FYS9555 Final project

«Higgs boson production mechanism classification from ATLAS Open Data»

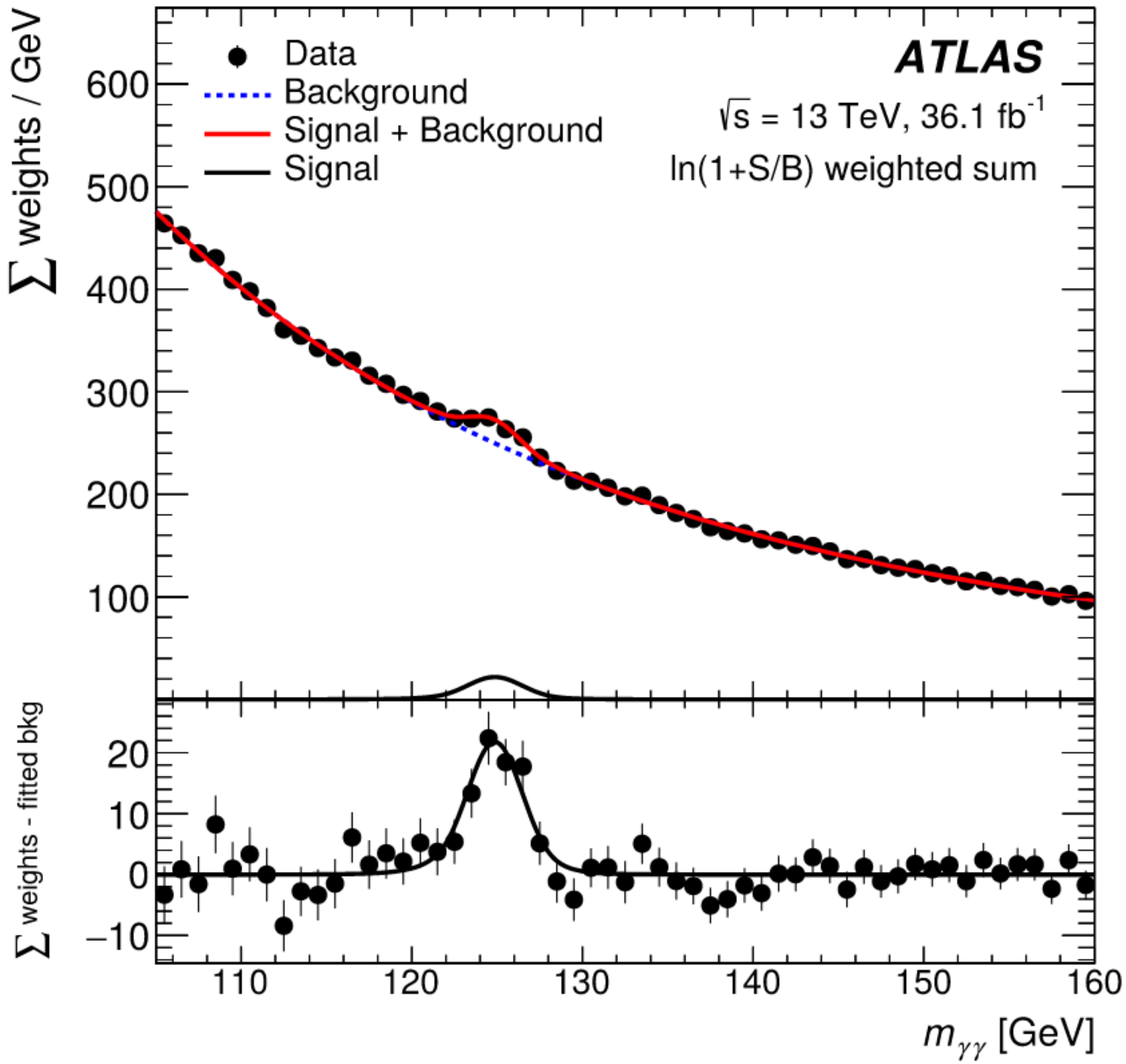
10 June 2020

Victor Ananyev

Outline

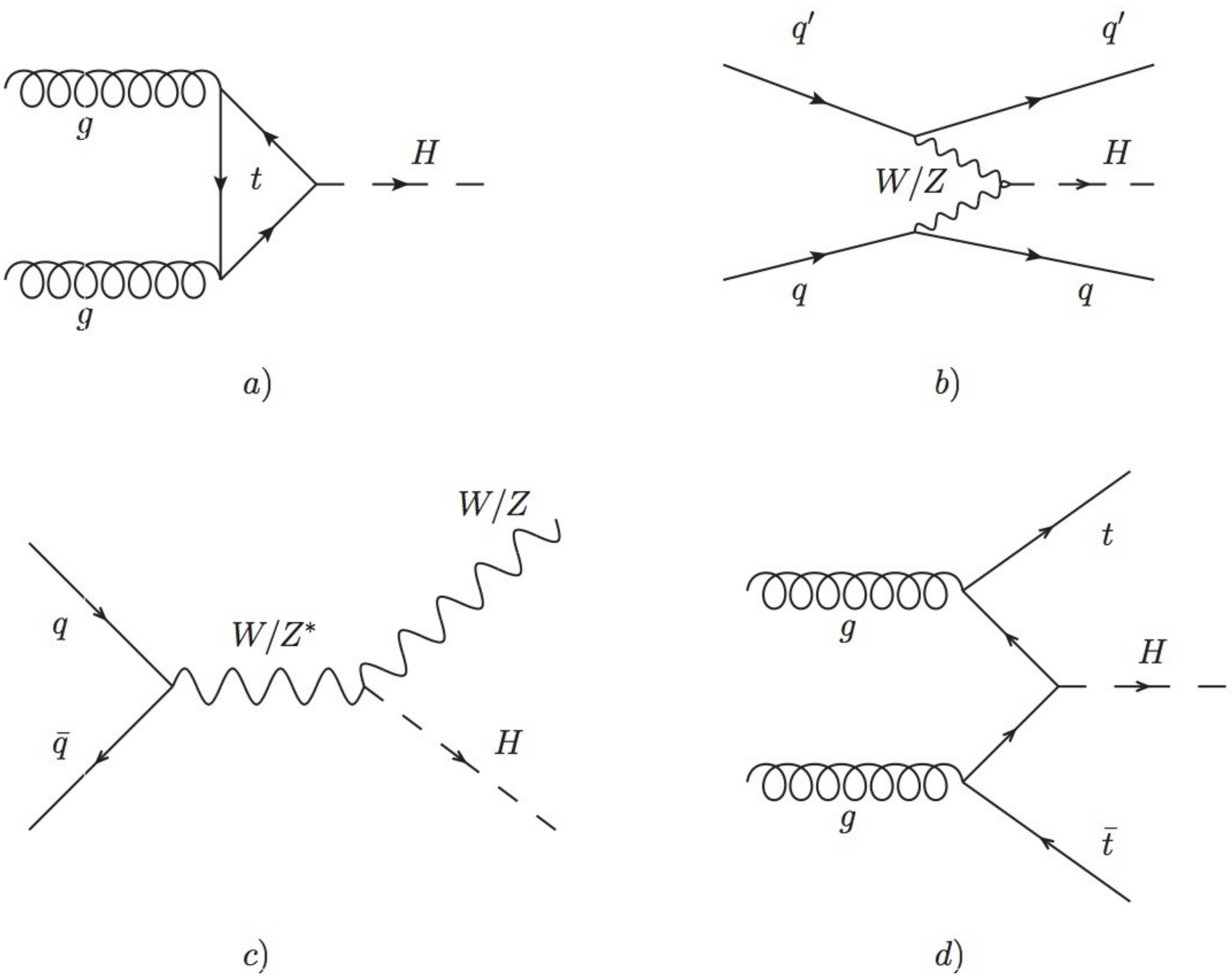
- Di-photon process
- Data acquisition
- What's inside?
- Dealing with jagged data in ML
- Fighting imbalanced data sets
- Fitting baseline ML models

Di-photon process



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

Major production mechanisms

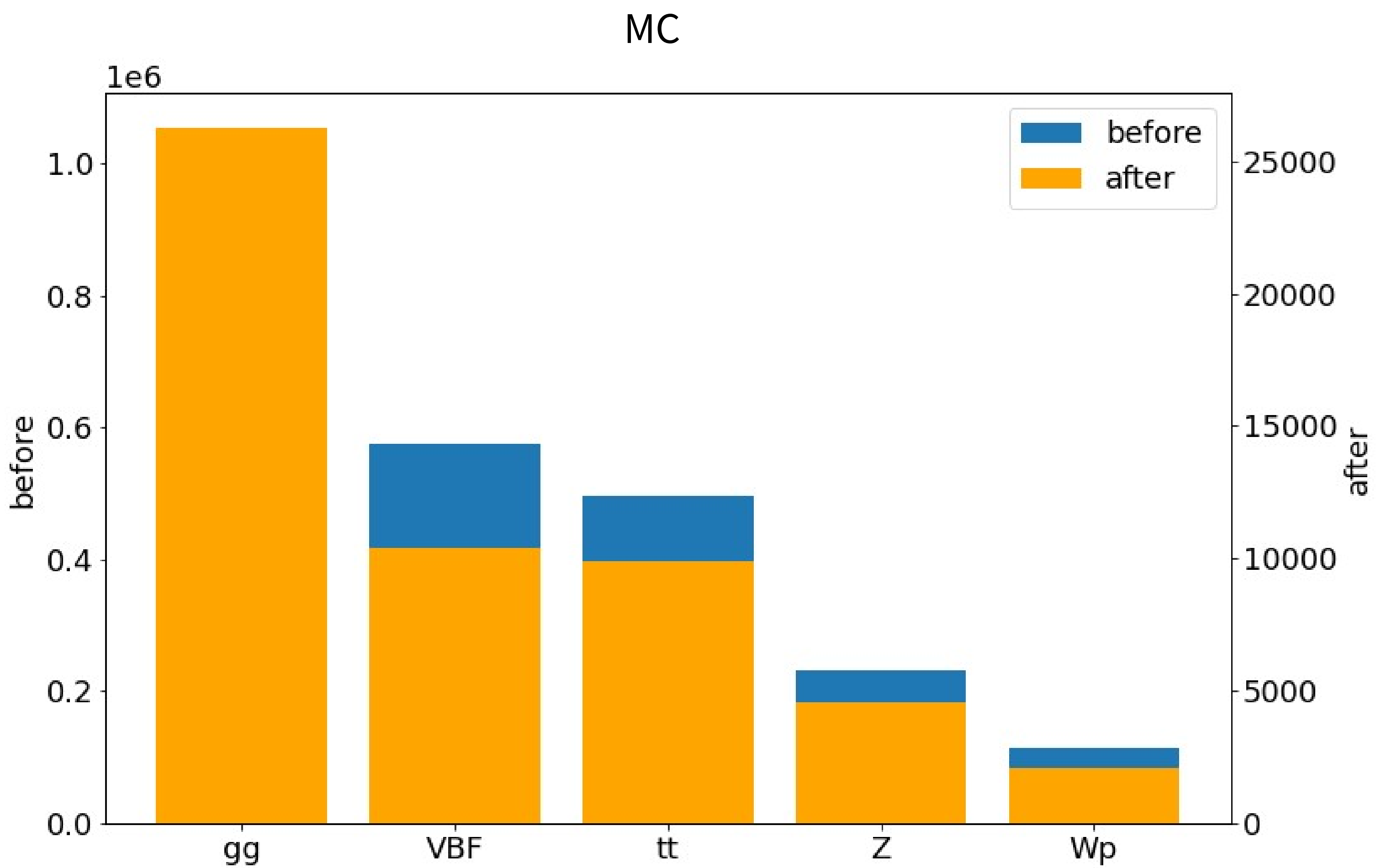


(borrowed from https://grab-group.ethz.ch/research/higgs-boson.html)

Event selection for Hγγ peak

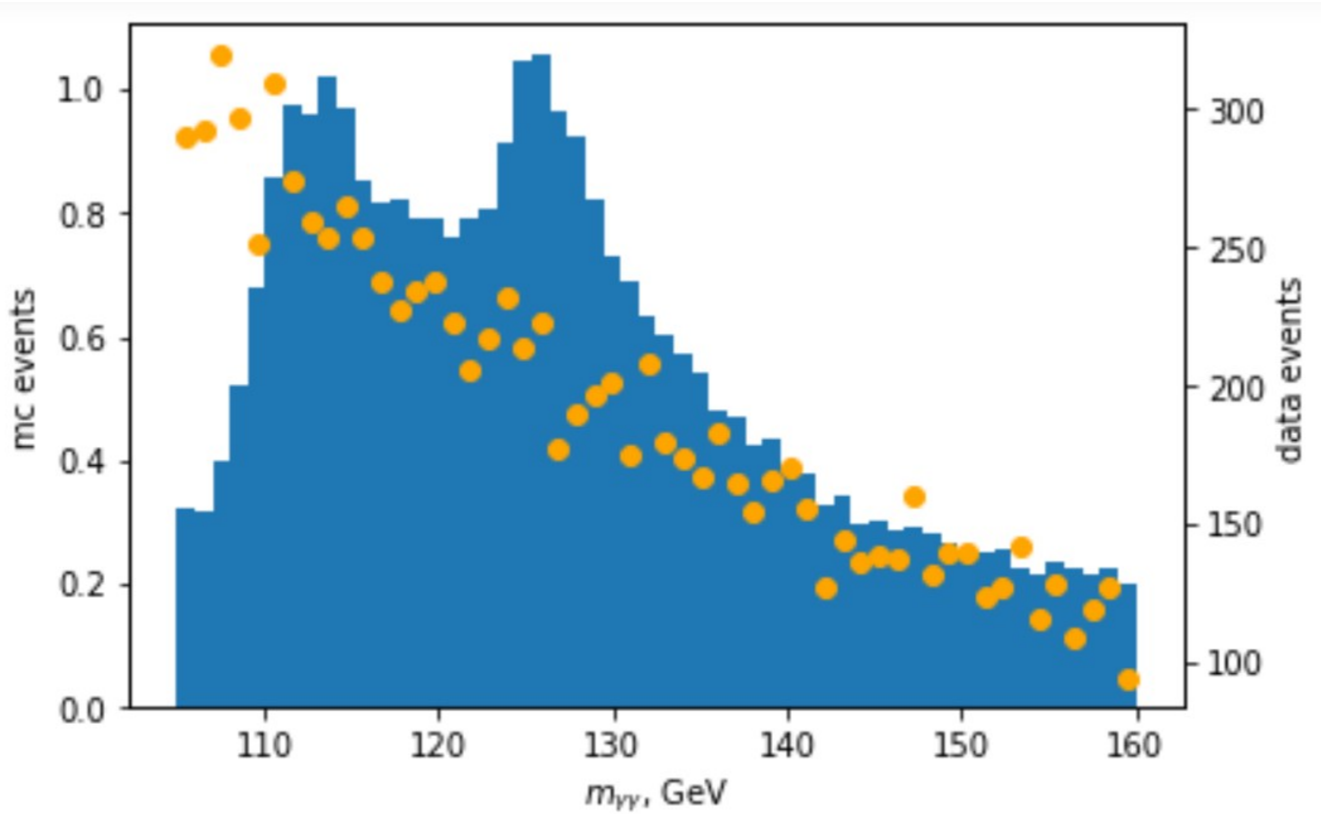
- Diphoton trigger is satisfied;
- Exactly two photons with $E_{\mathrm{T}} > 35$ and 25 GeV, respectively;
- Leading and subleading photon candidates are respectively required to have $E_{\mathrm{T}}/m_{\gamma\gamma} > 0.35$ and 0.25;
- Diphoton invariant mass $m_{\gamma\gamma}$ between 105 GeV and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

Selection efficiency ~ 1%



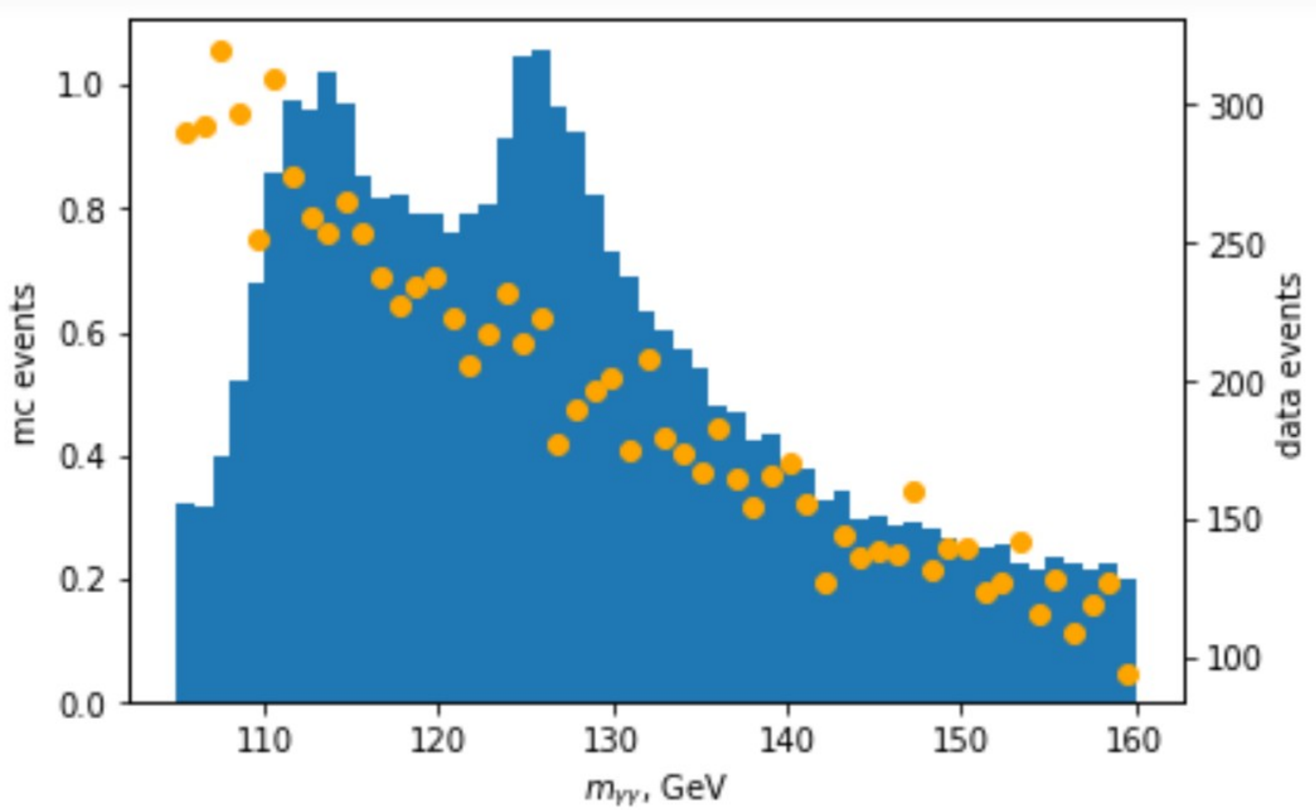
Number of events per process

Higgs mass from ATLAS Open Data



8M events in 7 minutes!

Acquire data efficiently



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

Data pipeline

CERNBox

1TB Root files storage

Data pipeline

CERNBox

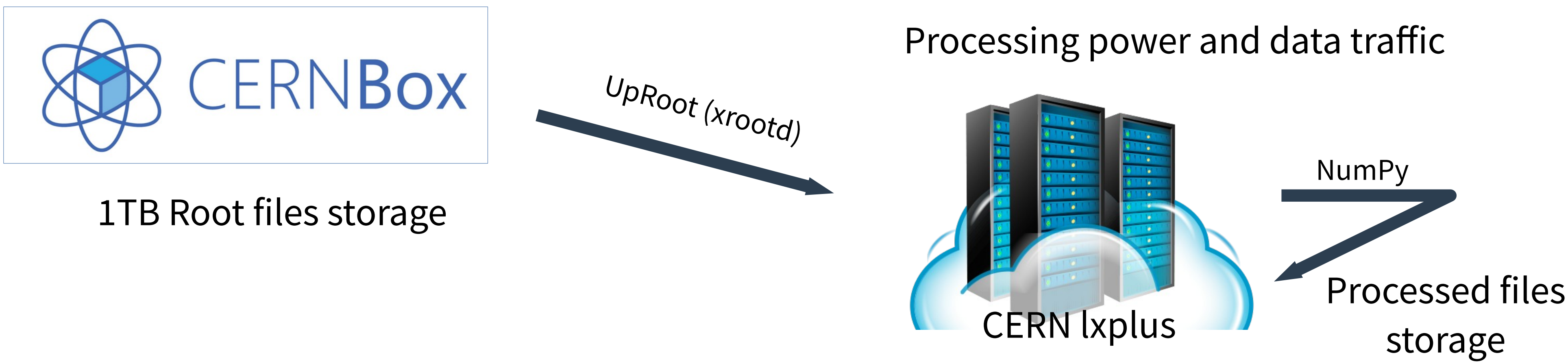
1TB Root files storage

UpRoot (xrootd)

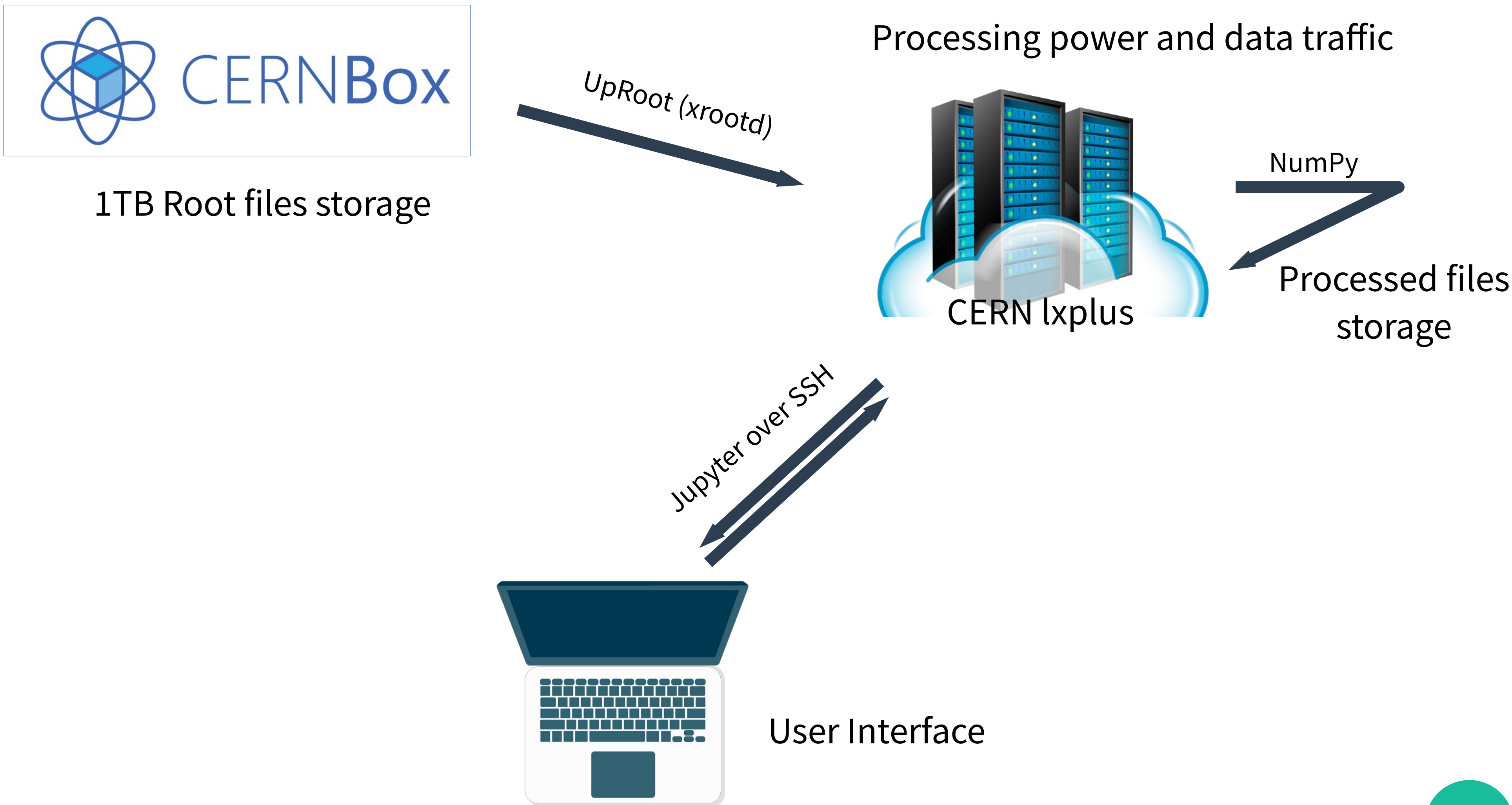
Processing power and data traffic

CERN lxplus

Data pipeline



Data pipeline



Data pipeline

Processing power and data traffic

1TB Root files storage

CERNBox

UpRoot (xrootd)

CERN lxplus

NumPy

2GB – 100 GB

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
- photon_truthMatched
- photon_trigMatched
- photon_pt
- photon_eta
- photon_phi
- photon_E
- photon_isTightID
- photon_ptcone30
- photon_etcone20
- photon_convType
- photon_pt_syst

- lep_n
- lep_truthMatched
- lep_trigMatched
- lep_pt
- lep_eta
- lep_phi
- lep_E
- lep_z0
- lep_charge
- lep_type
- lep_isTightID
- lep_ptcone30
- lep_etcone20
- lep_trackd0pvunbiased
- lep_tracksigd0pvunbiased
- lep_pt_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

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

- photon_n
- photon_truthM
- photon_trig
- photon_p
- photon_
- photon_phi
- photon_E
- photon_isTightID
- photo ne30
- photo ne20
- photo Type
- photo yst

800

Photon Features

- lep_n
- lep_truthMatched
- lep_trigMatched
- p_pt
- _eta
- _phi
- lep_E
- lep_z0
- lep_charge
- lep_type
- lep_isTightID
- lep_ptcone30
- lep_etcone20
- lep_trackd0pvunbiased
- lep_tracksigd0pvunbiased
- lep_pt_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

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

- photon_n
- photon_truthM
- photon_trig
- photon_p
- photon_eta
- photon_phi
- photon_E
- photon_isTightID
- photo ne30
- photo ne20
- photo Type
- photo yst

800

Photon Features

- lep_n
- lep_truthMatch
- lep_trigMa
- p_pt
- eta
- _phi
- lep_E
- lep_z0
- lep_c
- lep_t
- lep_i D
- lep_p 30
- le
- le vunbiased
- pvunbiased
- lep_pt_syst

600

Lepton Features

- tau_n
- tau_pt
- au_eta
- phi
- E
- 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

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

MASK MASK MASK

1000 events

- photon_n
- photon_truthM
- photon_trig
- photon_p
- photon_eta
- photon_phi
- photon_E
- photon_isTightID
- photo ne30
- photo ne20
- photo Type
- photo yst

800

Photon Features

- lep_n
- lep_truthMatch
- lep_trigMa
- p_pt
- eta
- _phi
- lep_E
- lep_z0
- lep_c
- lep_t
- lep_i D
- lep_p 30
- le
- le vunbiased
- pvunbiased
- lep_pt_syst

600

Lepton Features

- tau_n
- tau_pt
- au_eta
- _phi
- E
- _isTightID
- tau_truthMatched
- tau_trigMatched
- tau_nT
- tau_BD
- tau_pt_
- tau_cha

300

Tau Features

- jet_n
- jet_pt
- t_eta
- hi
- 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

250 events

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

1000 events

- photon_n
- photon_truthM
- photon_trig
- photon_p
- photon_eta
- photon_phi
- photon_E
- photon_isTightID
- photo ne30
- photo ne20
- photo Type
- photo yst

- lep_n
- lep_truthMatch
- lep_trigMa
- p_pt
- eta
- _phi
- lep_E
- lep_z0
- lep_c
- lep_t
- lep_i D
- lep_p 30
- le
- le vunbiased
- pvunbiased
- lep_pt_syst

- tau_n
- tau_pt
- au_eta
- _phi
- E
- _isTightID
- tau_truthMatched
- tau_trigMatched
- tau_nT
- tau_BD
- tau_pt_
- tau_cha

- jet_n
- jet_pt
- t_eta
- hi
- 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

250 events

MASK

MASK

MASK

800

600

250

Photon Features

Lepton Features

Tau Featur

MASK

...

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 …] > [0, 0, 0, 1, 0, 0 …]

MASK MASK MASK

1000 events

- photon_n
- photon_truthM
- photon_trig
- photon_p
- photon_eta
- photon_phi
- photon_E
- photon_isTightID
- photo ne30
- photo ne20
- photo Type
- photo yst

800

Photon Features

- lep_n
- lep_truthMatch
- lep_trigMa
- p_pt
- eta
- _phi
- lep_E
- lep_z0
- lep_c
- lep_t
- lep_i D
- lep_p 30
- le
- le ased
- biased
- lep_pt_syst

250

Lepton Feature

- tau_n
- tau_pt
- au_eta
- _phi
- E
- isTightID
- tau_truthMatched
- tau_trigMatched
- tau_nT
- tau_BD
- tau_pt_
- tau_ch

250

Tau Featur

- jet_n
- jet_pt
- eta
- hi
- 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

250 events

Micro features

MASK MASK

Mask bubbling for feature extraction

Simulation wide

- XSection
- SumWeights

Macro features

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

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

Micro features

1000 events

- photon_n
- photon_truthM
- photon_tri
- photon_p
- photon_
- photon_phi
- photon_E
- photon_isTightID
- photo ne30
- photo ne20
- photo Type
- photo yst

- lep_n
- lep_truthMatc
- lep_trigMa
- p_pt
- eta
- _phi
- lep_E
- lep_z0
- lep_c
- lep_t
- lep_i D
- lep_p 30
- le
- le ased
- ibiased
- _syst

- tau_n
- tau_pt
- au_eta
- _phi
- _E
- _isTightID
- tau_truthMatched
- tau_trigMatched
- tau_nT
- tau_BD
- tau_pt_
- tau_cha

- jet_n
- jet_pt
- _eta
- hi
- t
- 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

250 events

MASK MASK MASK

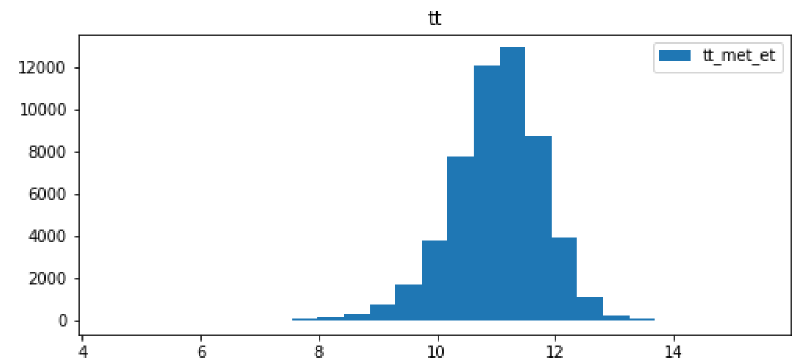
250 250 250

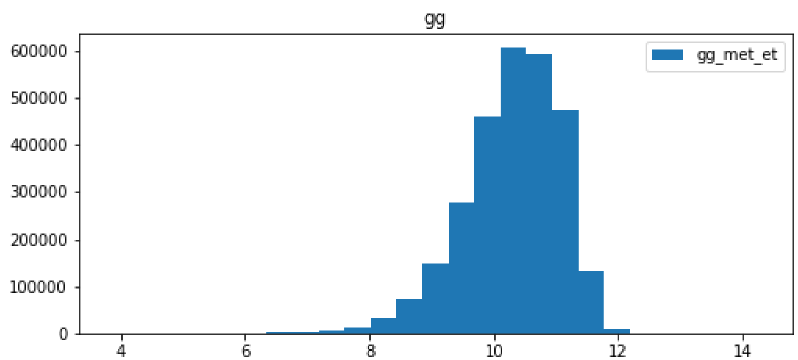
Photon Featur **Lepton ture** **Tau tur** ...

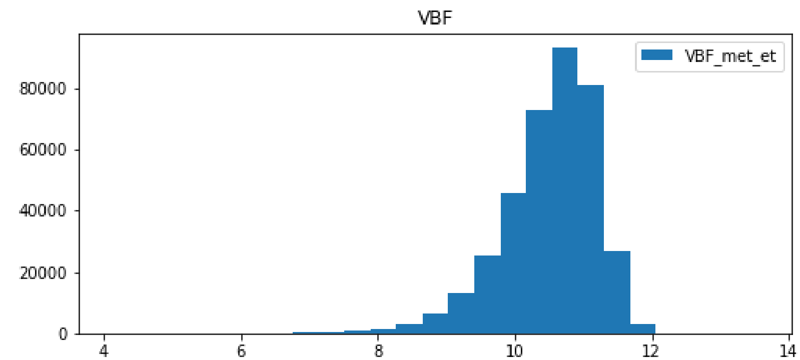
MASK MASK MASK

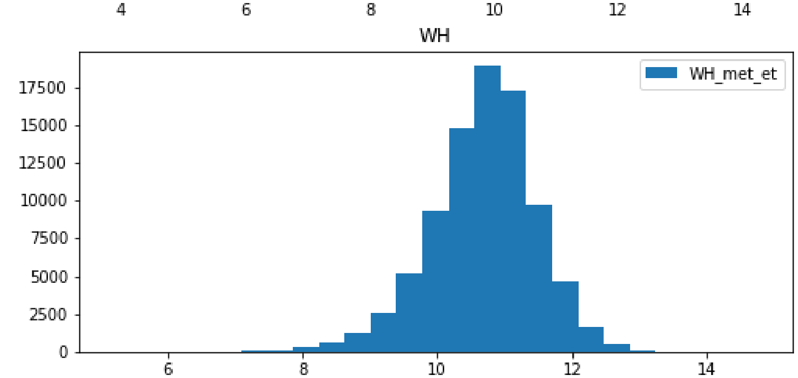
Missing energy Et (log x scale)

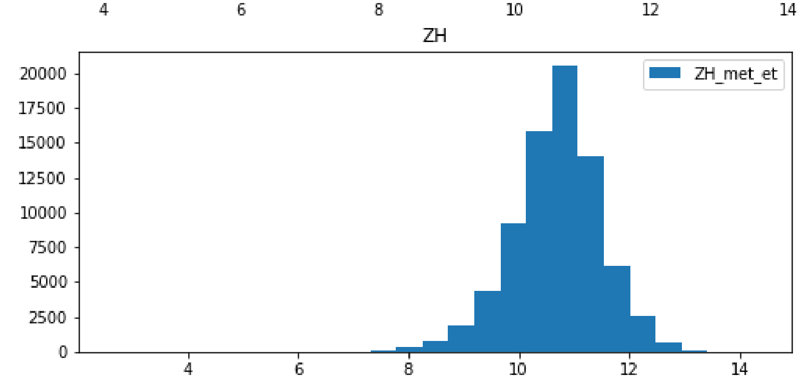
Within 100 000 events batch





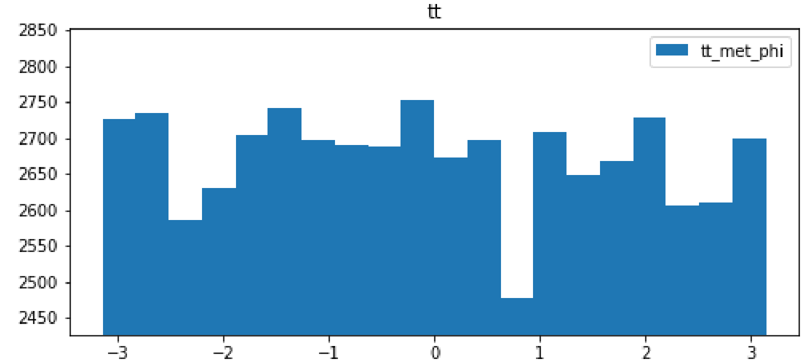


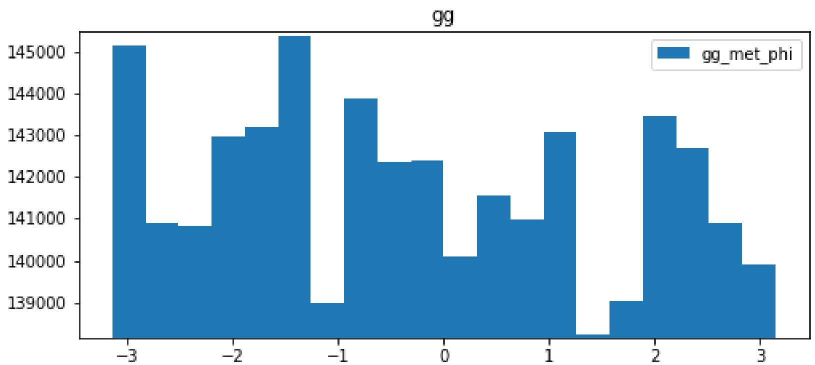


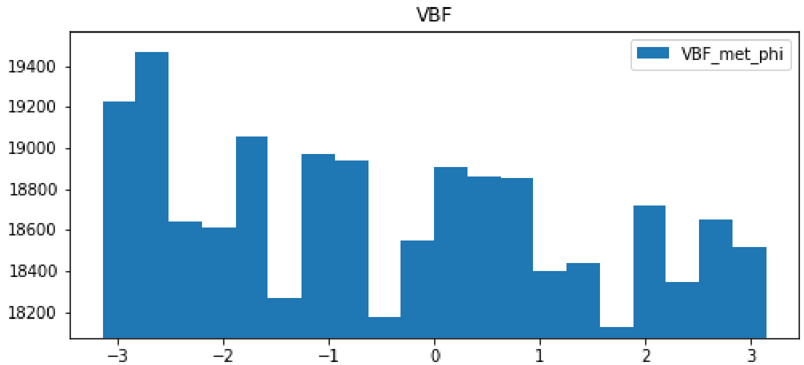


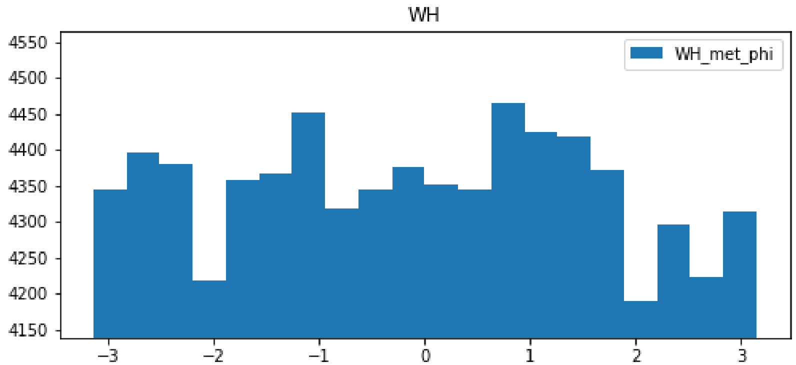
Missing energy Phi

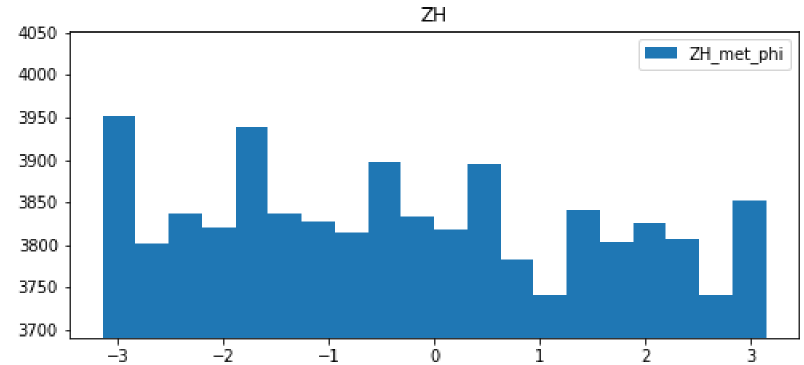
Within 100 000 events batch





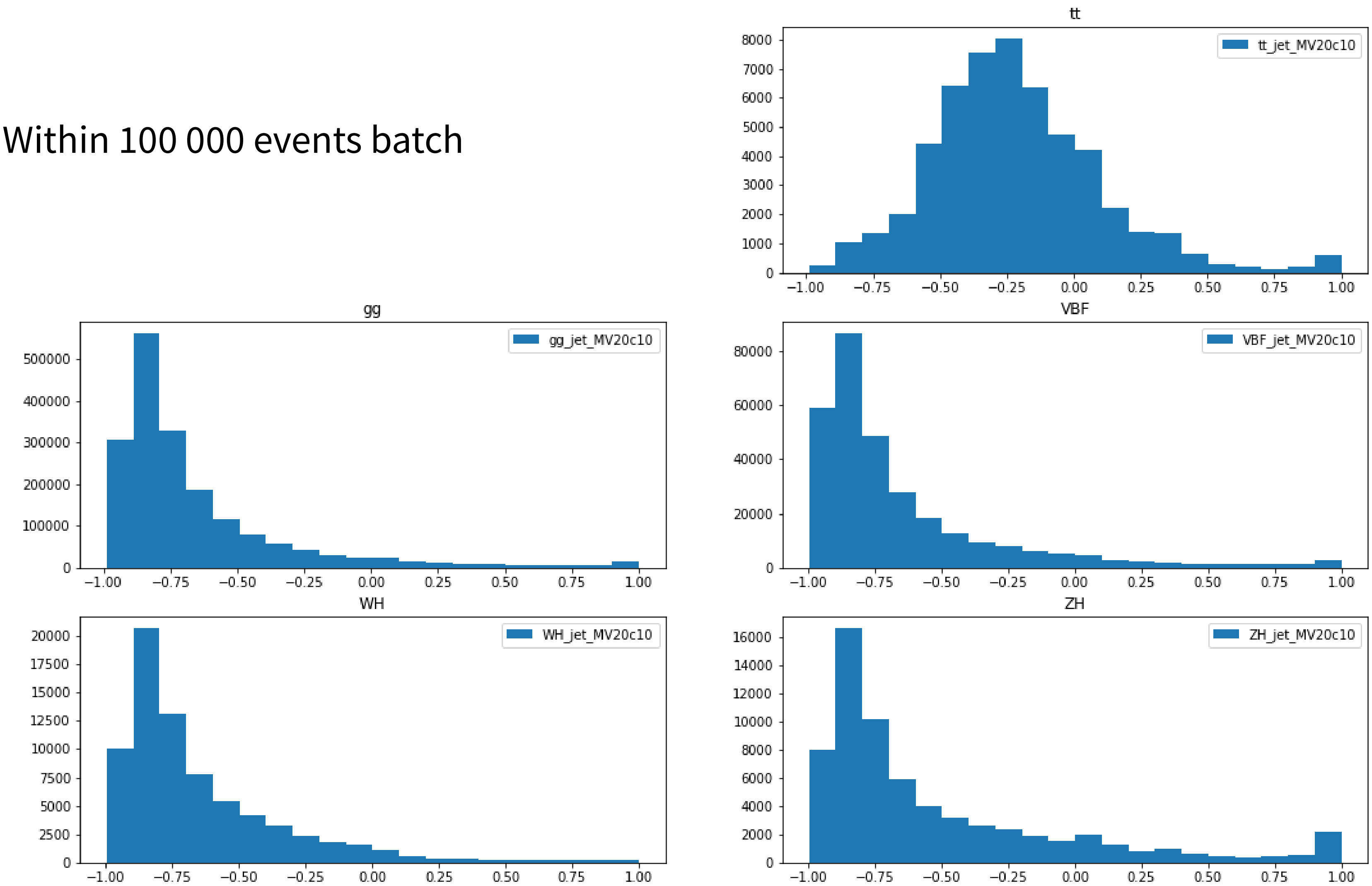






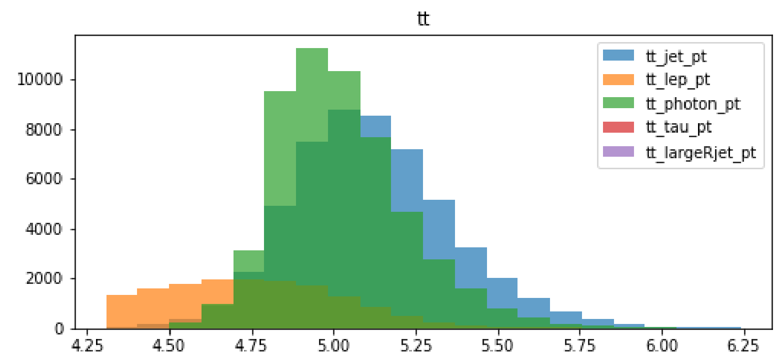
Jet b-tagging score

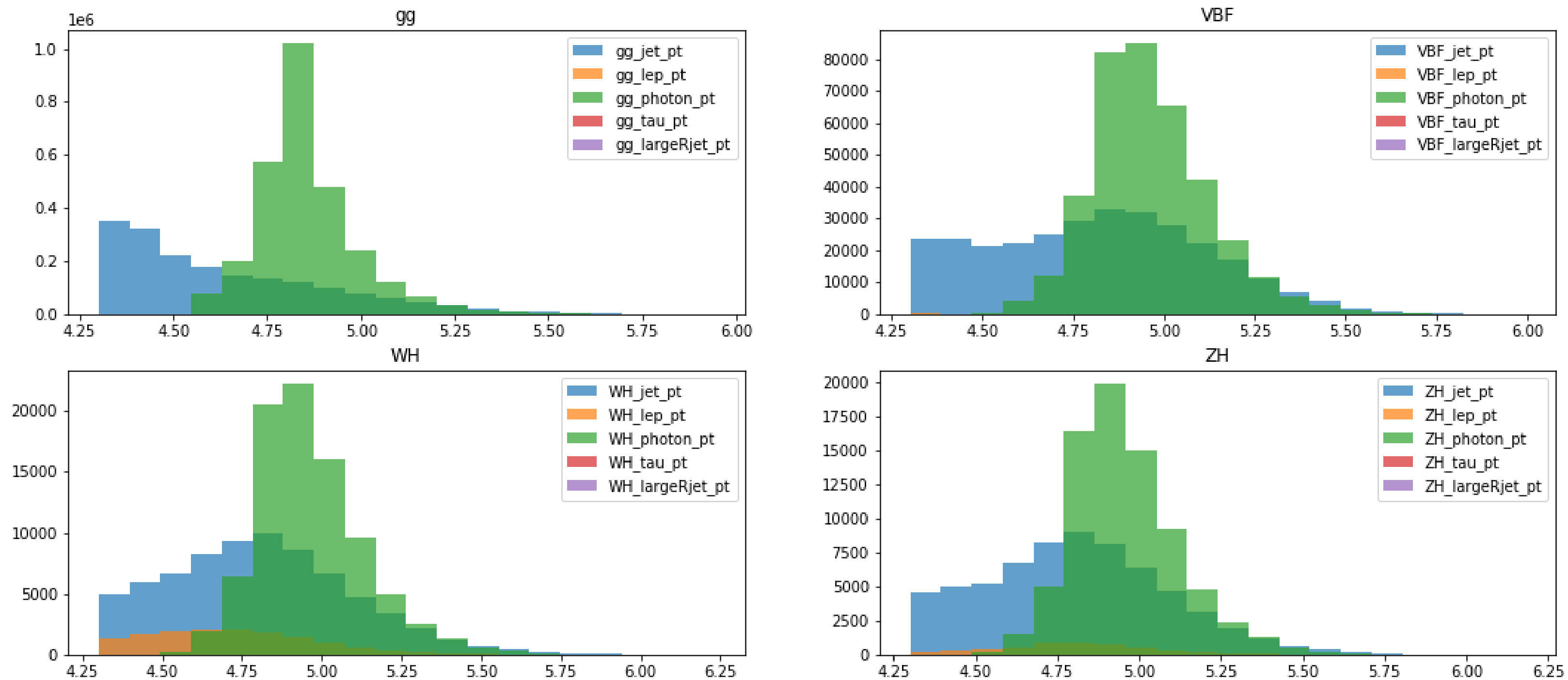
Within 100 000 events batch



Max pT per particle type (log x scale)

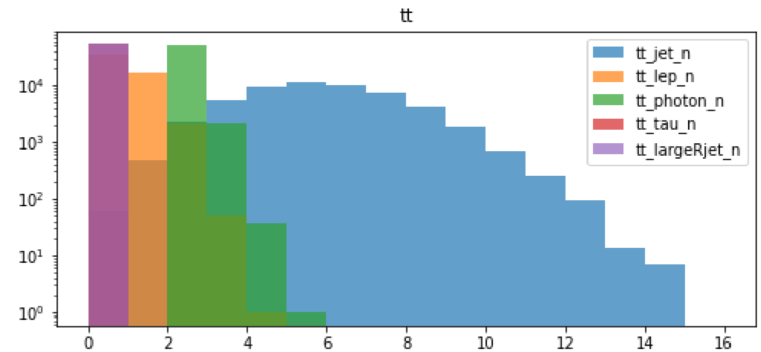
Within 100 000 events batch

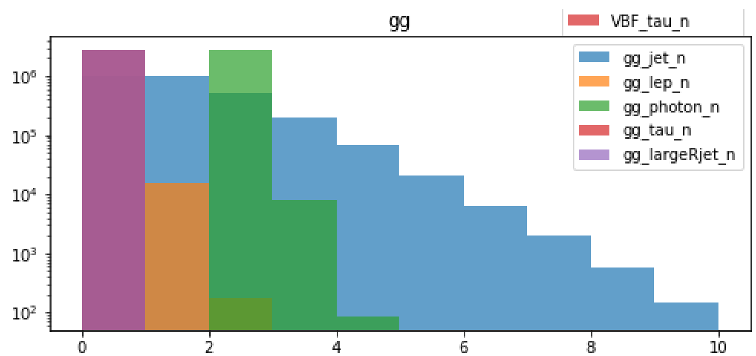


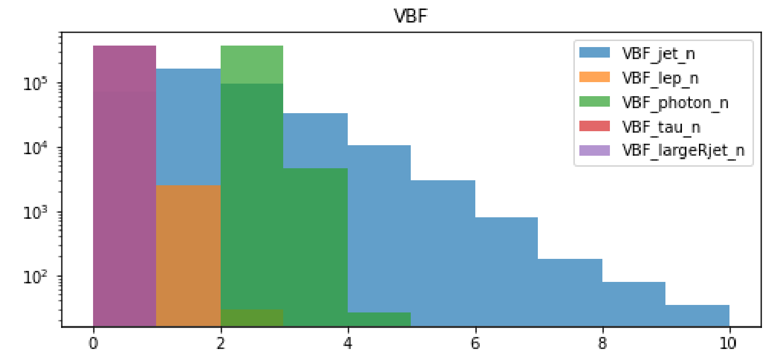


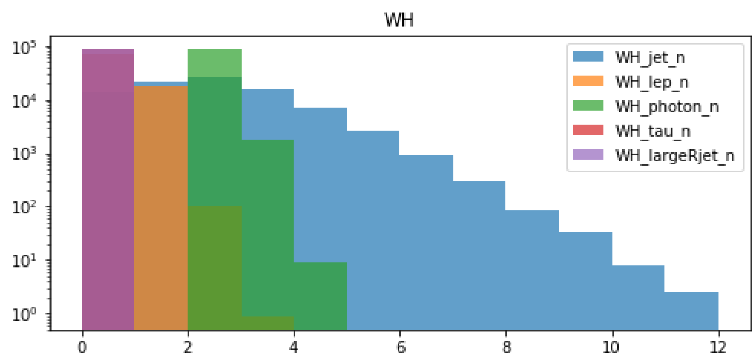
Number of particles (log y scale)

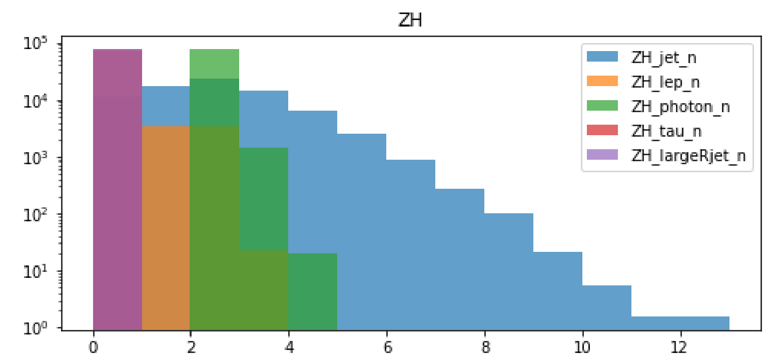
Within 100 000 events batch



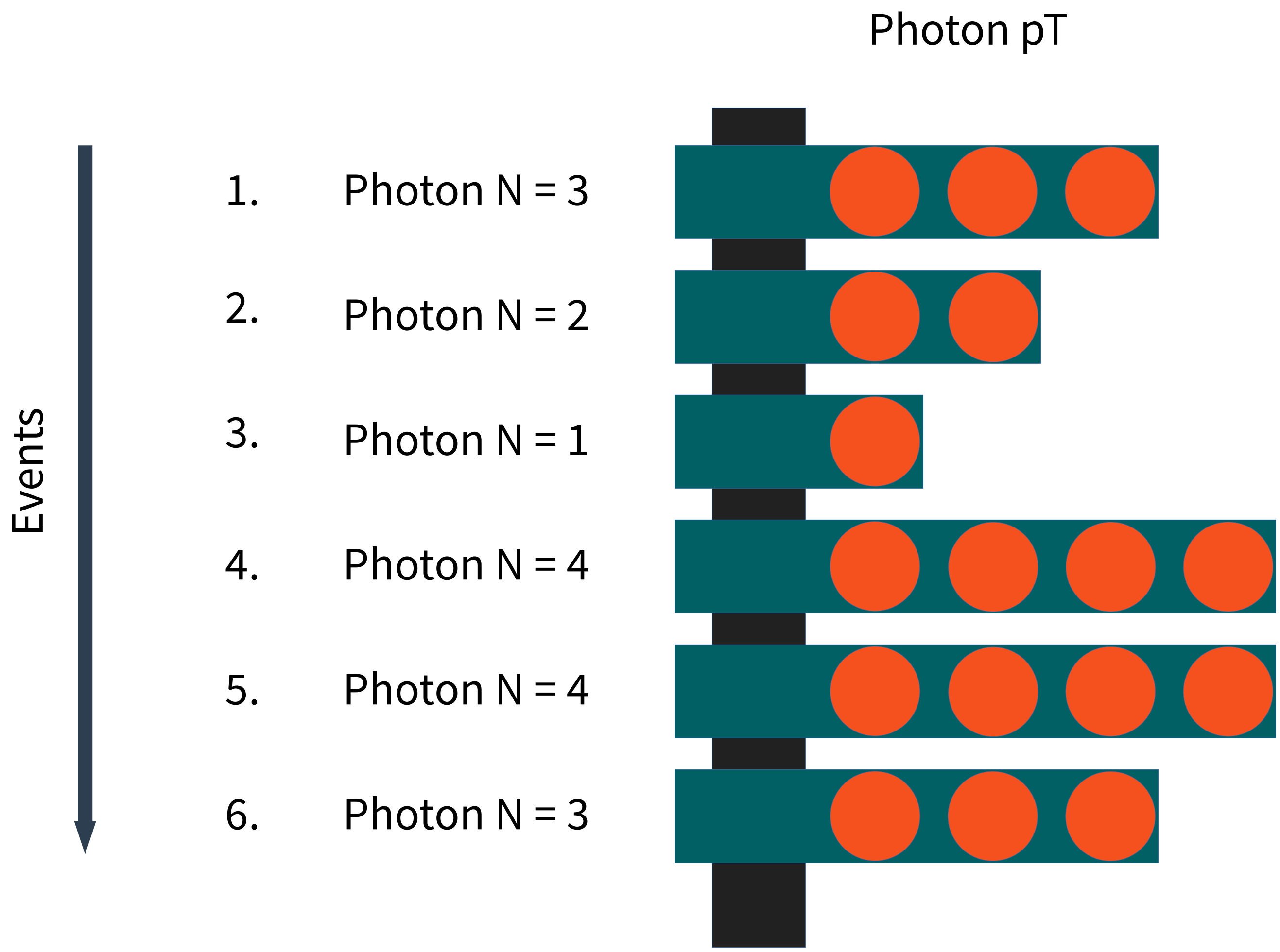




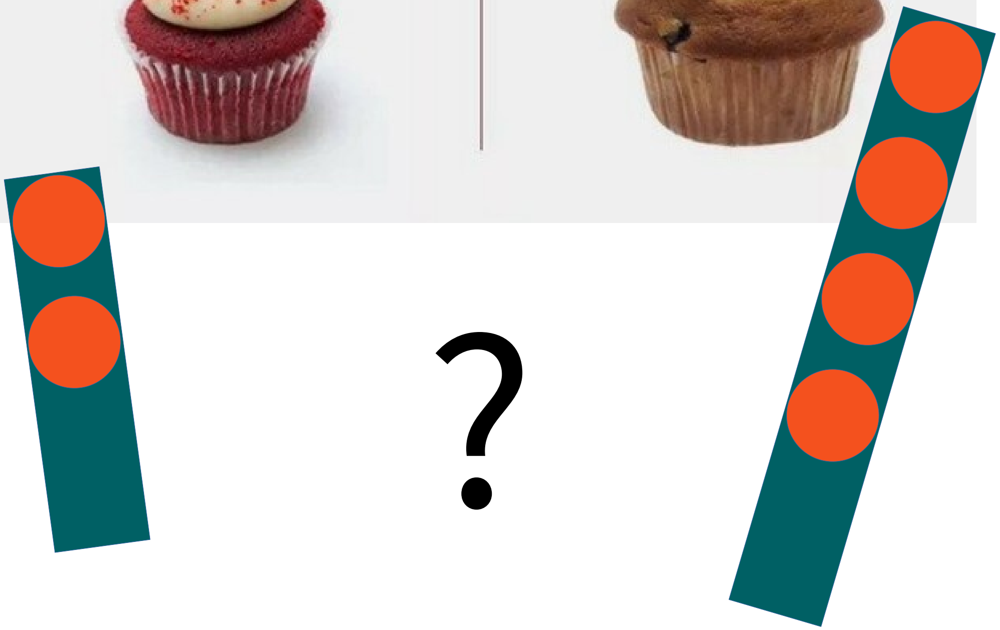




Jagged data. Structure of micro features

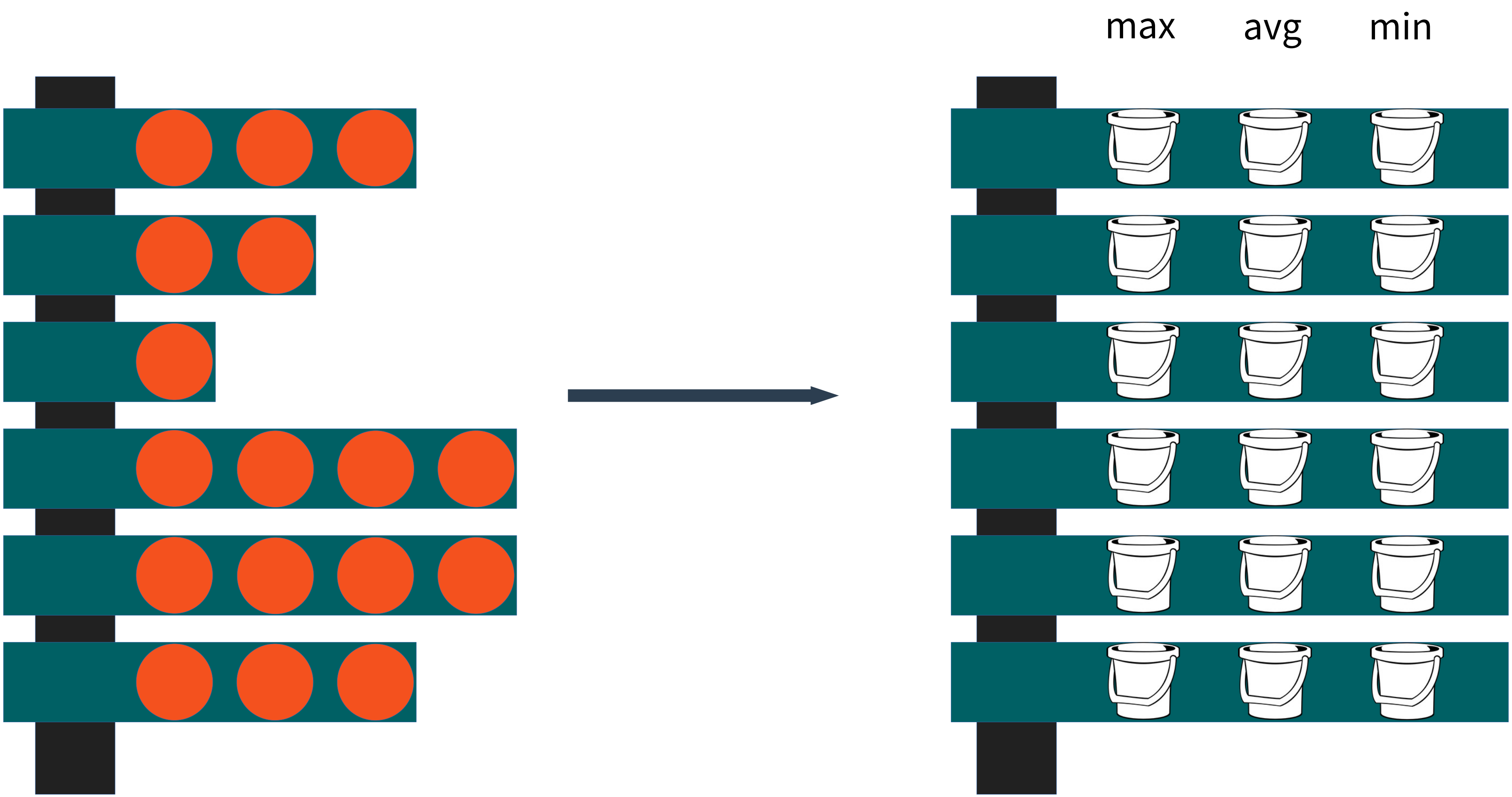


How similar are them?

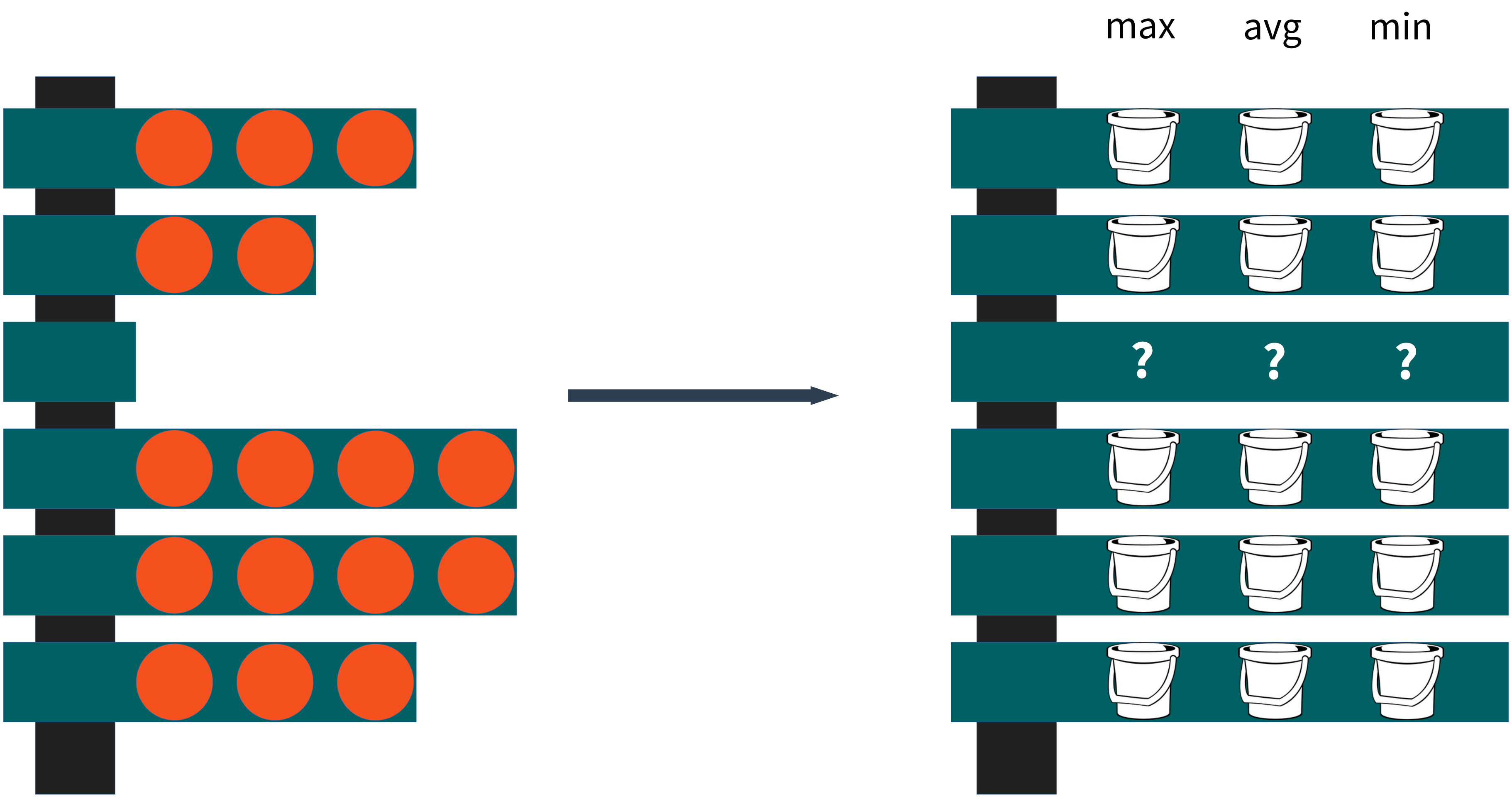


?

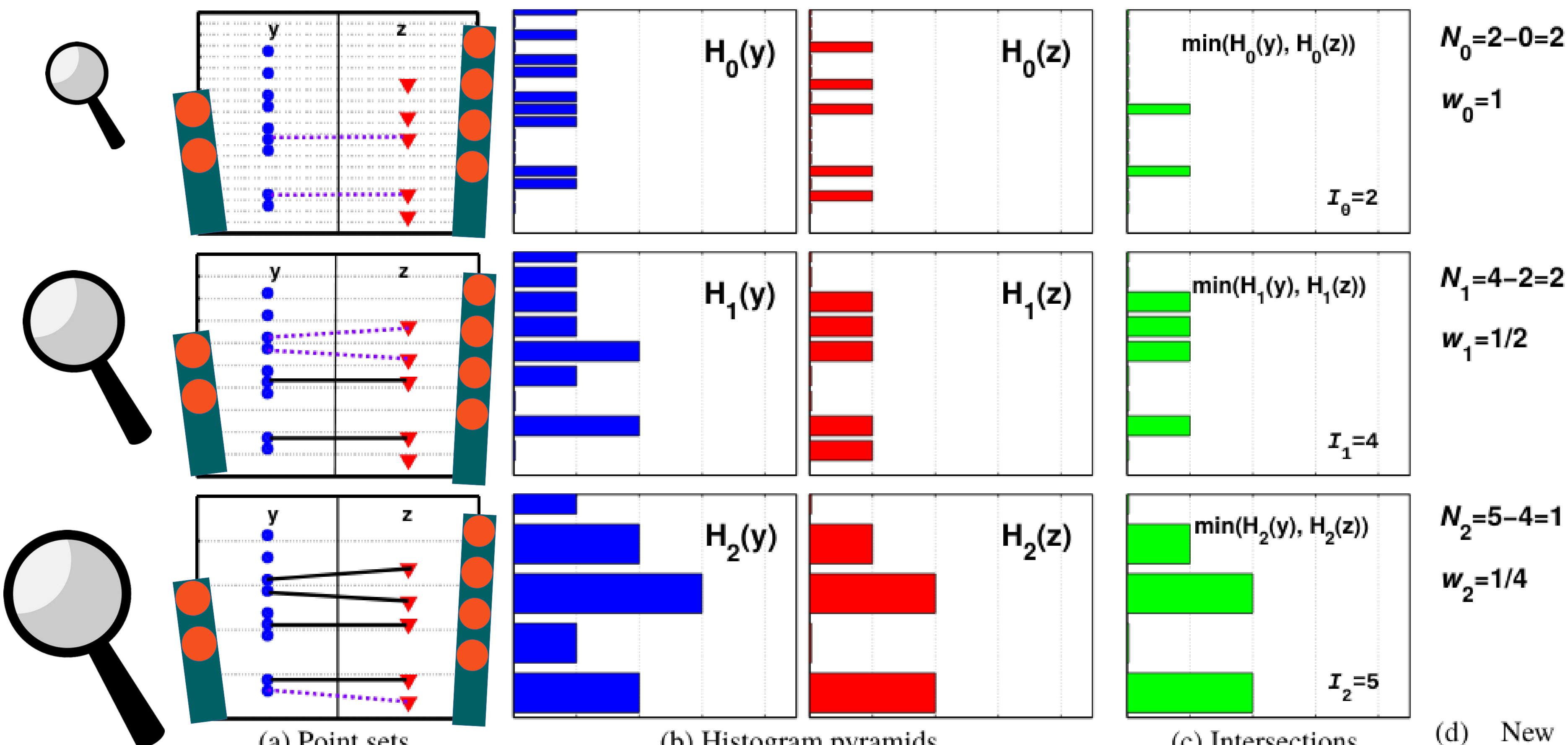
Option: Aggregation



Option: Aggregation



Option: Set Kernels (Pyramid match)



(a) Point sets (b) Histogram pyramids (c) Intersections (d) New matches

Reference: http://jmlr.csail.mit.edu/papers/volume8/grauman07a/grauman07a.pdf

Option: Set Kernels

+ Operation in non-linearly transformed feature space at no cost (Kernel trick)

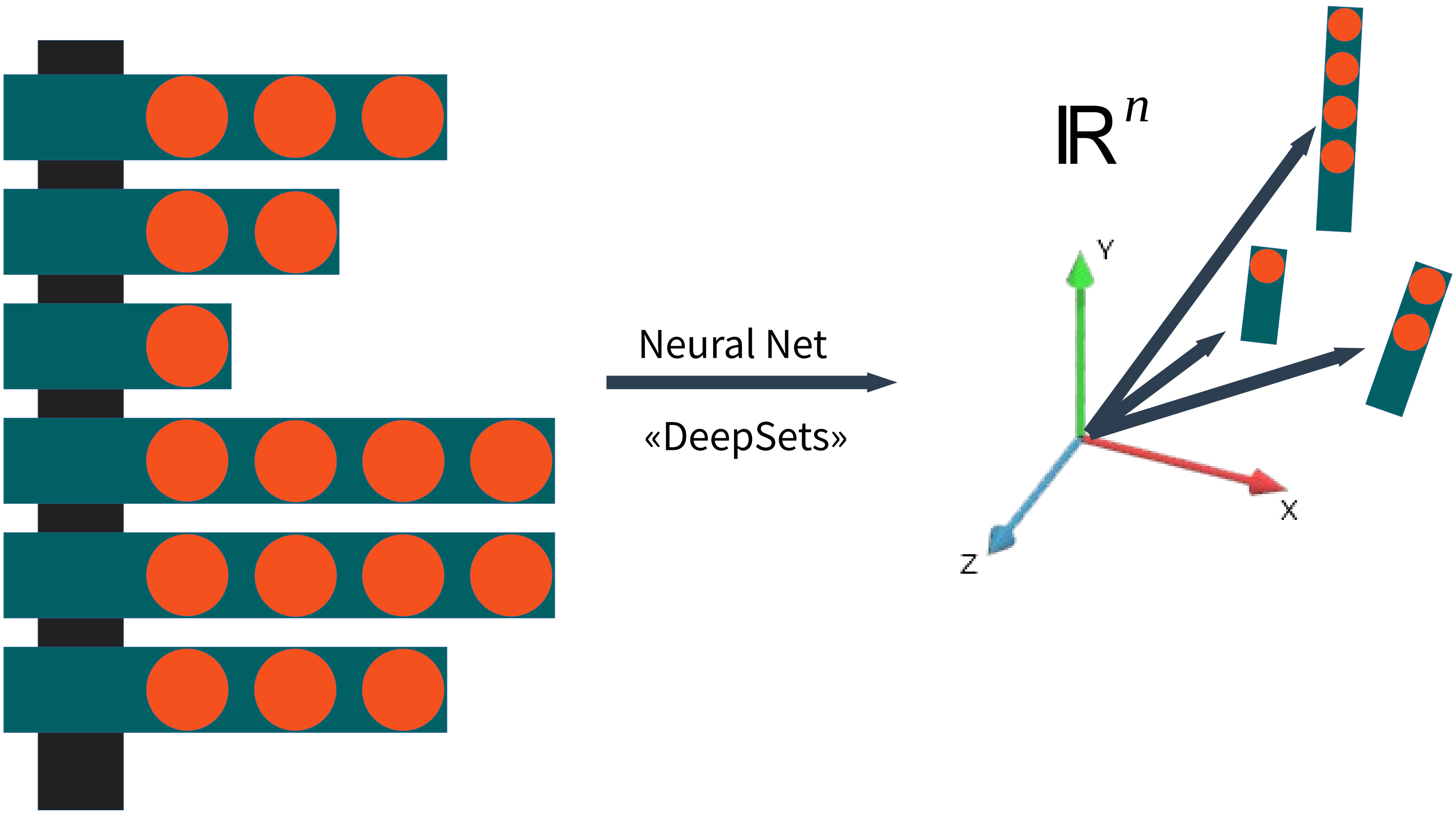
  - Interpretable. Not a black box.

− Compute and store N x 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.

Reference: *StackExchange «What are the limitations of kernel methods?»*

Option: Embedding with NN «DeepSets»



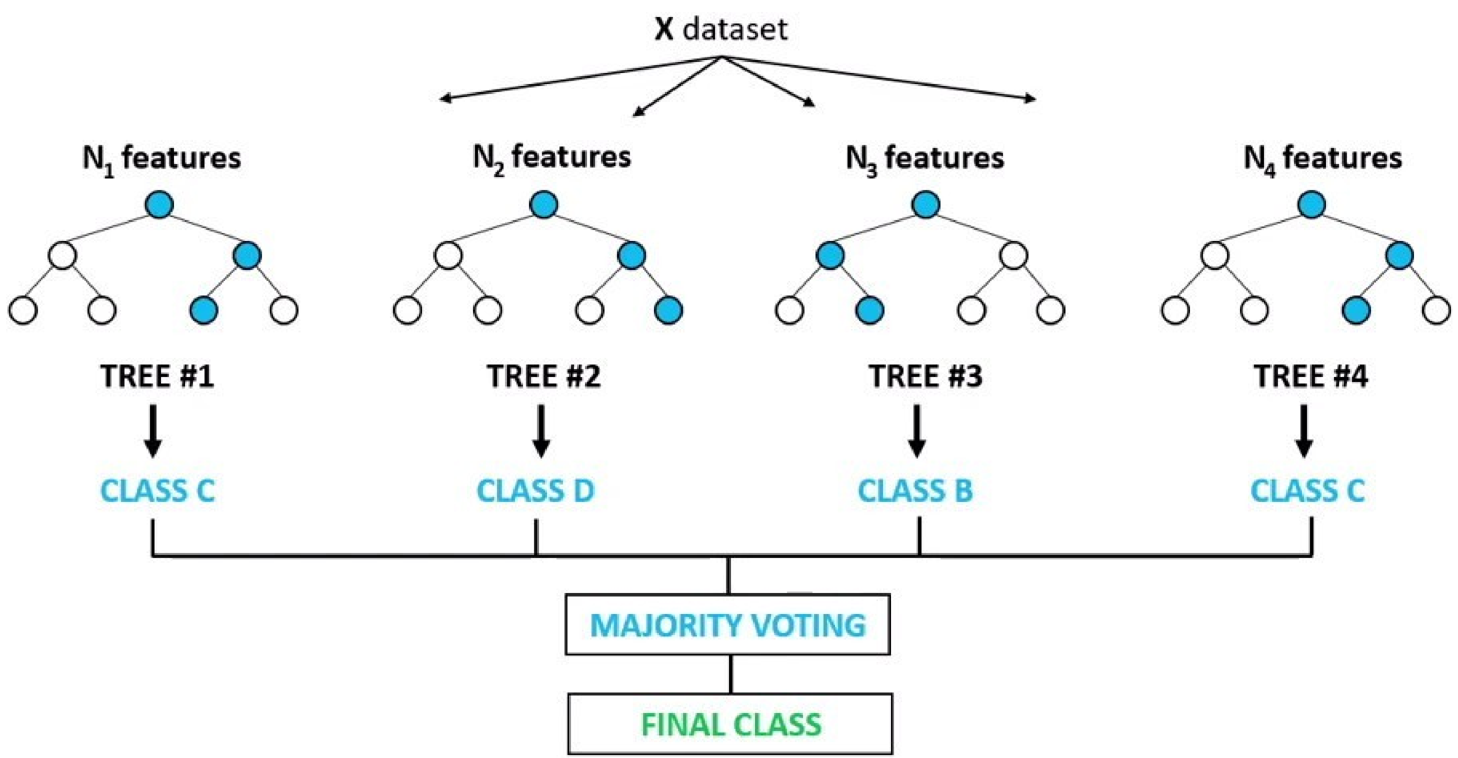
Reference: https://arxiv.org/abs/1703.06114

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

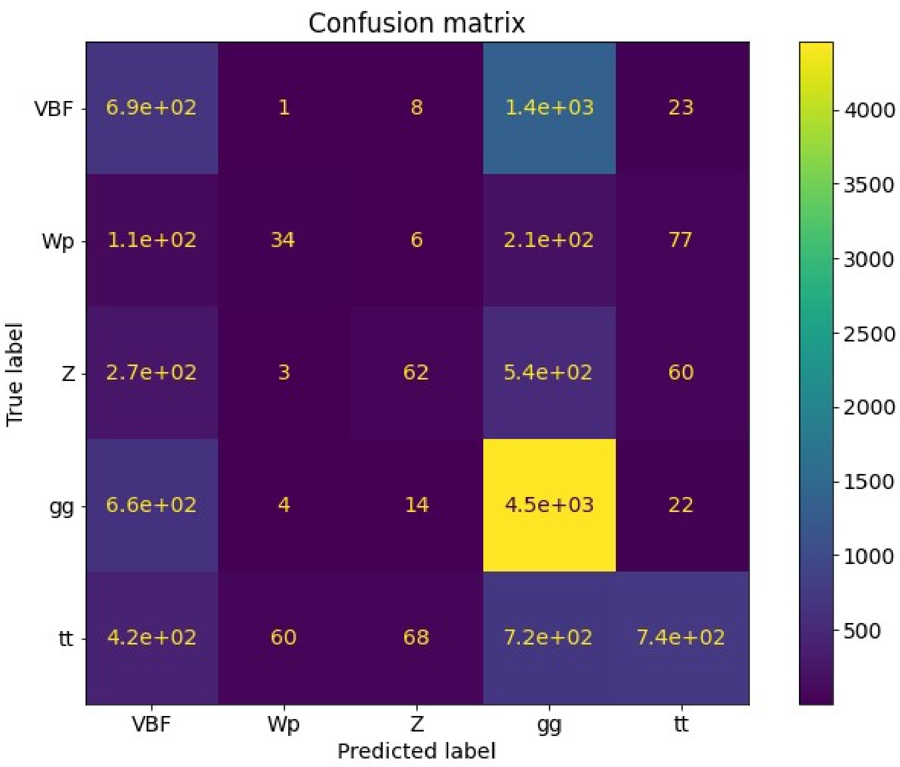
Reference: *StackExchange «What are the limitations of kernel methods?»*

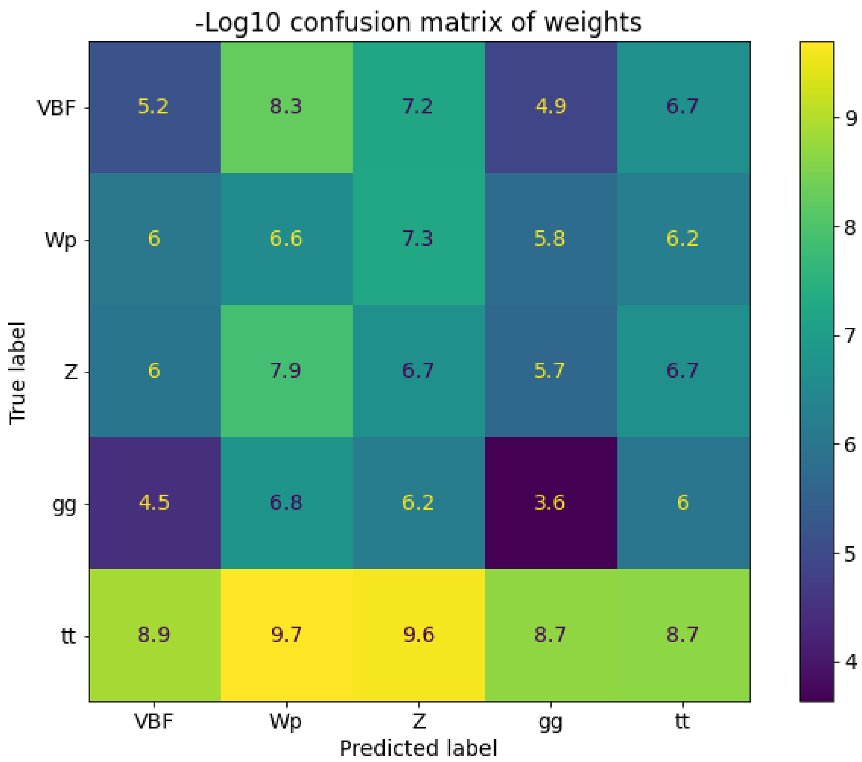
Baseline classifier: Random forest



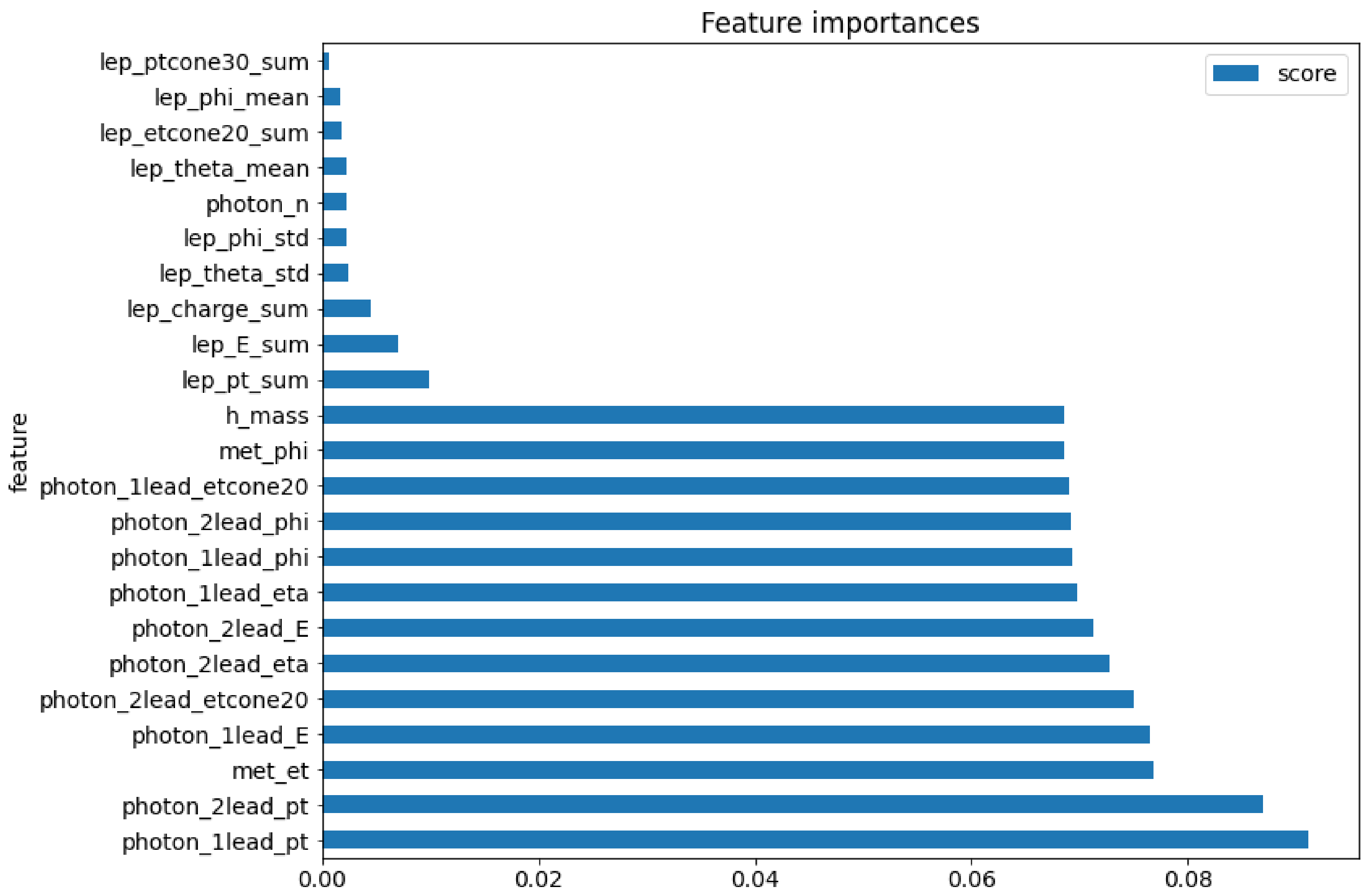
Reference: https://www.wikiwand.com/en/Random_forest#From_bagging_to_random_forests

Baseline classifier: Random forest

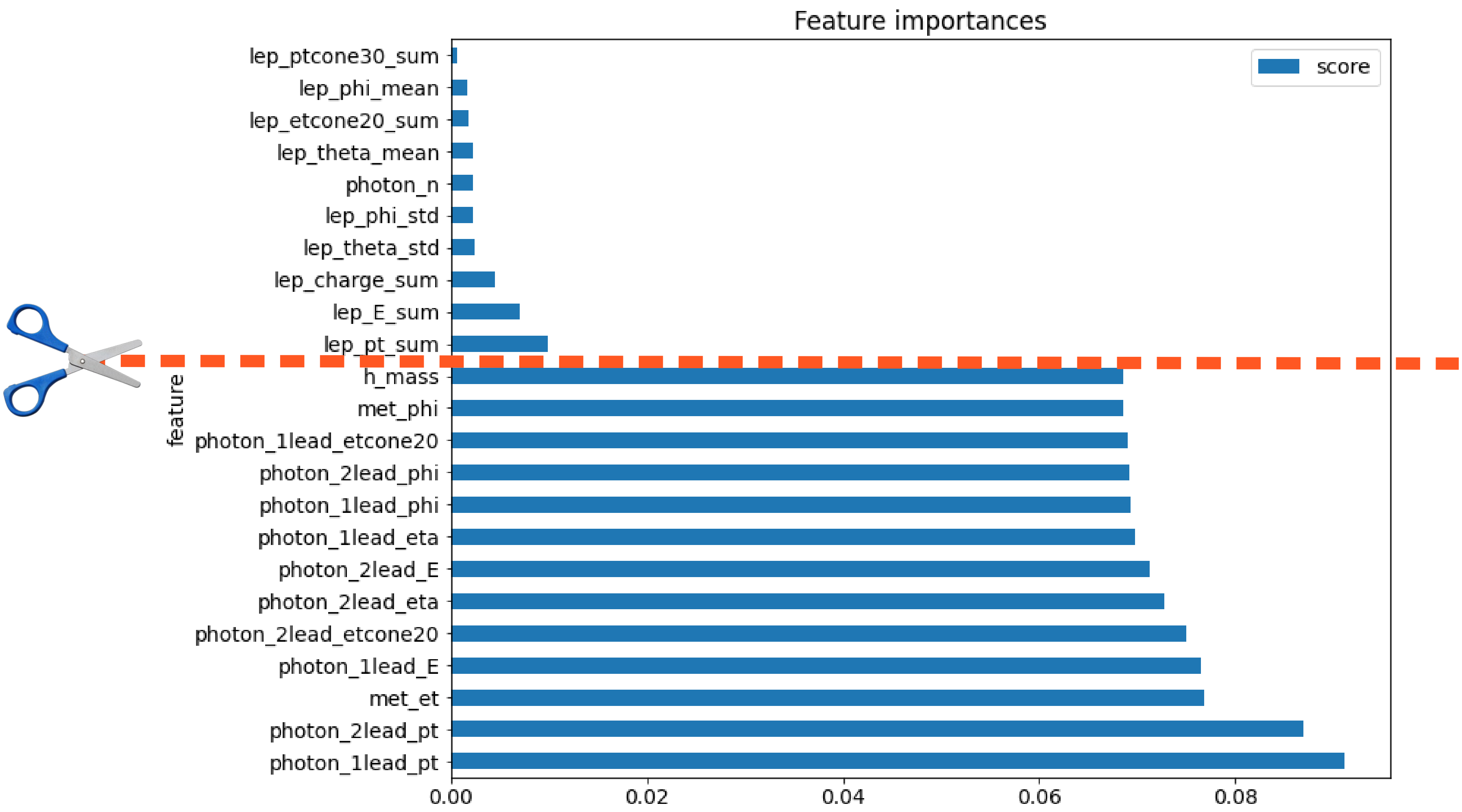




Baseline classifier: Random forest



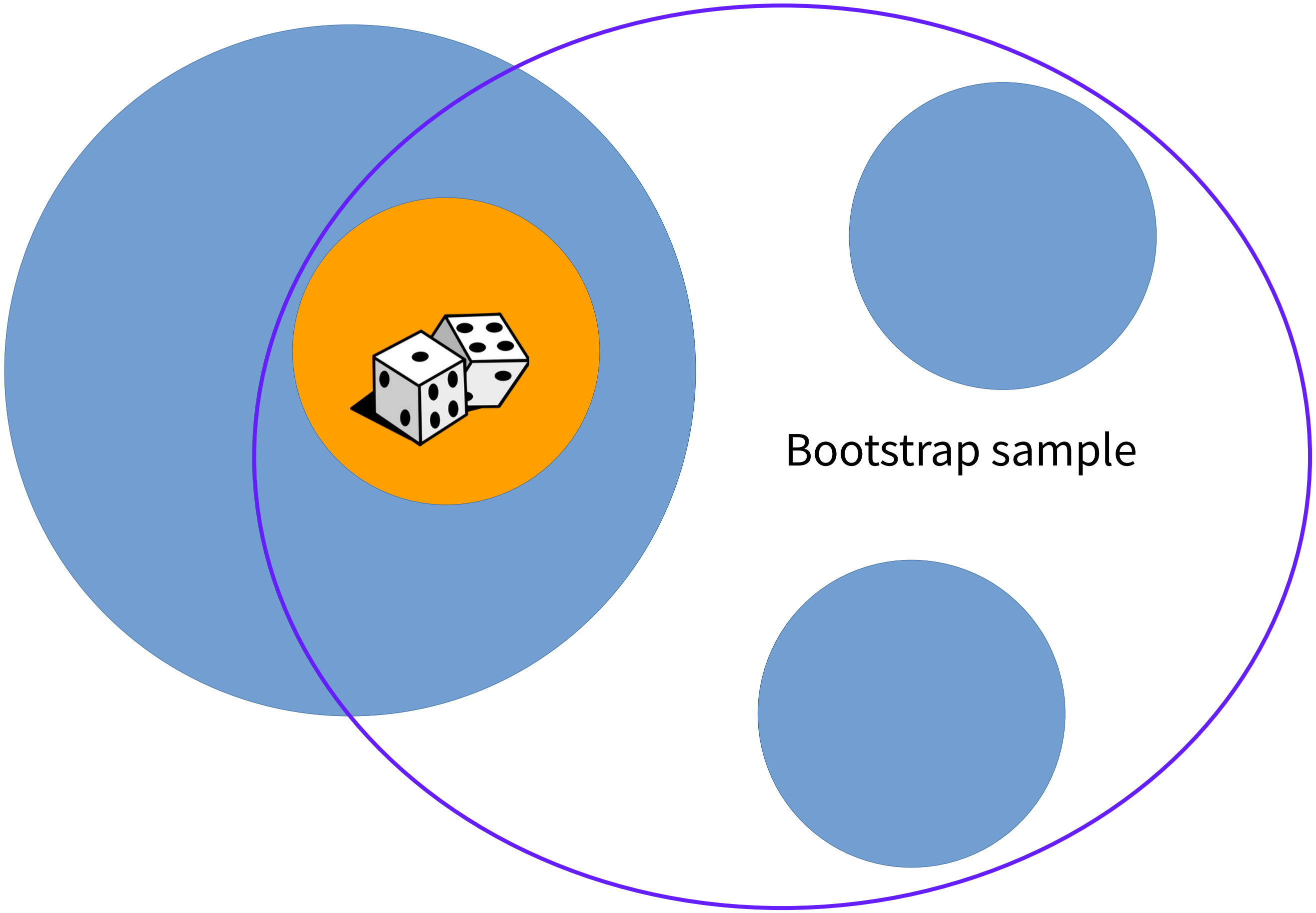
Baseline classifier: Random forest



Imbalanced data

	event_num	weight_sums	weight_frac	weight_per_event
VBF	10369	9.612055e-05	0.064293	9.269992e-09
Wp	2112	1.702021e-05	0.011384	8.058813e-09
Z	4571	1.697019e-05	0.011351	3.712578e-09
gg	26258	1.364896e-03	0.912952	5.198020e-08
tt	9861	2.851057e-08	0.000019	2.891245e-12

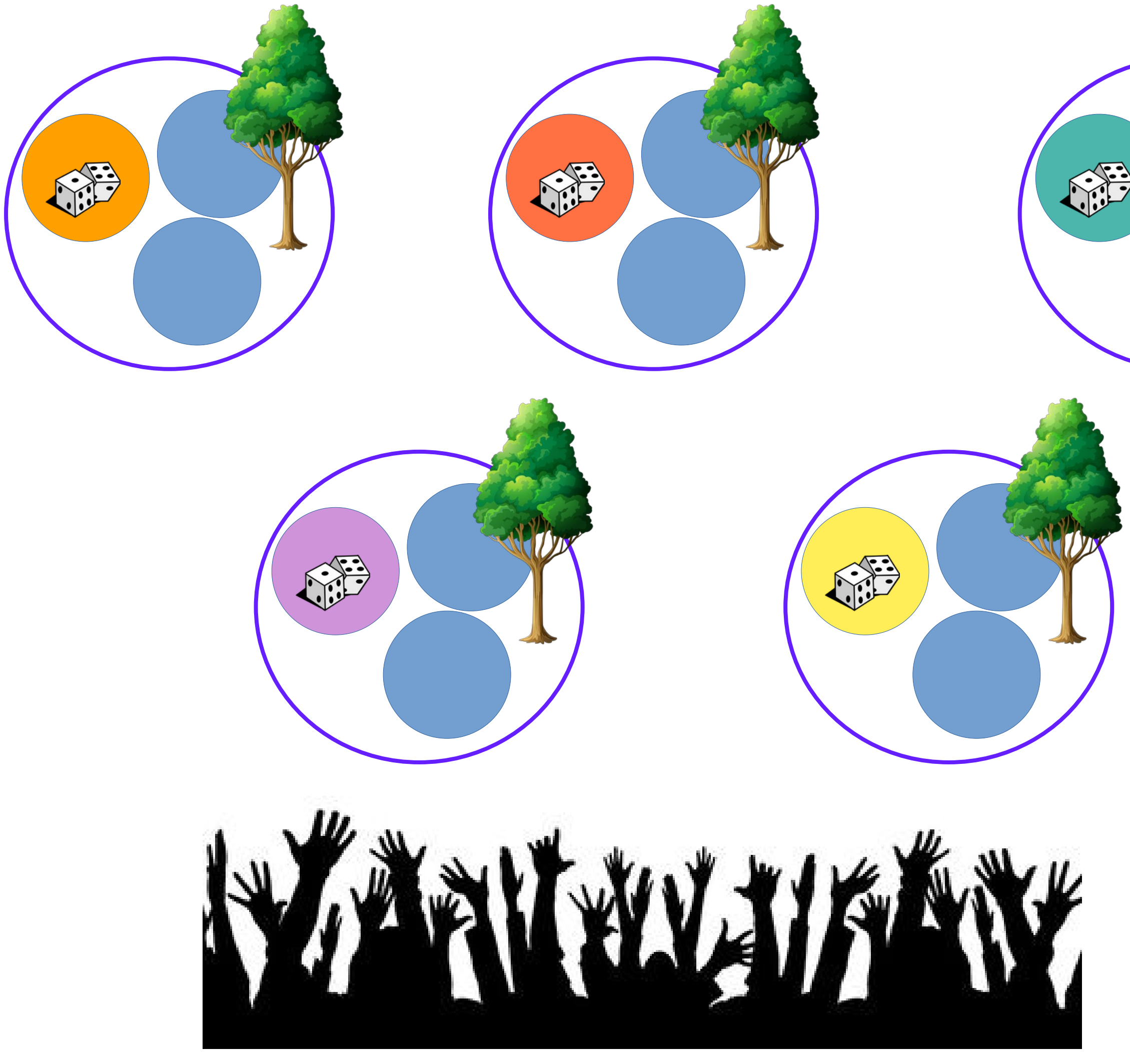
Fighting imbalanced data with bagging



Undersampling

Reference: «Bagging and Random forest for imbalanced classification»

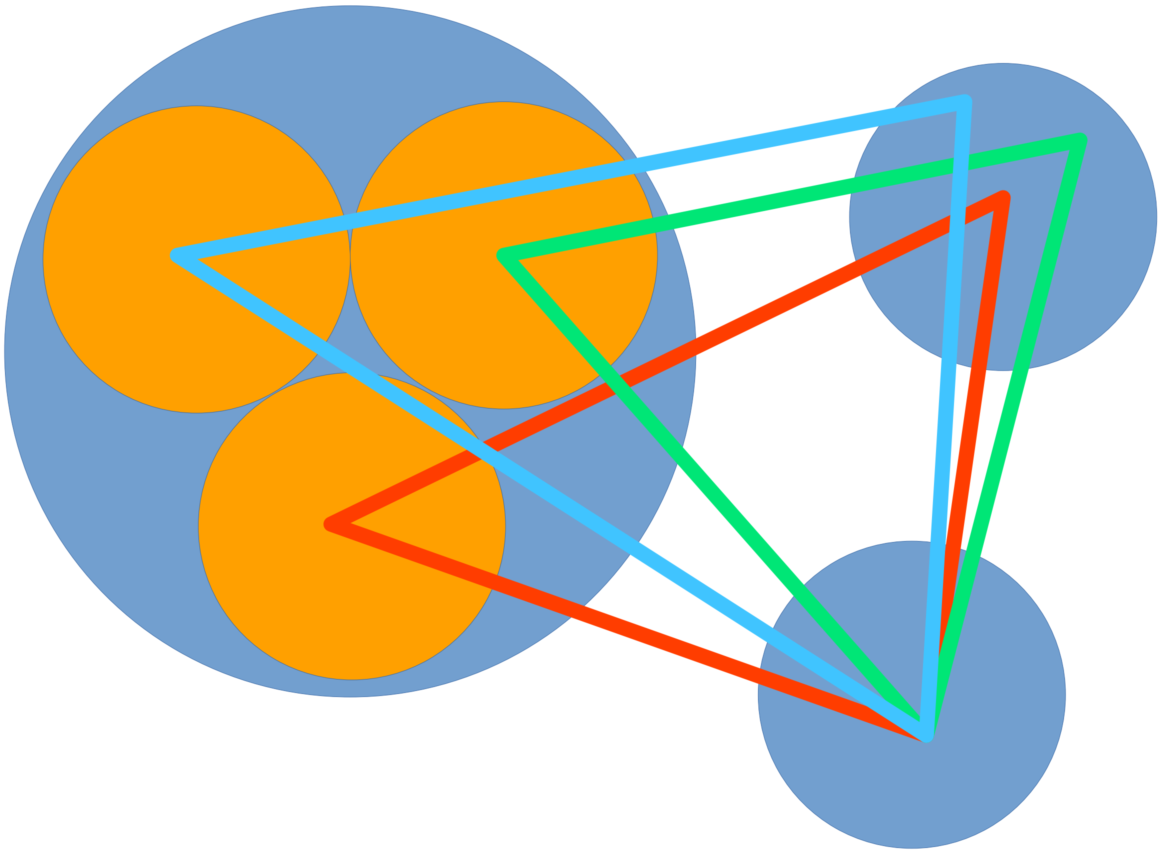
Fighting imbalanced data with bagging



Bagging

Reference: «Bagging and Random forest for imbalanced classification»

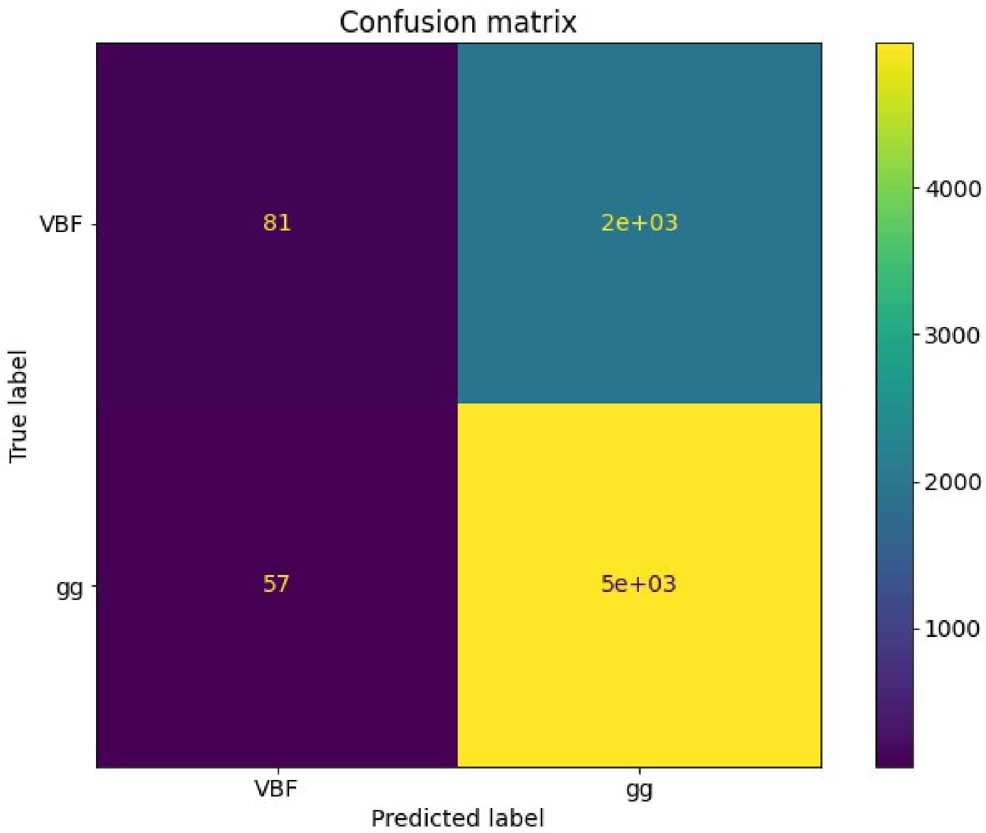
Fighting imbalanced data with bagging

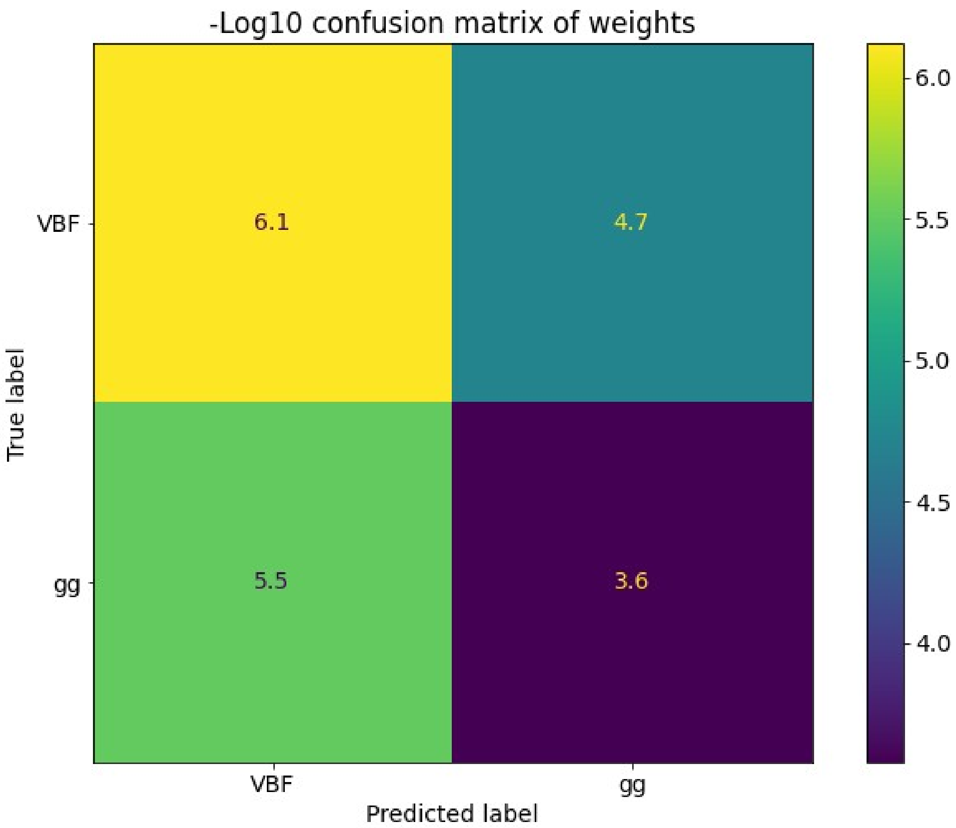


EasyEnsemble

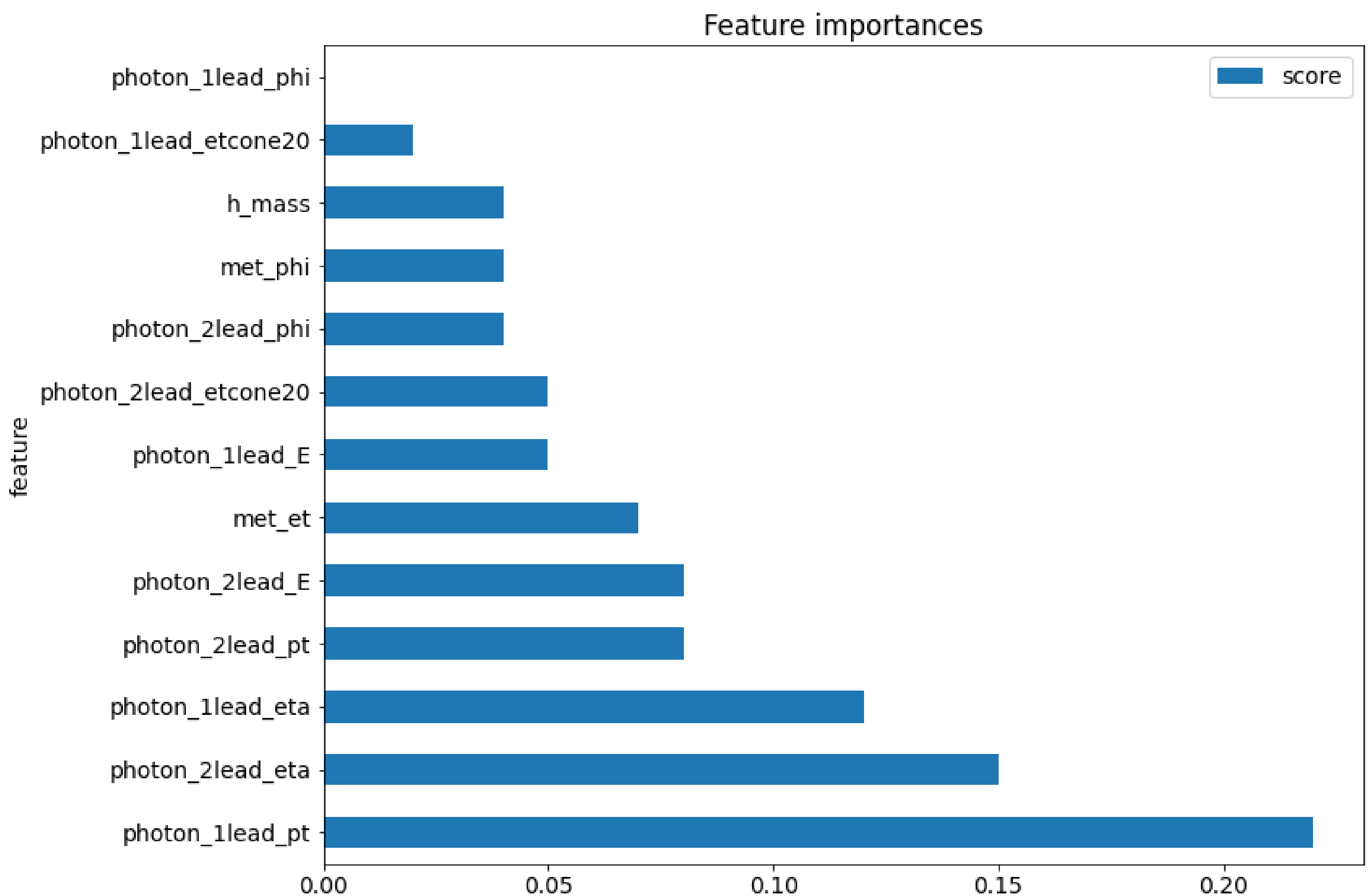
Reference: «Bagging and Random forest for imbalanced classification»

EasyEnsemble model



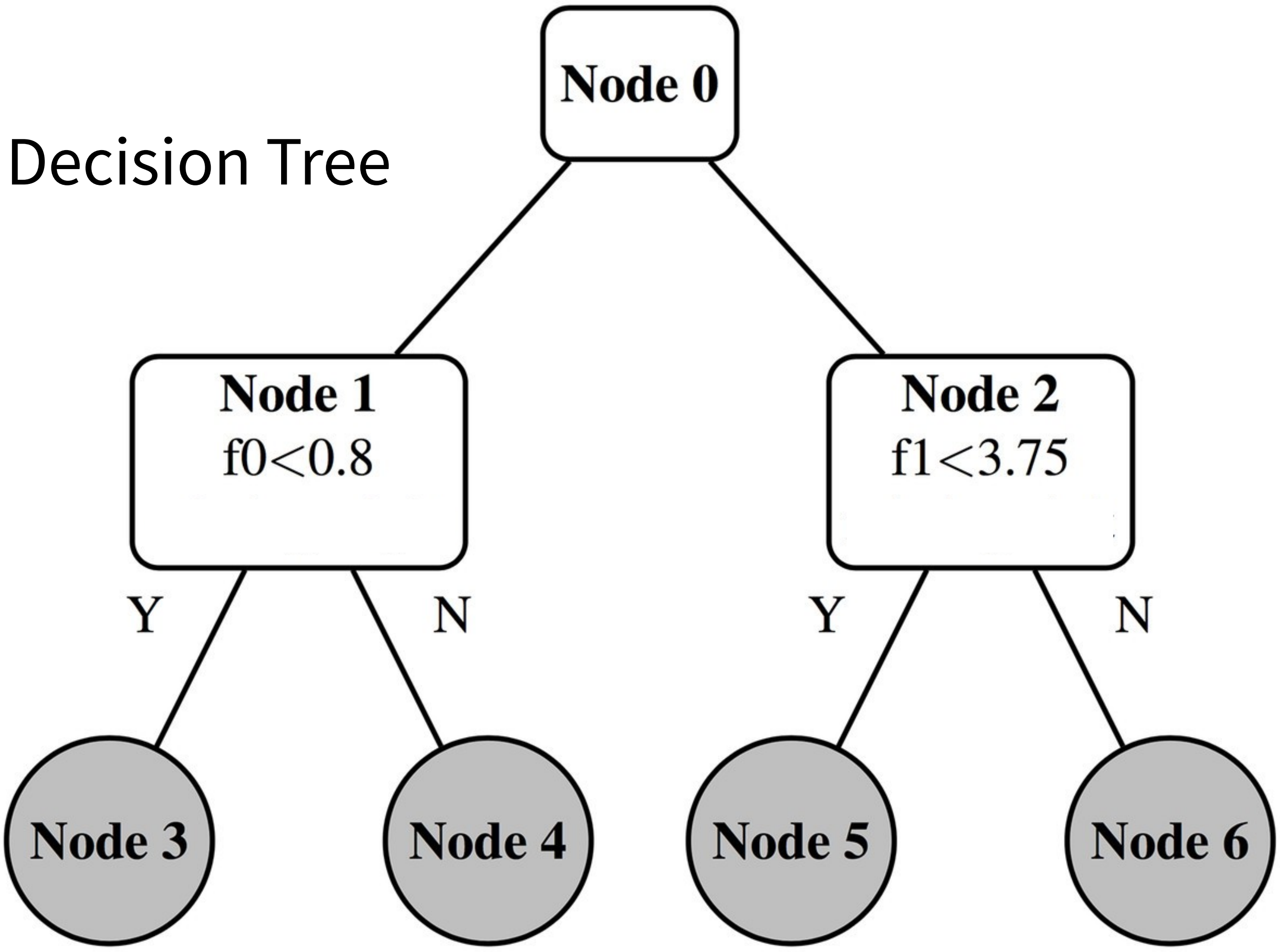


EasyEnsemble model



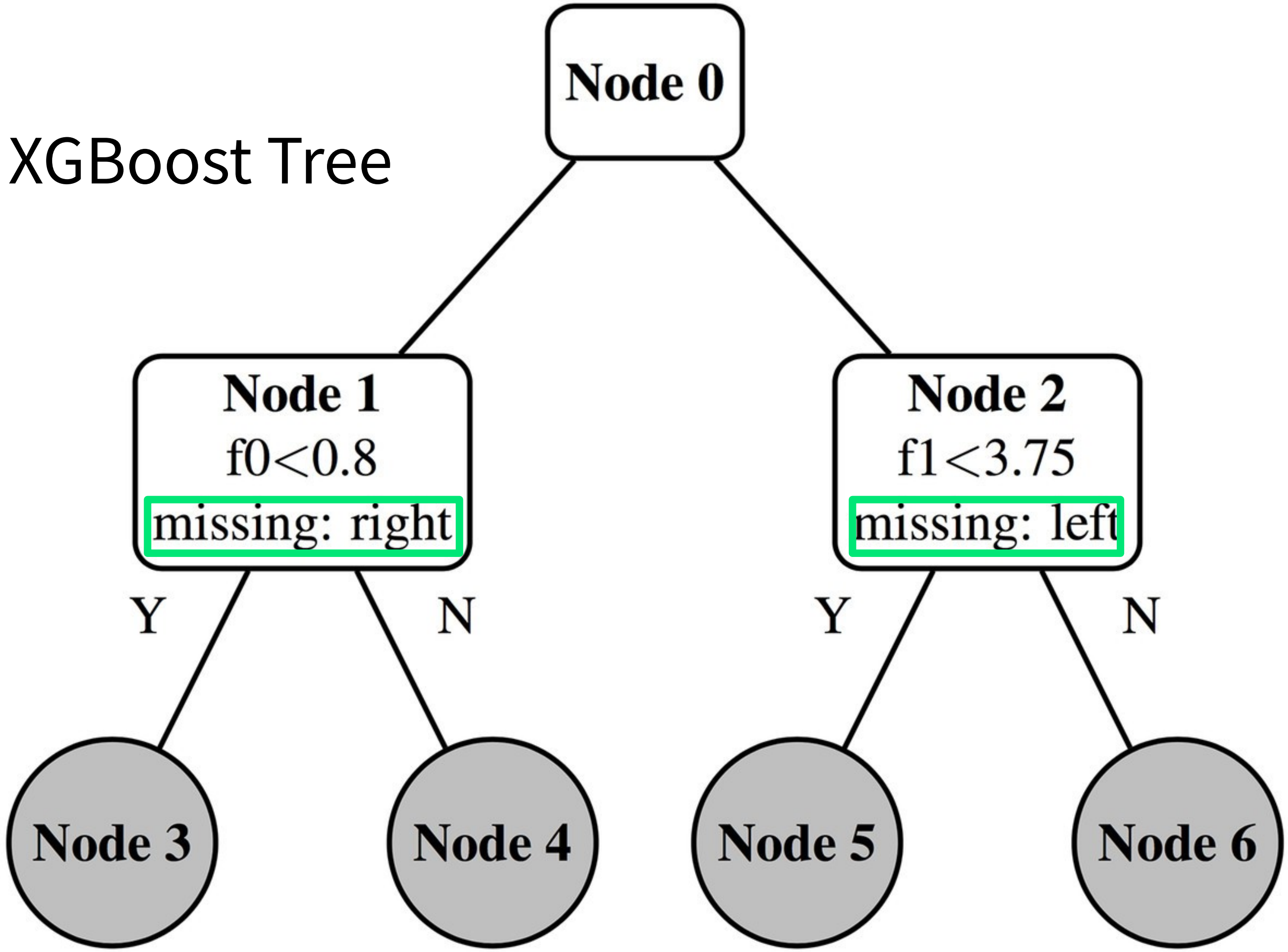
Alternative to downsampling. XGBoost

Decision Tree



Alternative to downsampling. XGBoost

XGBoost Tree

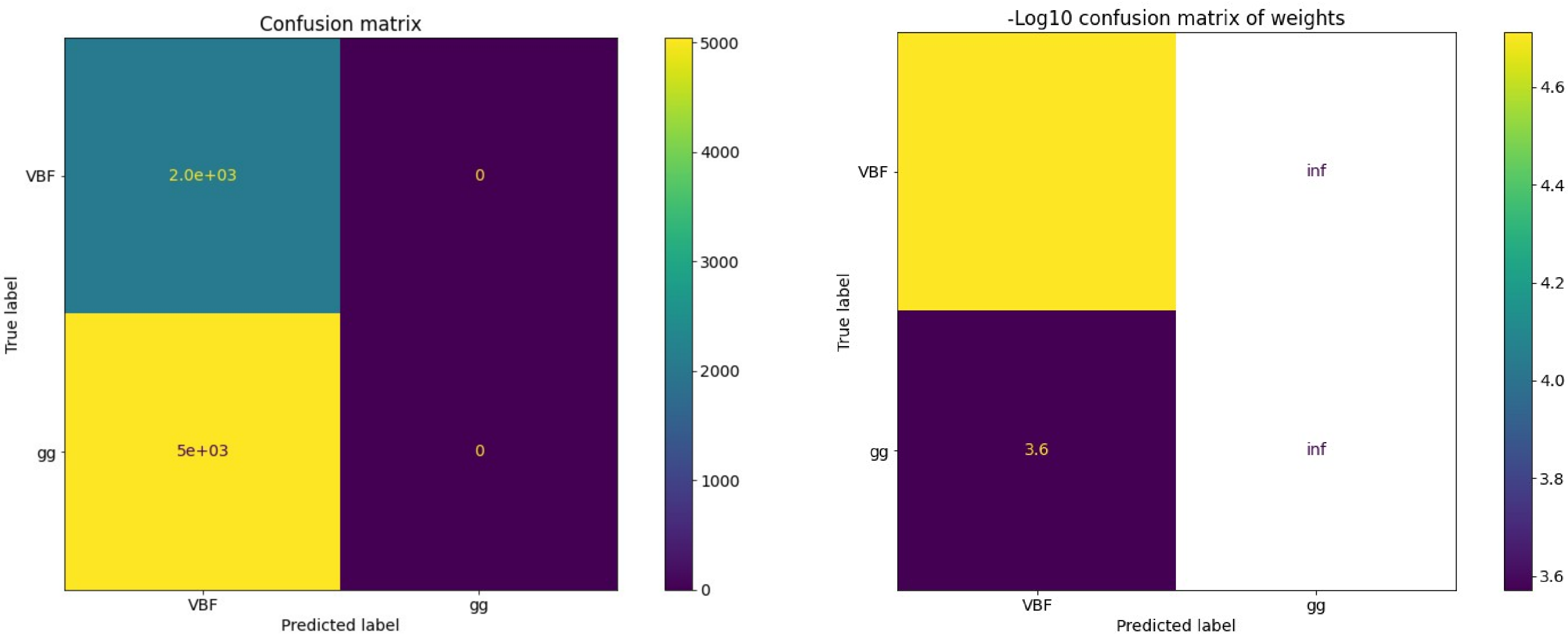


Conclusion

- 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)
- HEP datasets are imbalanced. Possible solutions include:
  - downsampling
  - oversampling
  - missing data distribution fit (xgboost)

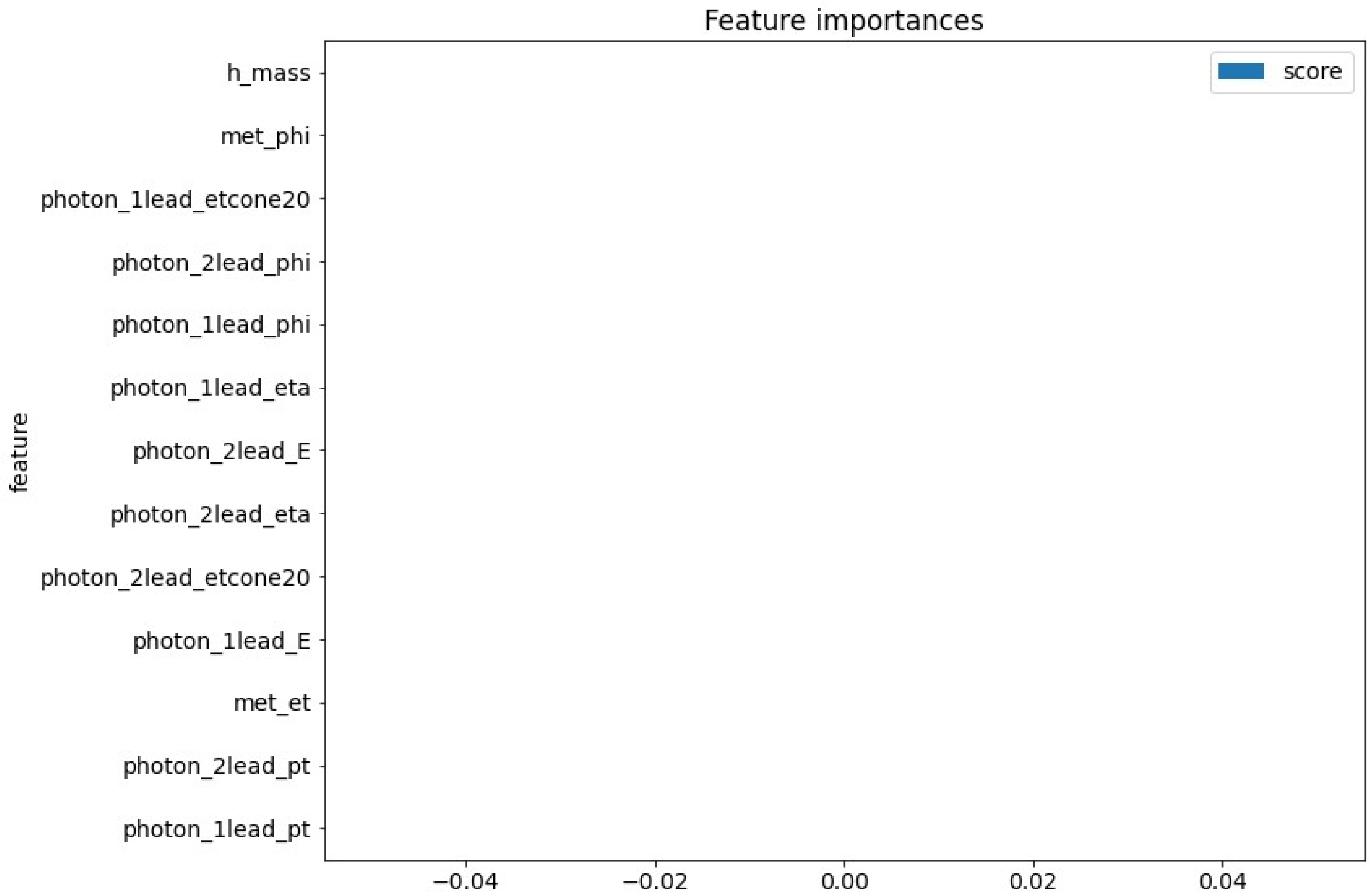
Backup

XGBoost



Confusion matrix
True label
Predicted label
VBF
gg
2.0e+03
0
5e+03
0
5000
4000
3000
2000
1000
0
-Log10 confusion matrix of weights
inf
3.6
inf
4.6
4.4
4.2
4.0
3.8
3.6

XGBoost



Feature importances
score
h_mass
met_phi
photon_1lead_etcone20
photon_2lead_phi
photon_1lead_phi
photon_1lead_eta
photon_2lead_E
photon_2lead_eta
photon_2lead_etcone20
photon_1lead_E
met_et
photon_2lead_pt
photon_1lead_pt
feature
−0.04
−0.02
0.00
0.02
0.04