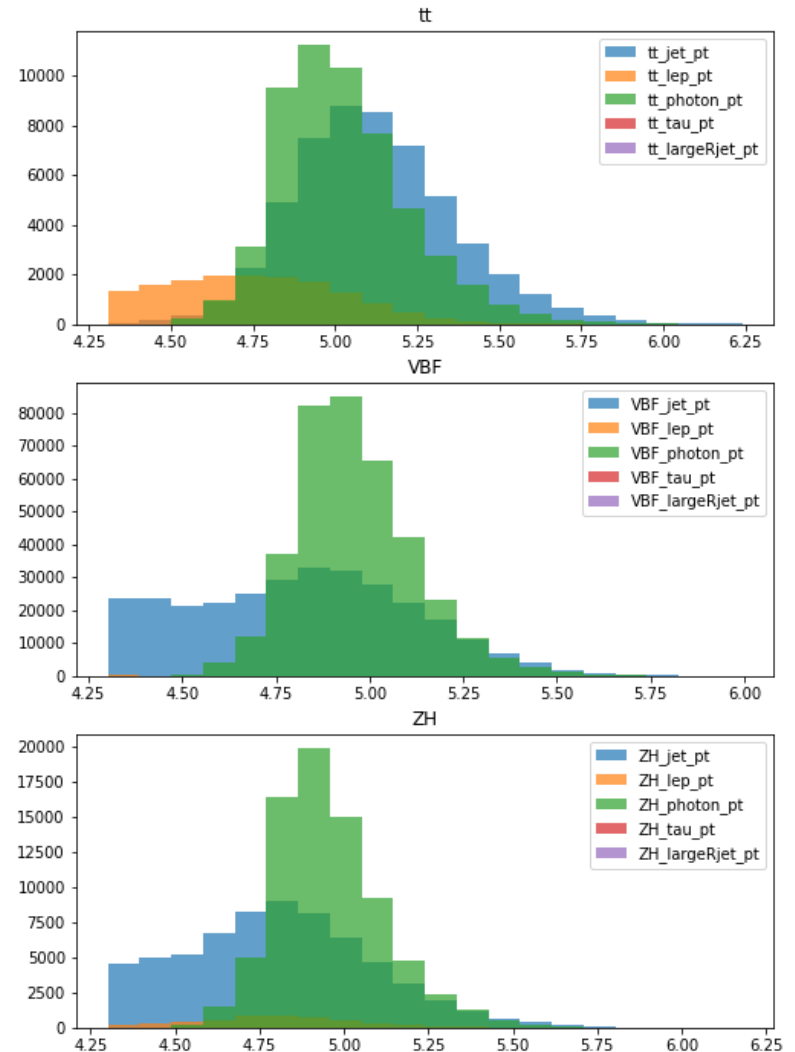
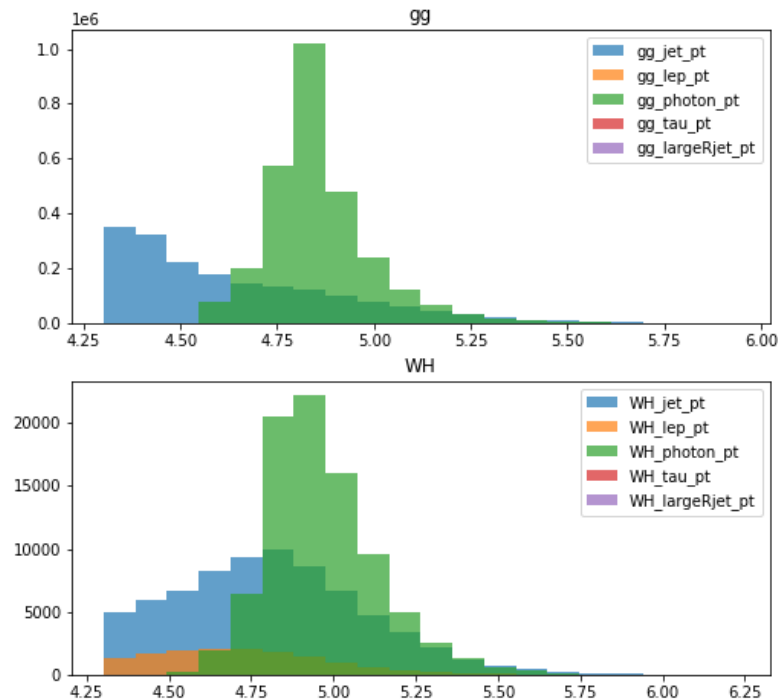
Jet b-tagging score

Within 100 000 events batch



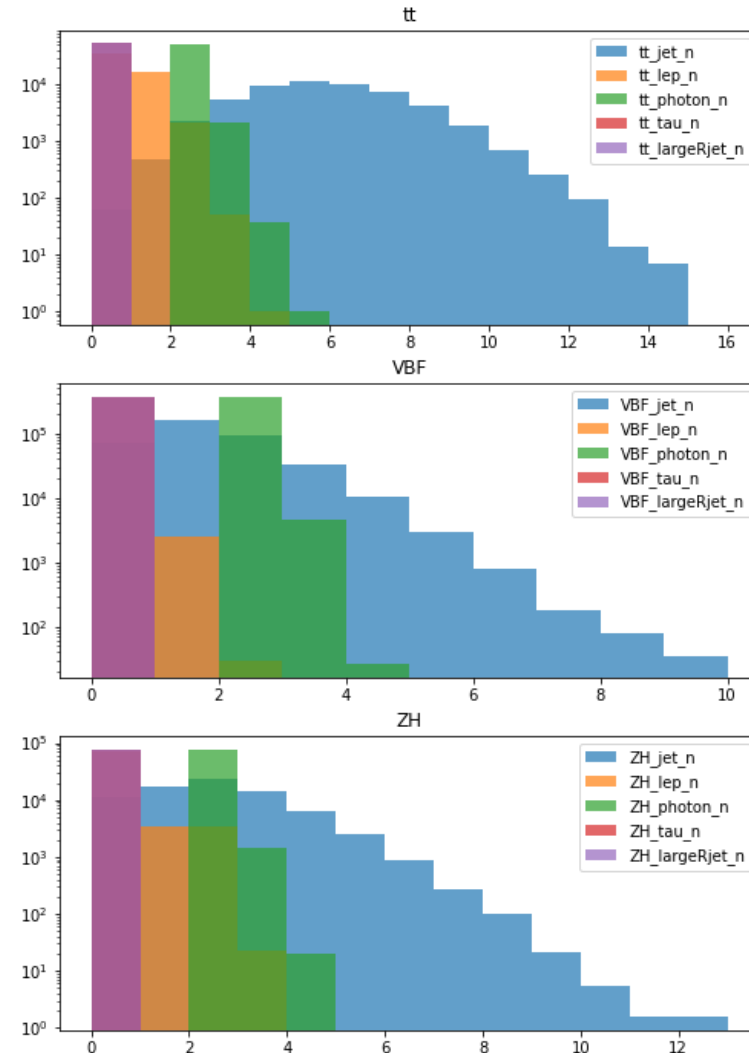
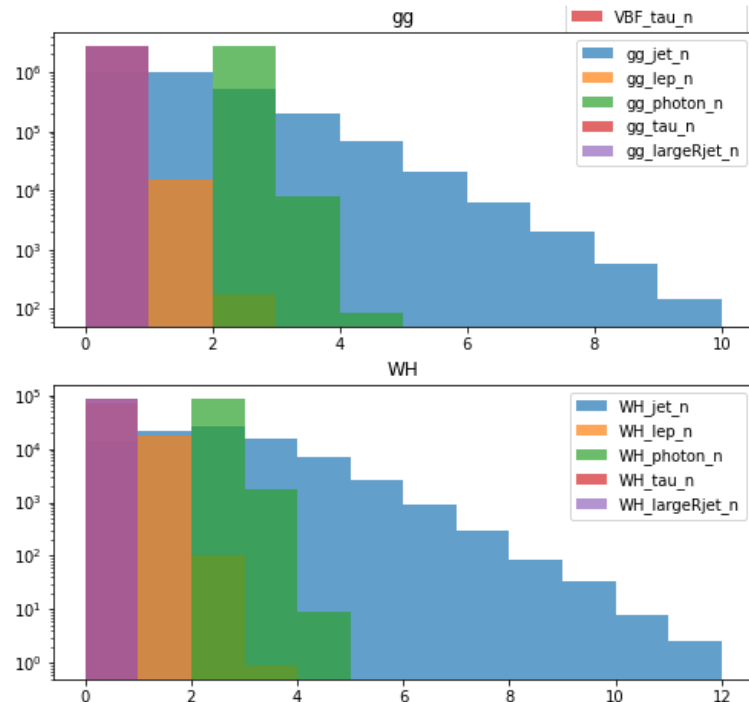
Max pT per particle type (log x scale)

Within 100 000 events batch

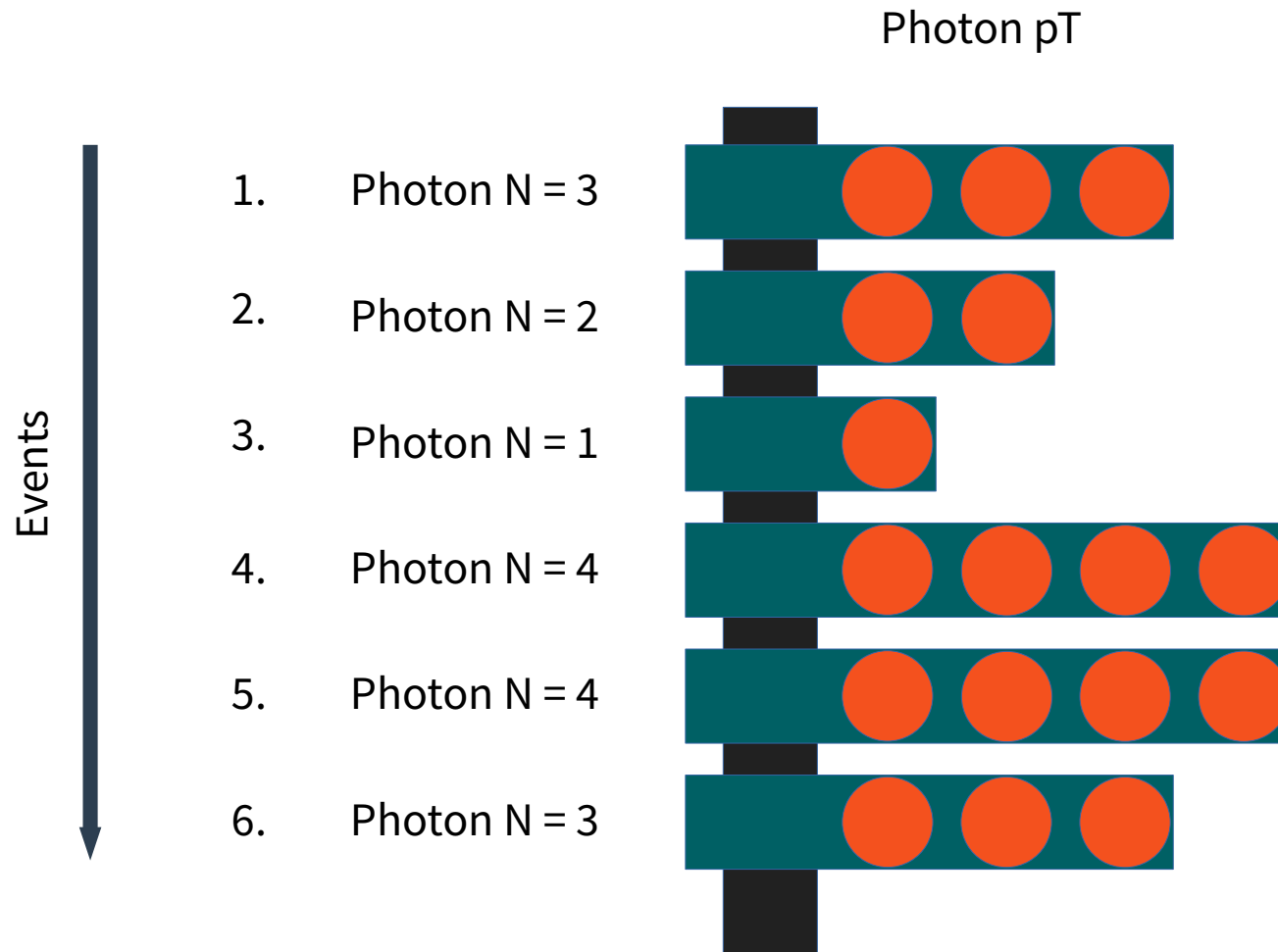


Number of particles (log y scale)

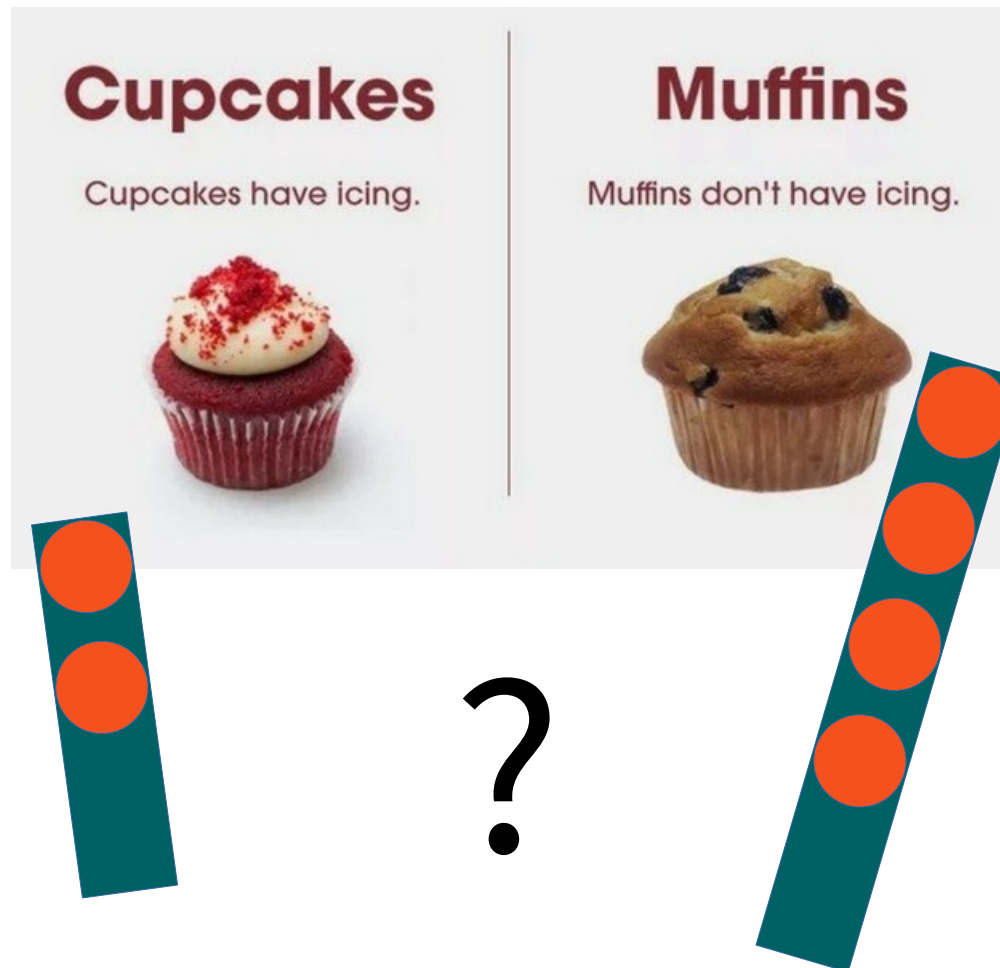
Within 100 000 events batch



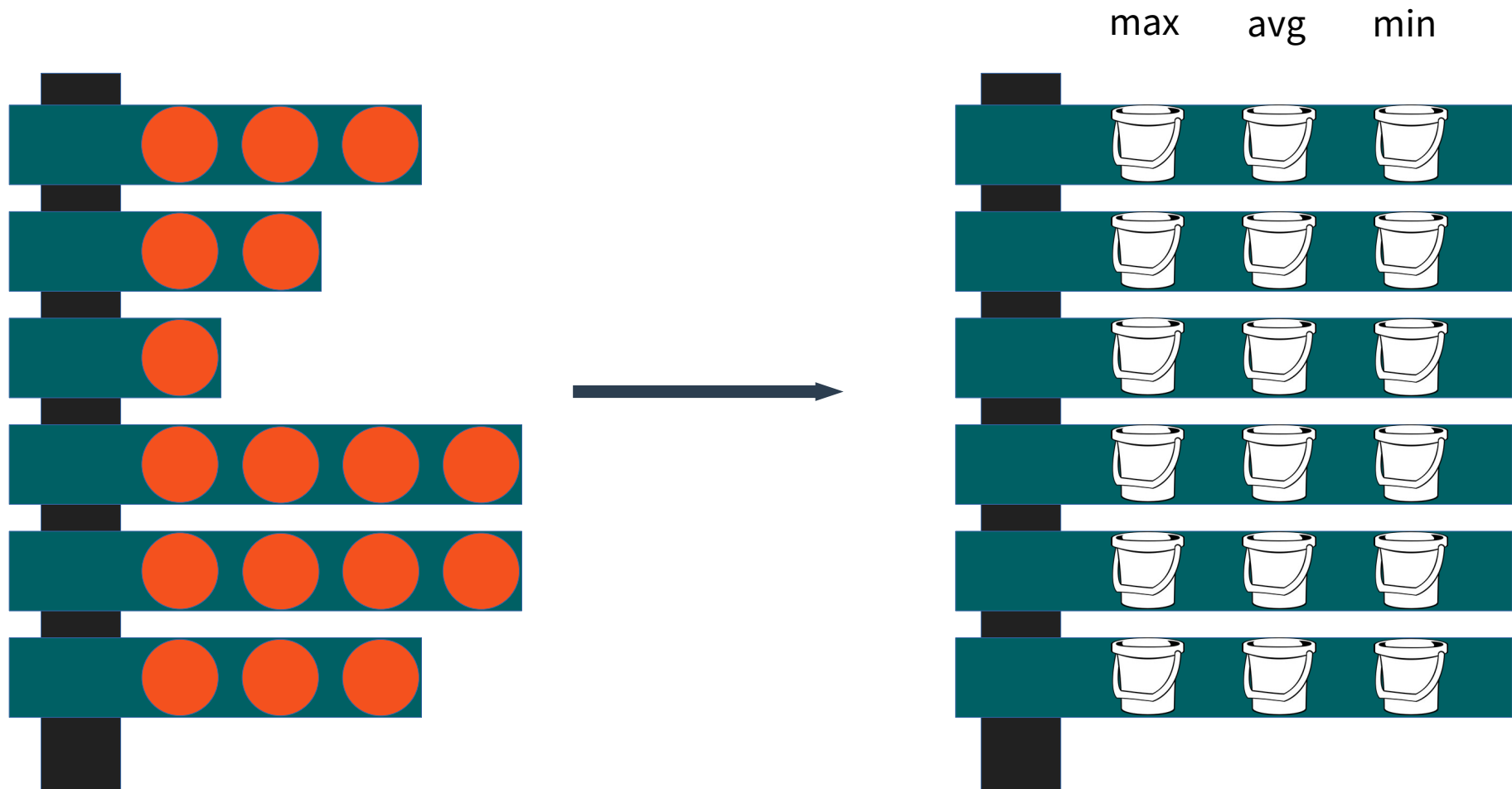
Jagged data. Structure of micro features



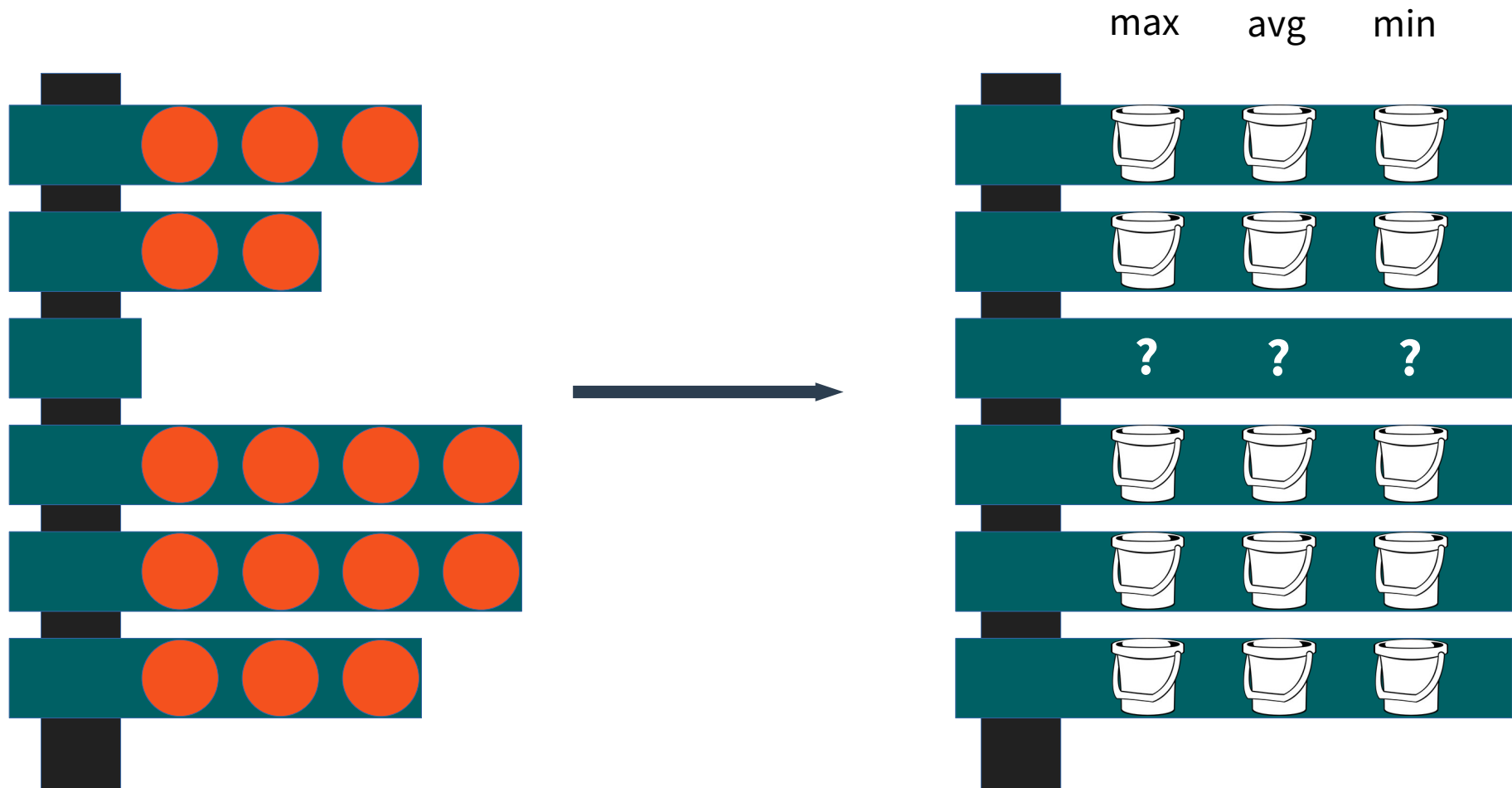
How similar are them?



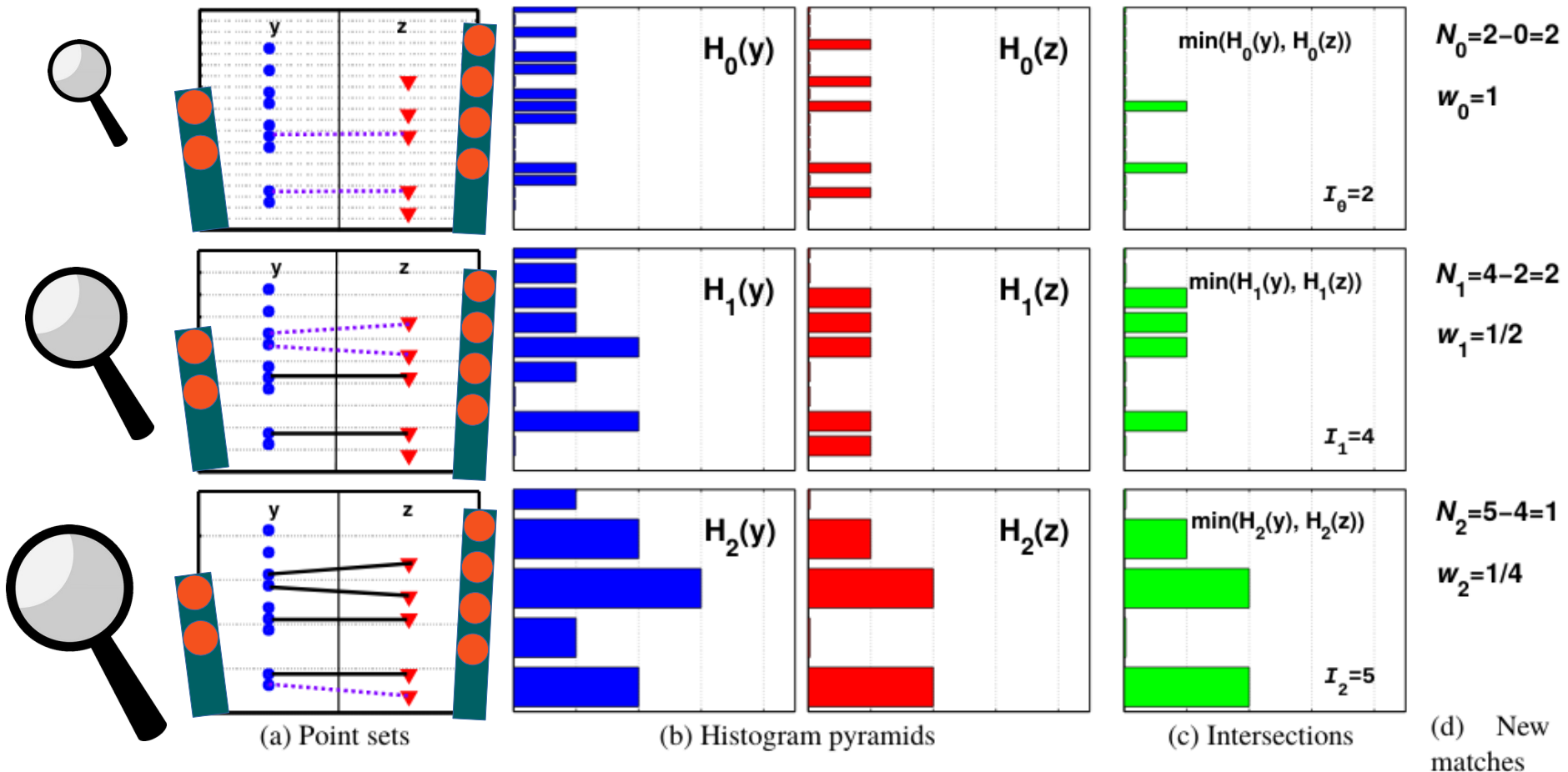
Option: Aggregation



Option: Aggregation



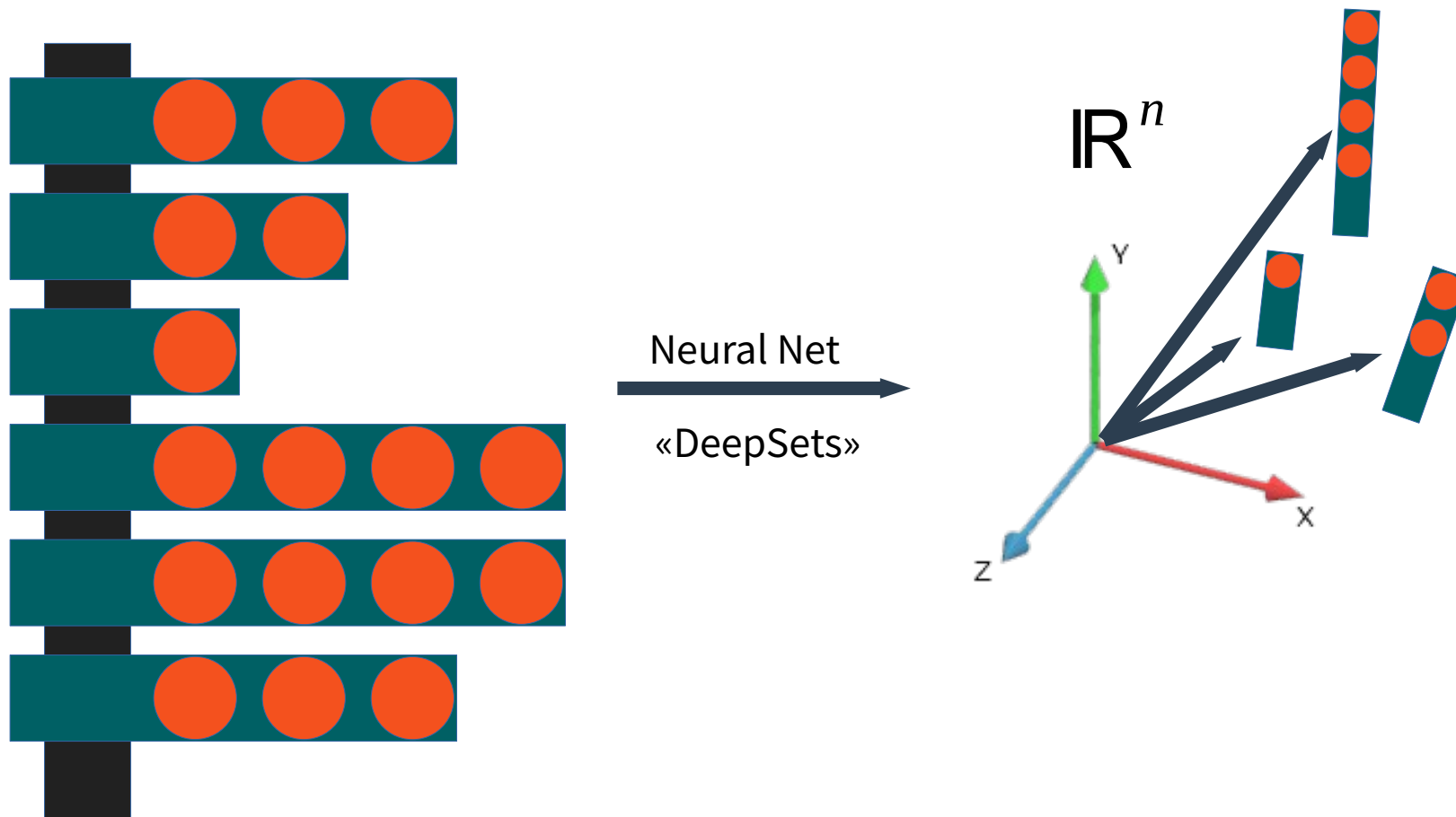
Option: Set Kernels (Pyramid match)



Option: Set Kernels

- + Operation in non-linearly transformed feature space at no cost (Kernel trick)
 - Interpretable. Not a black box.
- Compute and store $N \times N$ matrix of distances (N – number of training instances)
 - N^2 memory consumption
 - Incremental learning is not easily achievable. 1 extra training point triggers N calls to kernel.

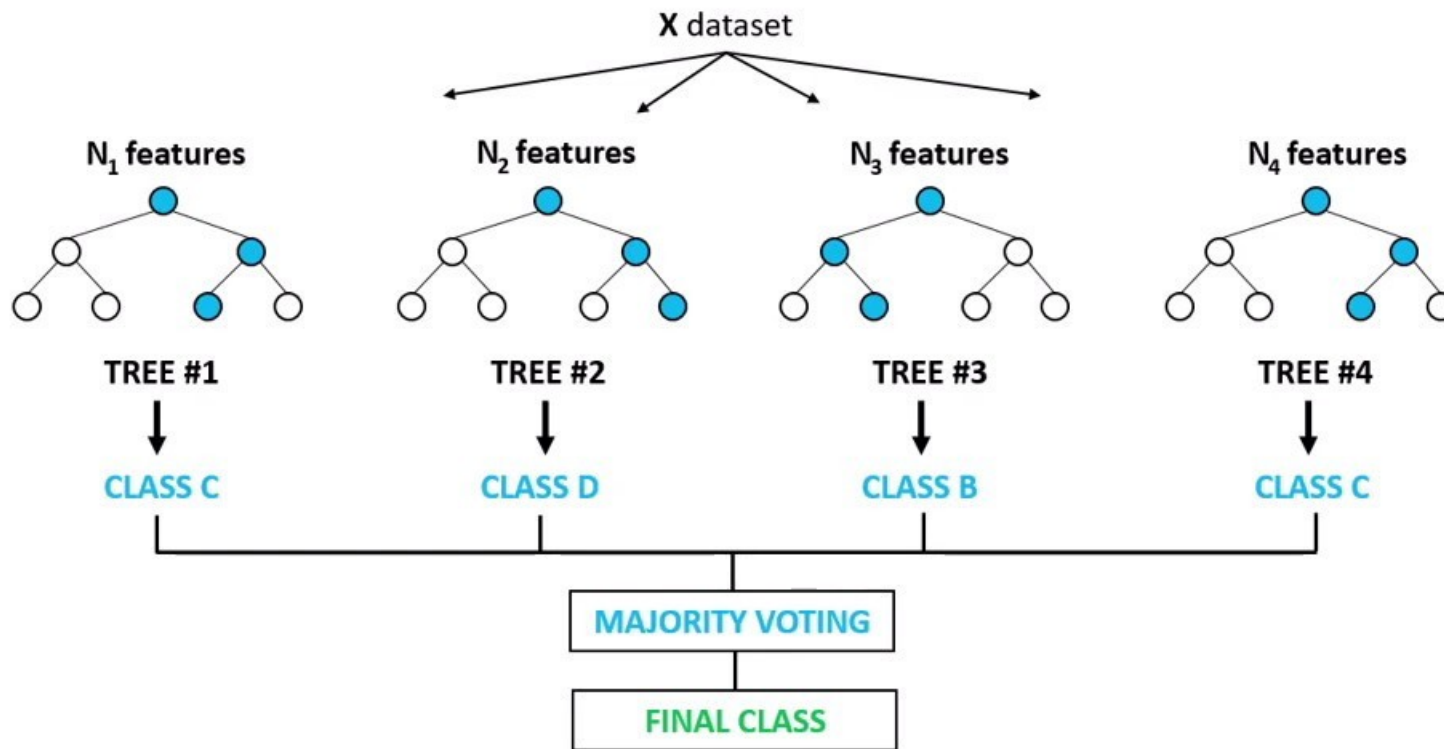
Option: Embedding with NN «DeepSets»



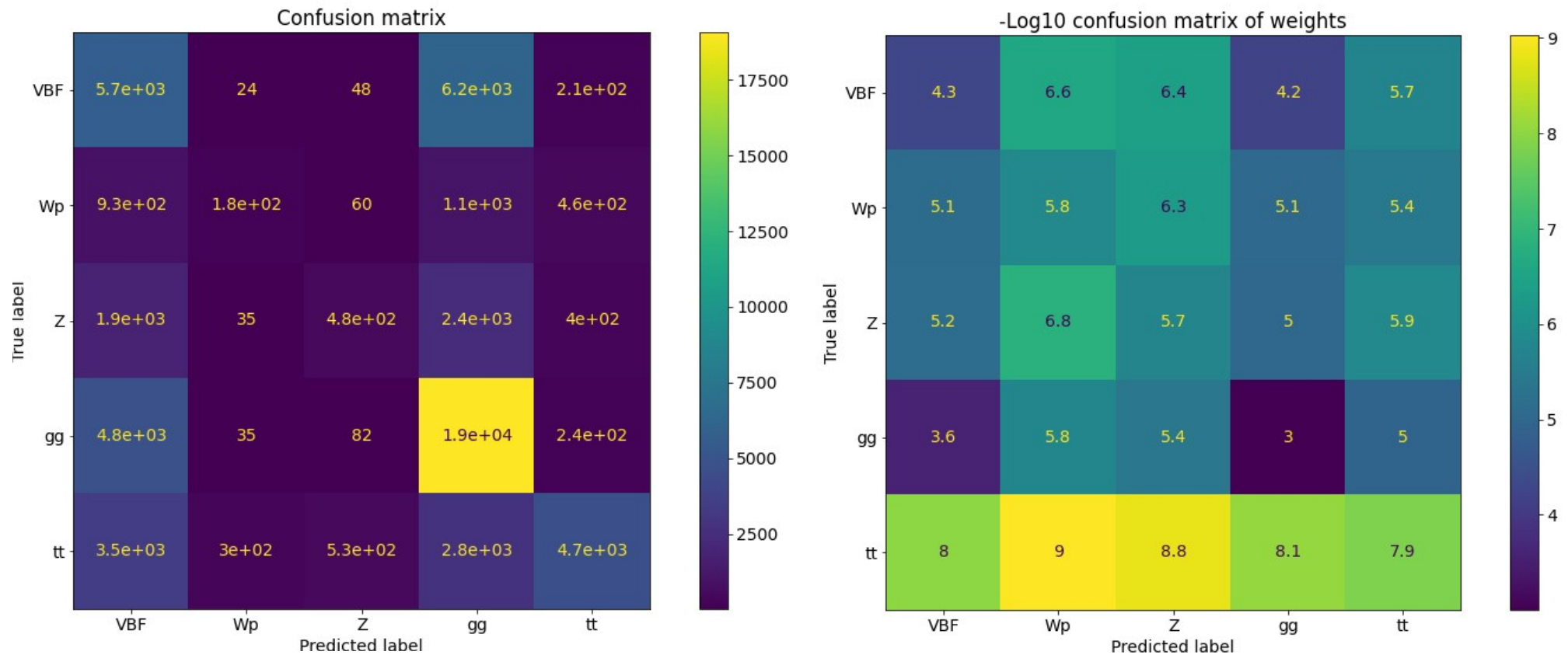
Option: Embedding with NN «DeepSets»

- + Order independence. Treat set of particles as a *set*
 - Get feature embedding model in addition to the classifier
- Neural networks are black box
 - Training requires significant amount of data

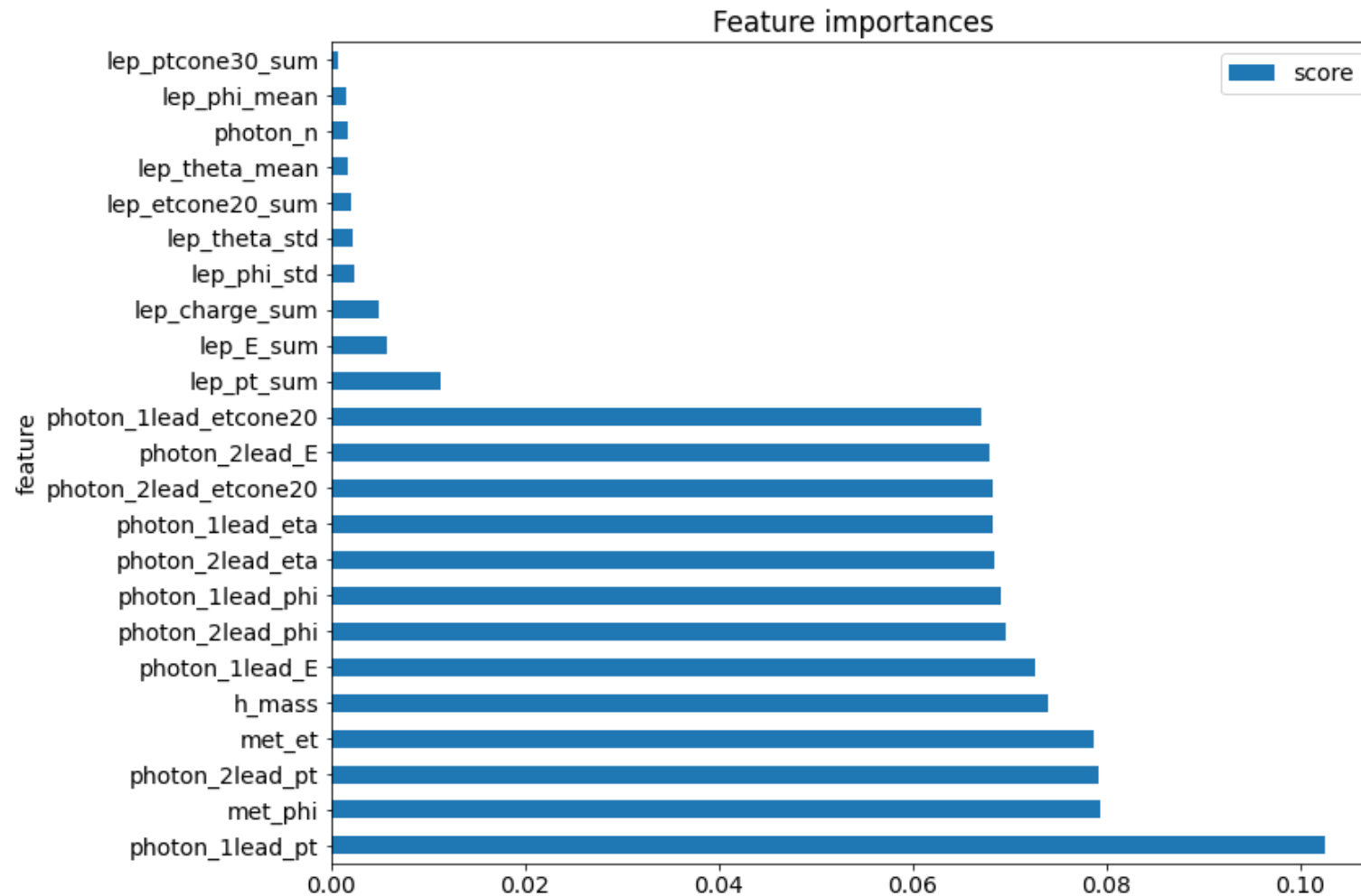
Baseline classifier: Random forest



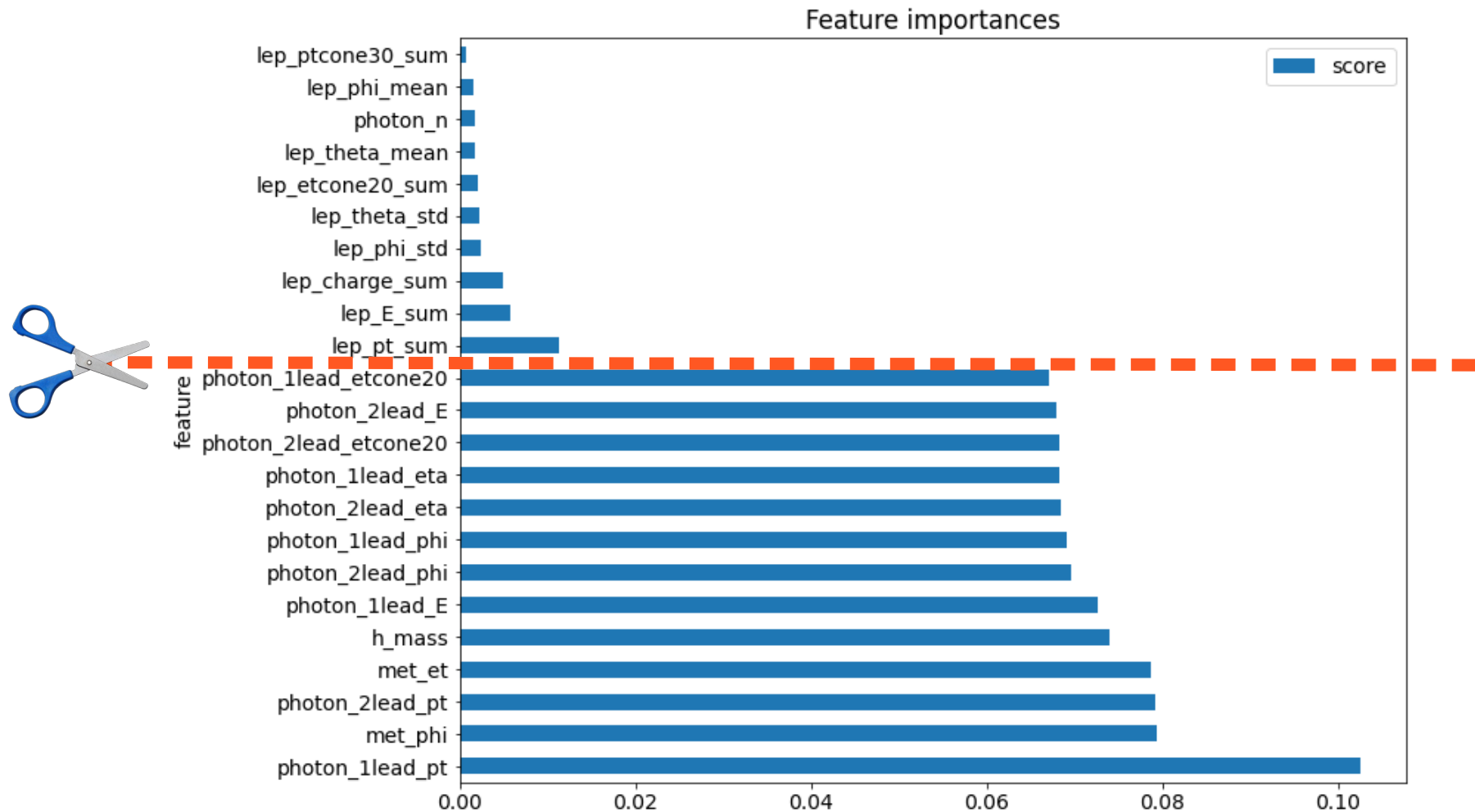
Baseline classifier: Random forest



Baseline classifier: Random forest



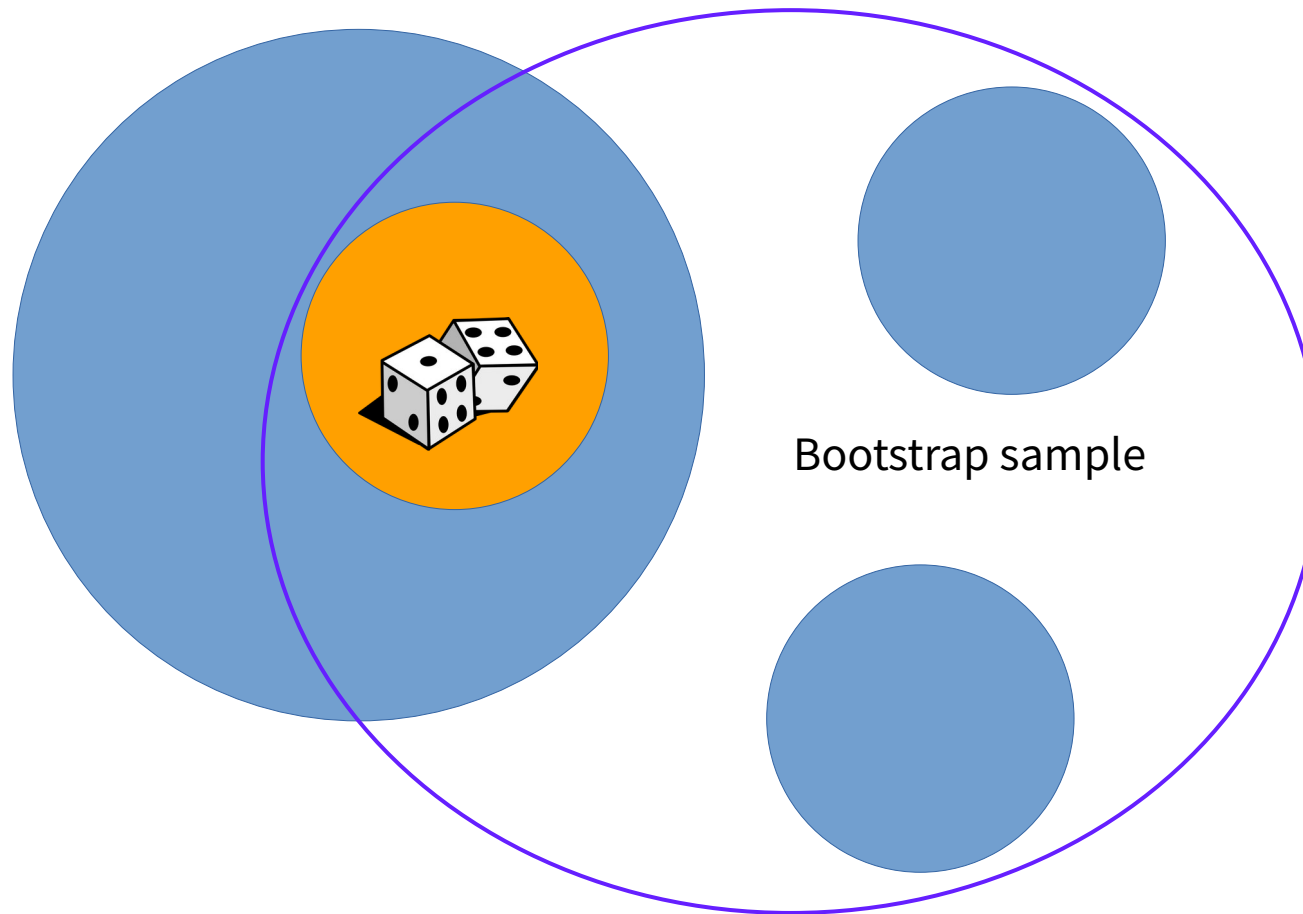
Baseline classifier: Random forest



Imbalanced data

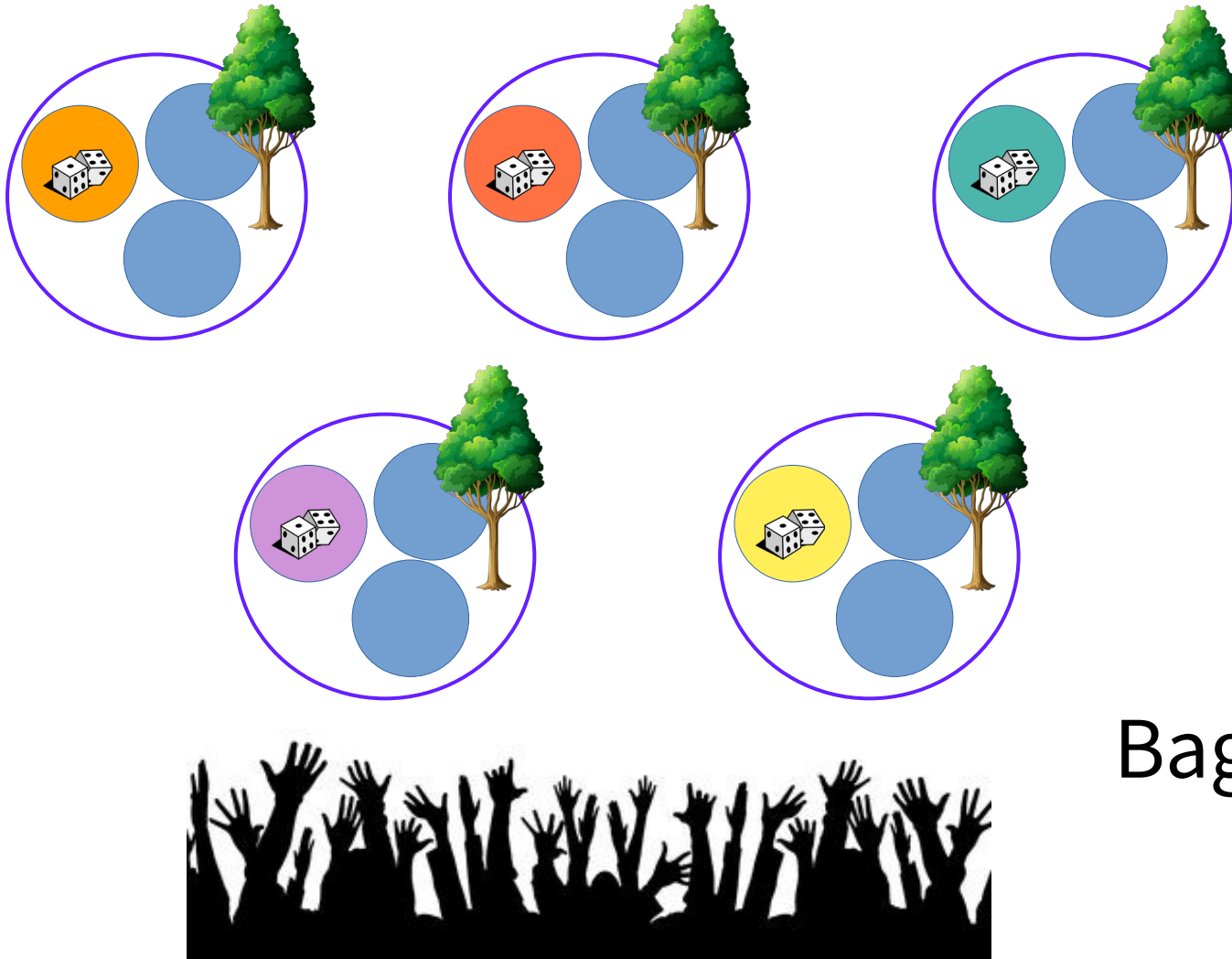
	event_num	weight_sums	weight_frac	weight_per_event
VBF	61021	5.649031e-04	0.080207	9.257520e-09
Wp	12905	1.033396e-04	0.014673	8.007720e-09
Z	26377	9.719725e-05	0.013800	3.684924e-09
gg	121078	6.277415e-03	0.891295	5.184604e-08
tt	59236	1.730808e-07	0.000025	2.921885e-12

Fighting imbalanced data with bagging



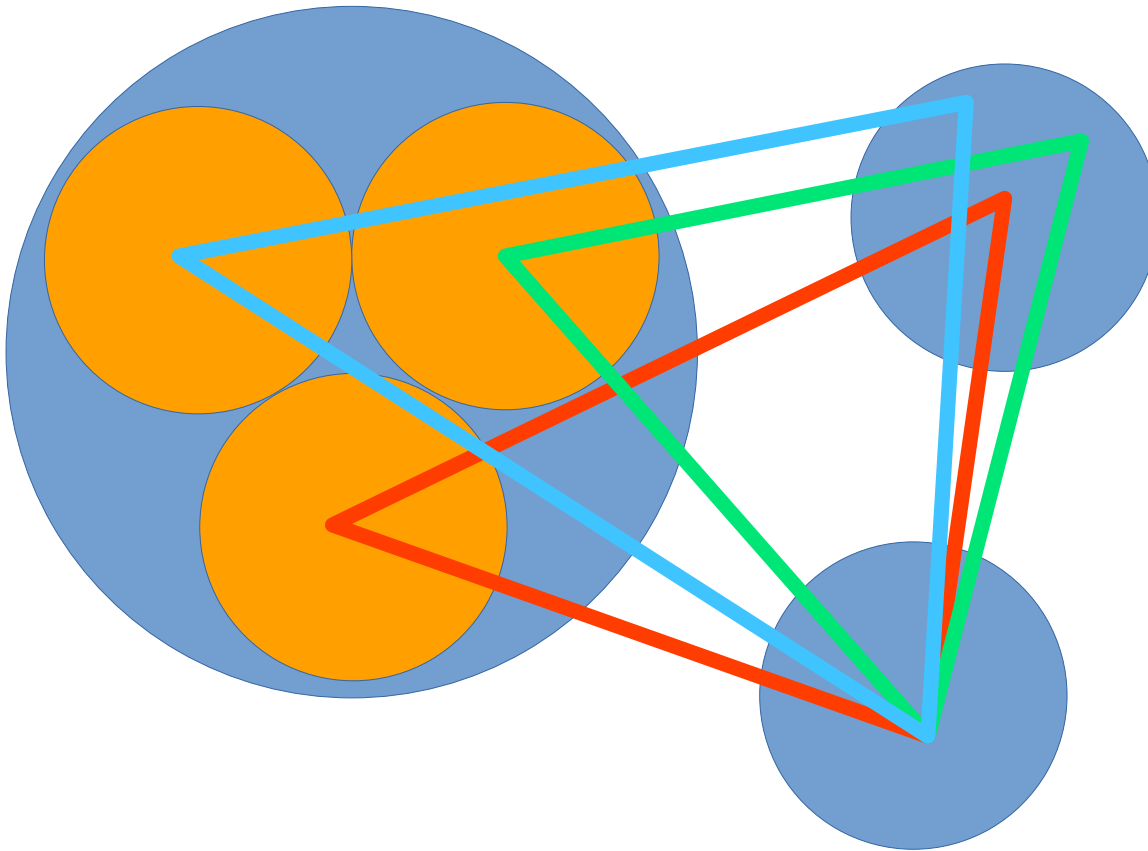
Undersampling

Fighting imbalanced data with bagging



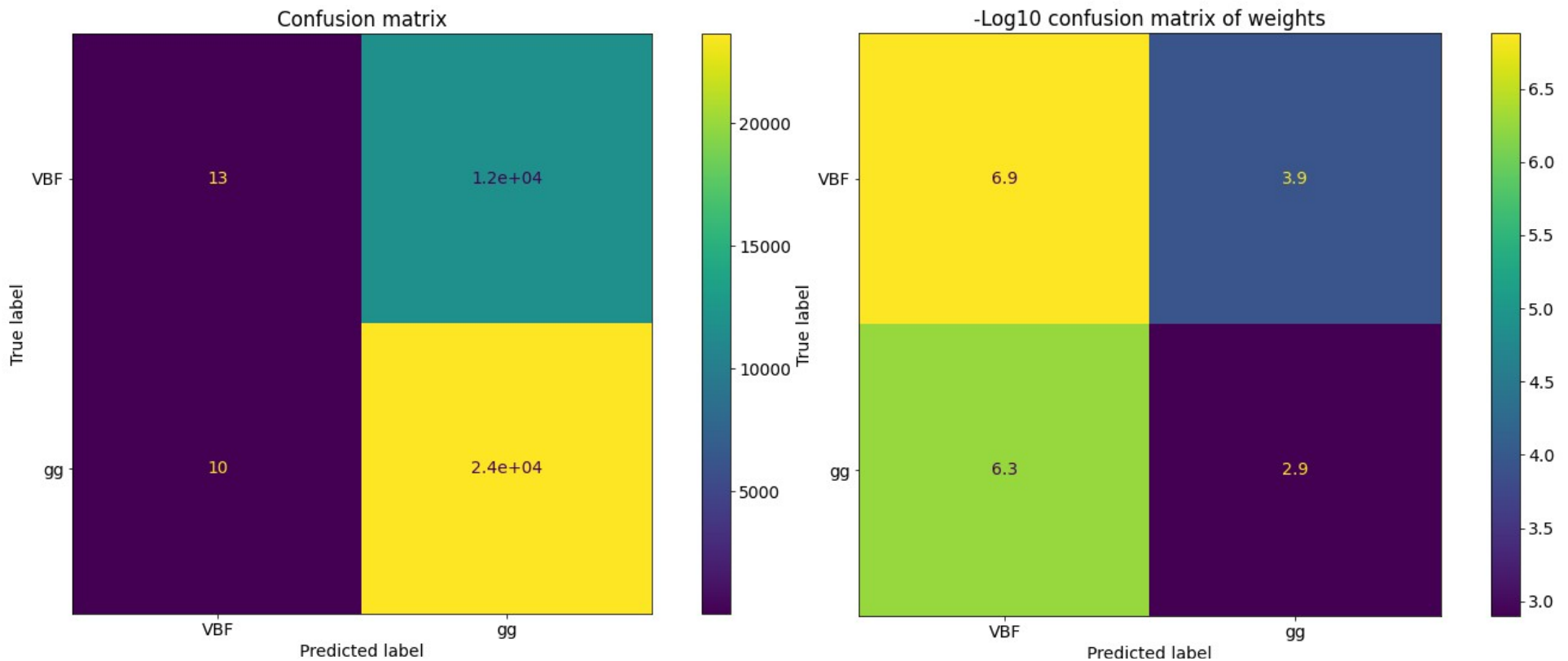
Bagging

Fighting imbalanced data with bagging

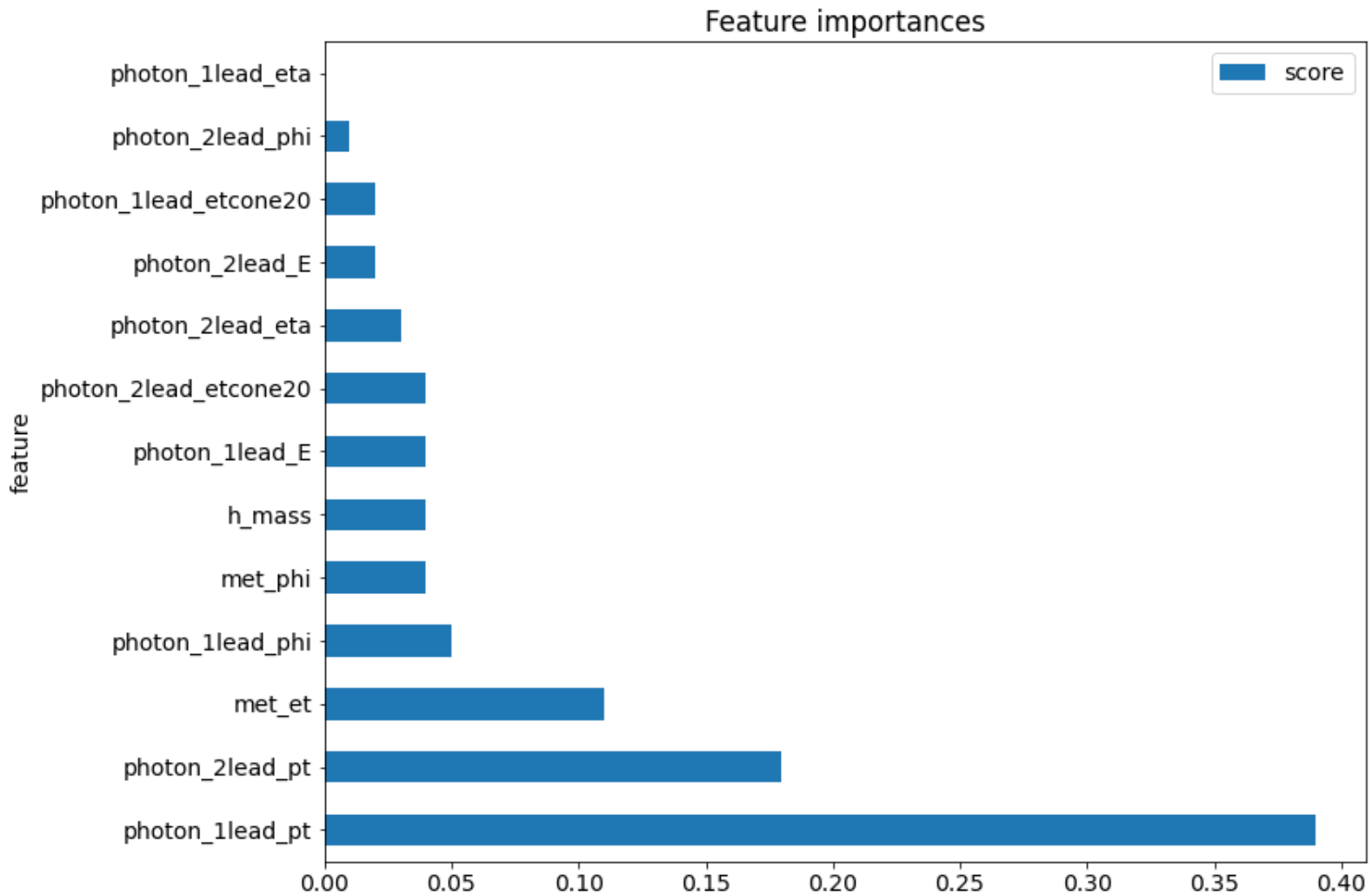


EasyEnsemble

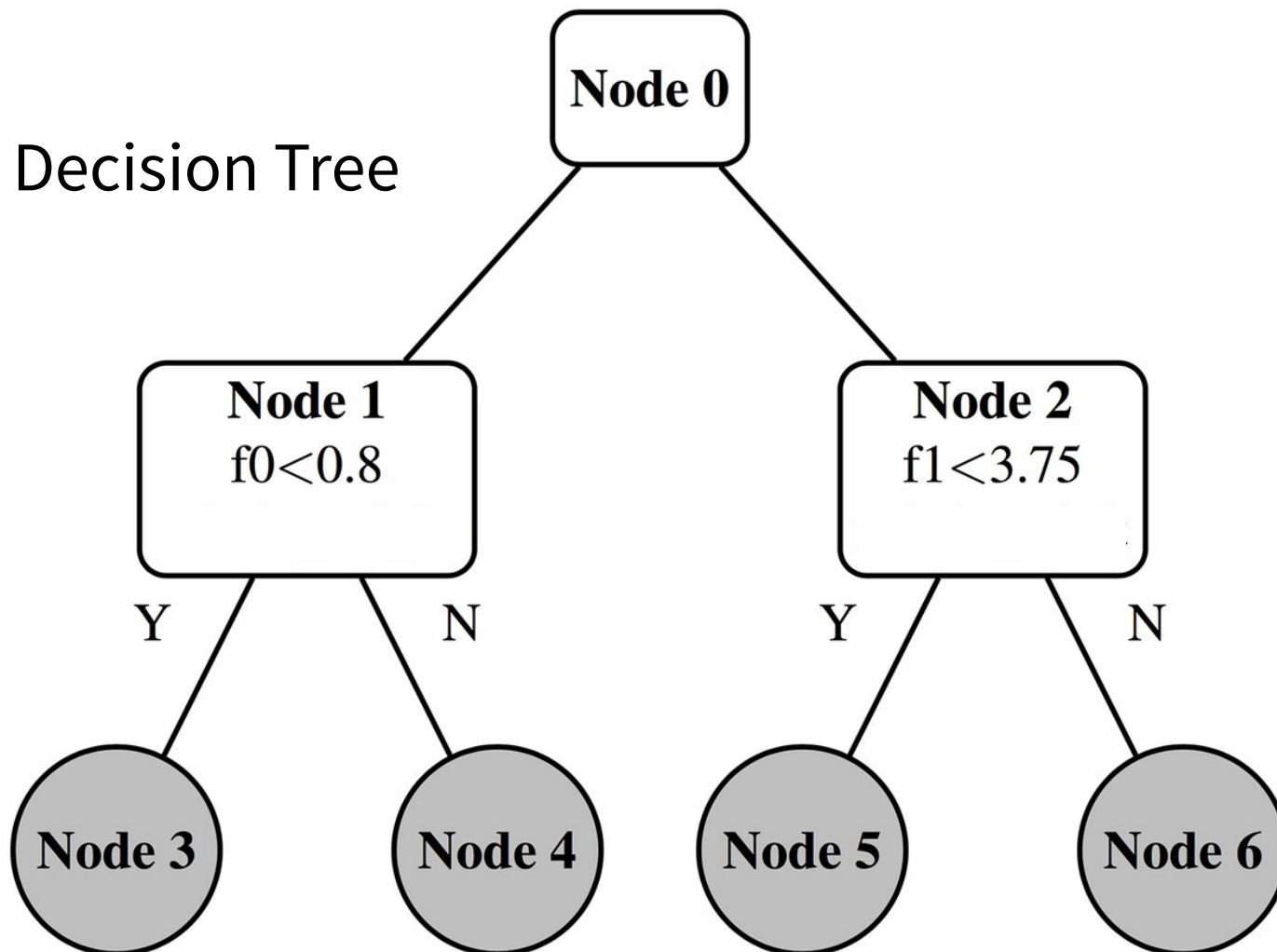
EasyEnsemble model



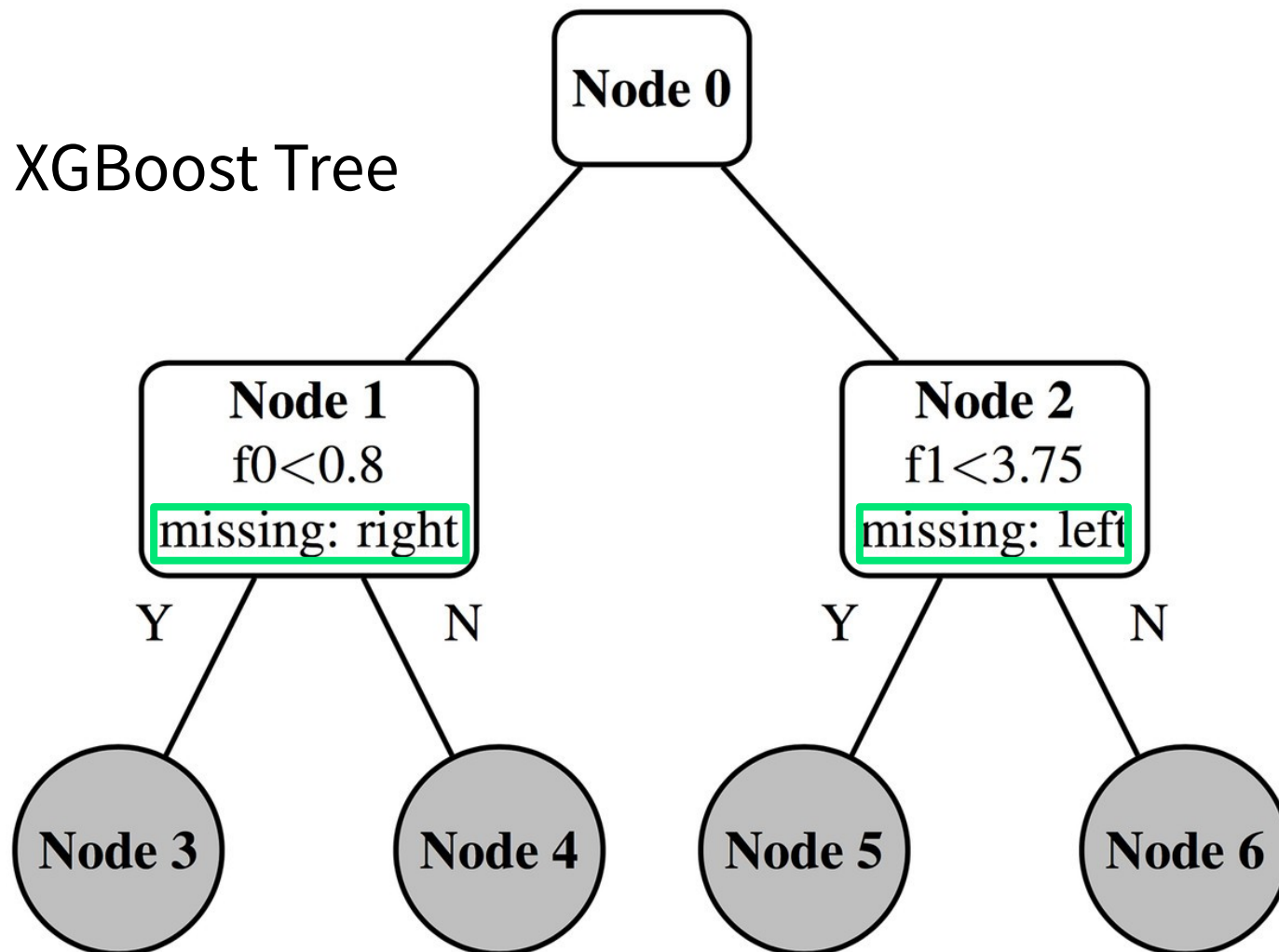
EasyEnsemble model



Alternative to downsampling. XGBoost



Alternative to downsampling. XGBoost



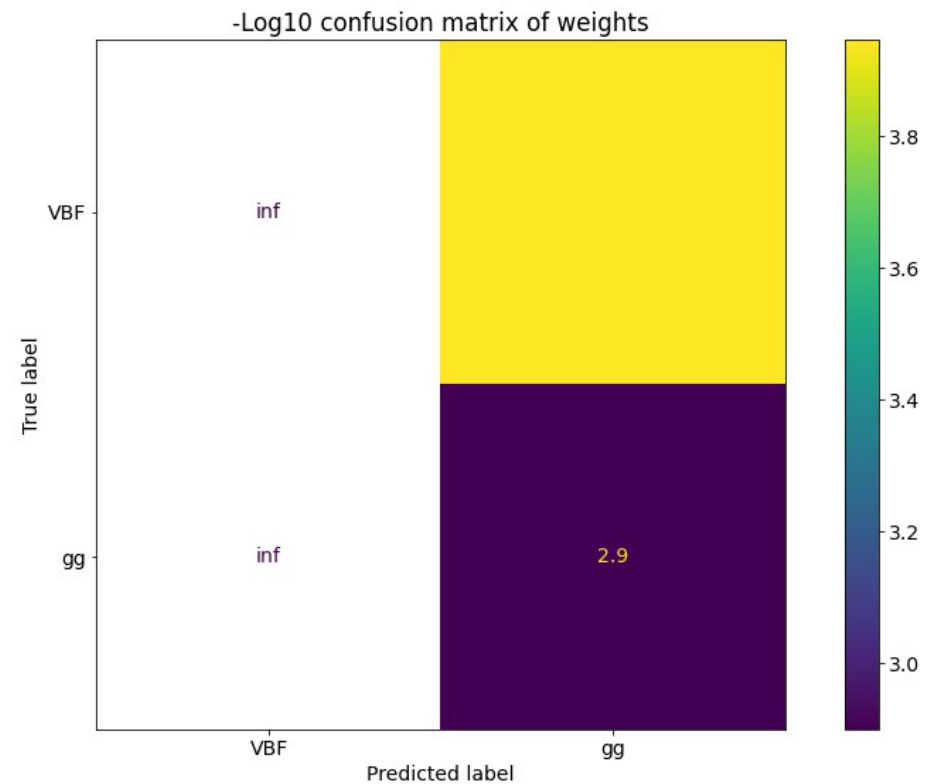
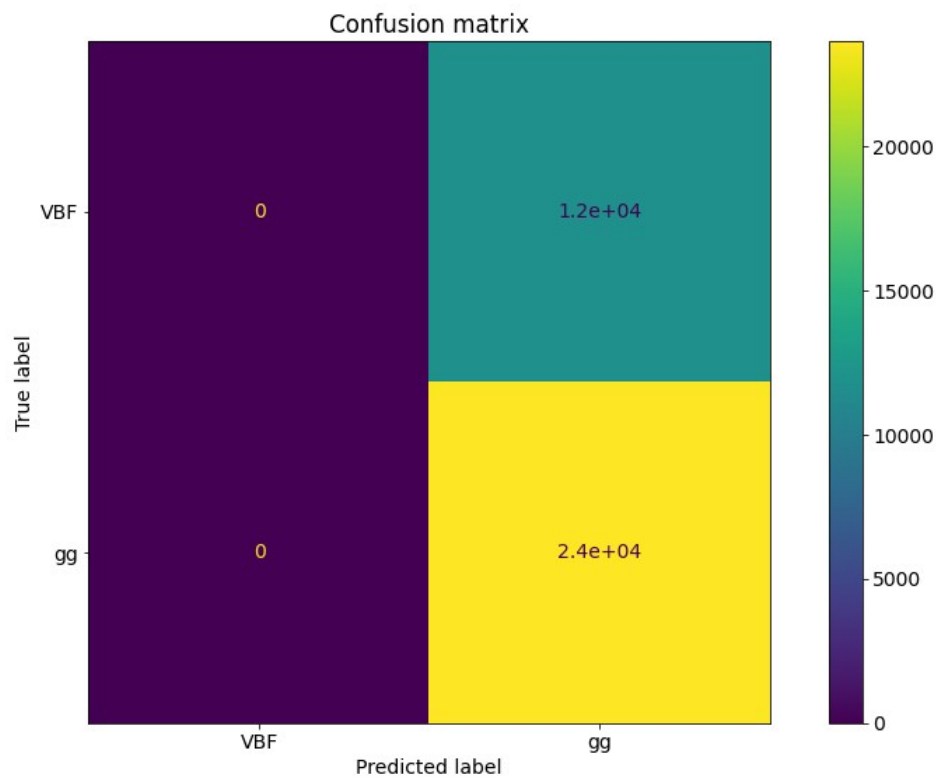
Conclusion

- It is possible to analyze HEP data with Python without loss in efficiency (in comparison to C++)
- HEP datasets are jagged. Possible solutions include:
 - feature aggregation
 - set kernels (Pyramid Match)
 - feature embedding (DeepSets or alternatives)
 - missing data distribution fit (xgboost)
- HEP datasets are imbalanced. Possible solutions include:
 - downsampling
 - oversampling



Backup

XGBoost



XGBoost

