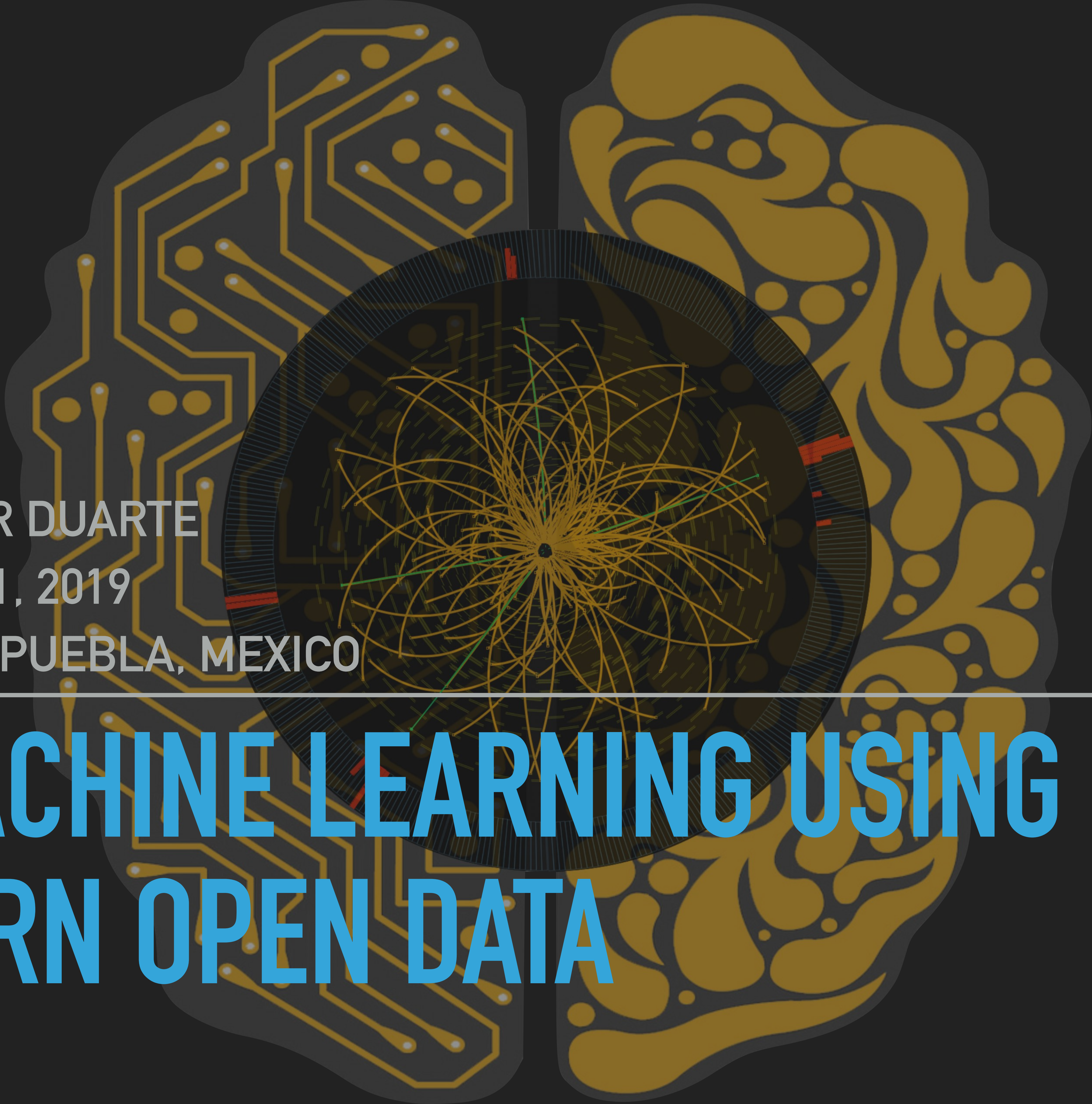


JAVIER DUARTE

MAY 21, 2019

LHCP, PUEBLA, MEXICO

MACHINE LEARNING USING CERN OPEN DATA

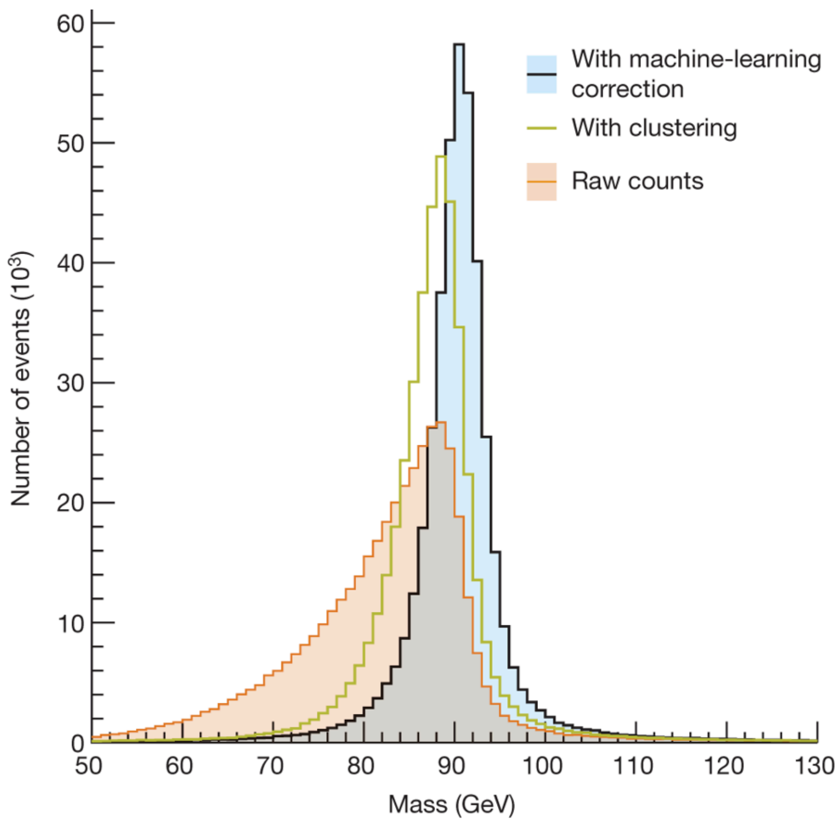
OUTLINE

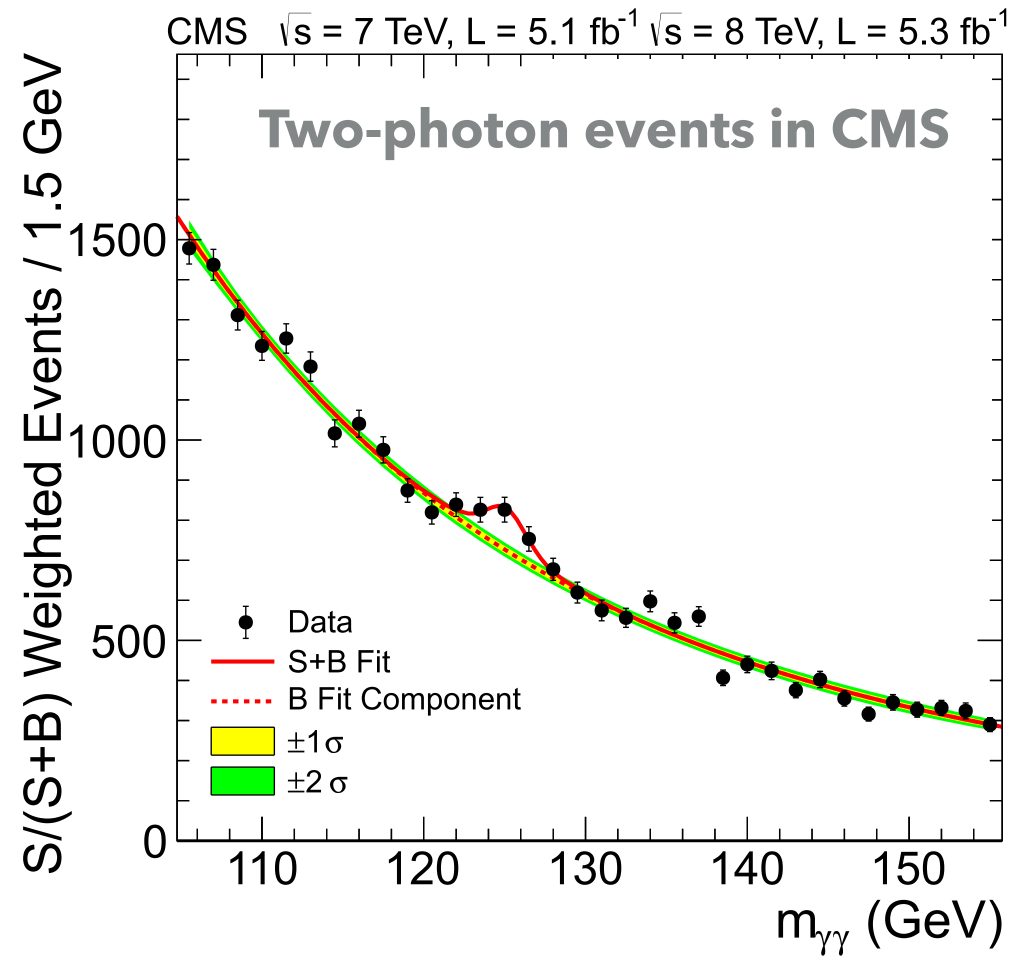
- High energy physics + machine learning
  - CMS open data as ML reference datasets & challenge
- Examples of ML using CMS open data
- ML-dedicated open data release
  - Pixel tracking studies
  - Higgs to bb tagging
- Summary and outlook

HIGH ENERGY PHYSICS + MACHINE LEARNING

Nature 560, 41 (2018)

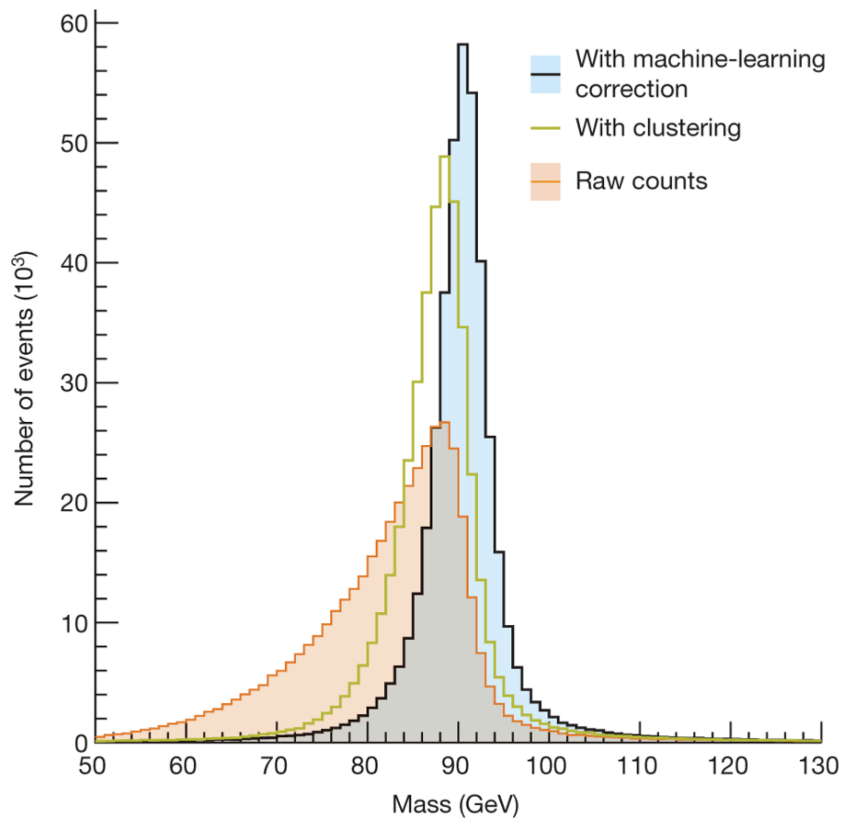
- **Machine learning** was vital to make big discoveries like the Higgs boson on July 4, 2012

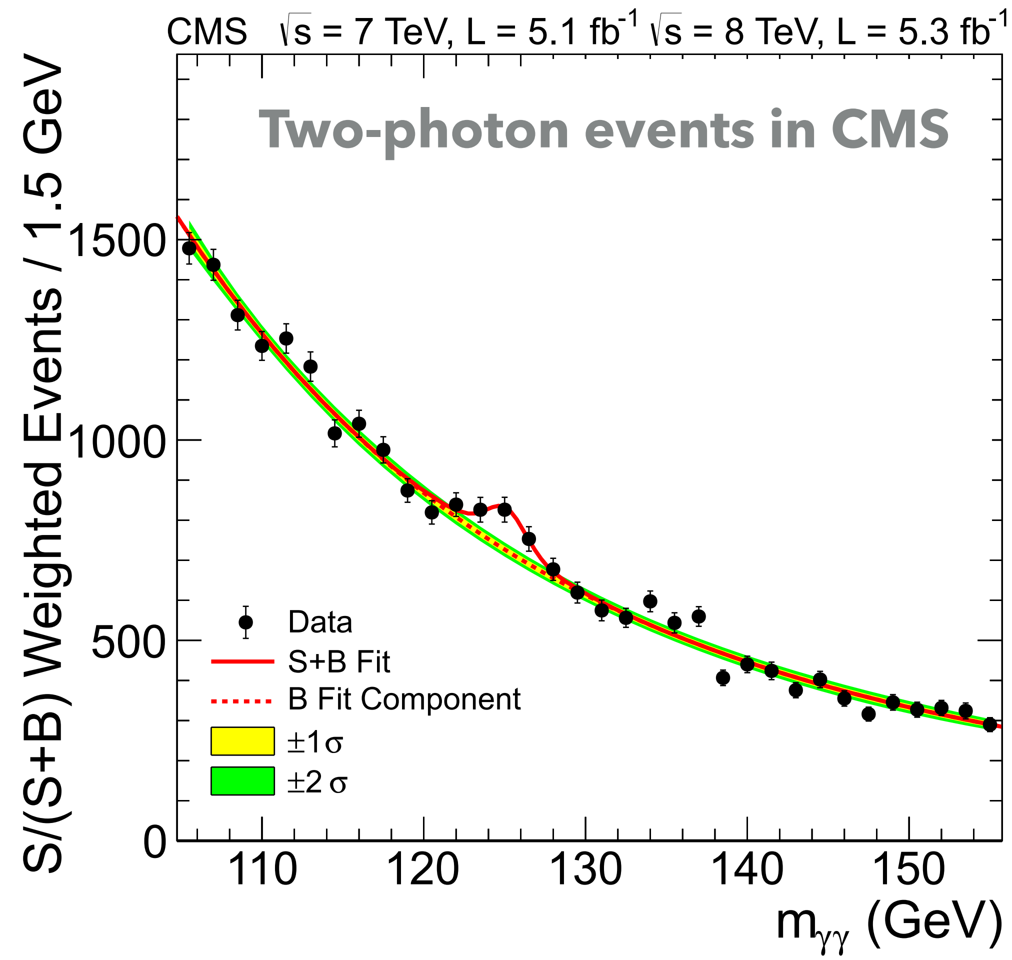


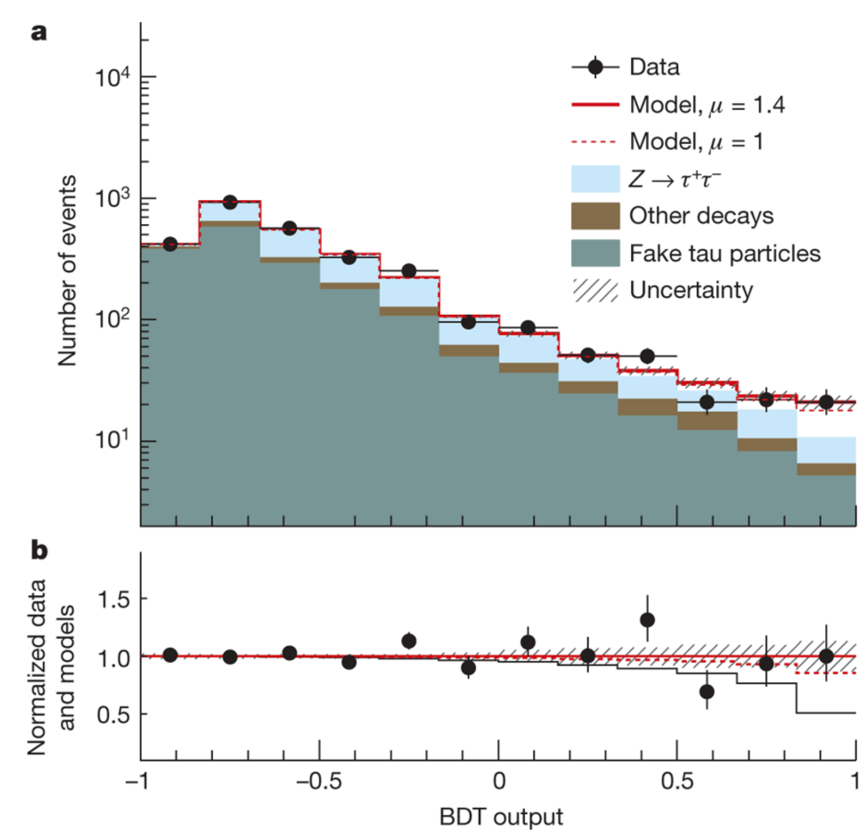


HIGH ENERGY PHYSICS + MACHINE LEARNING

- **Machine learning** was vital to make big discoveries like the Higgs boson on July 4, 2012







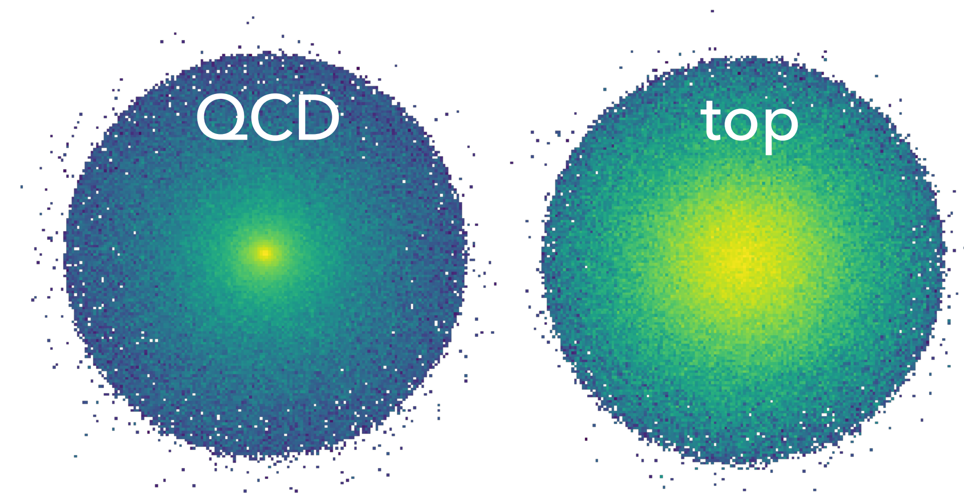
- Today, ML is **enabling** new reconstruction, particle identification, measurements, and searches never thought possible at the LHC

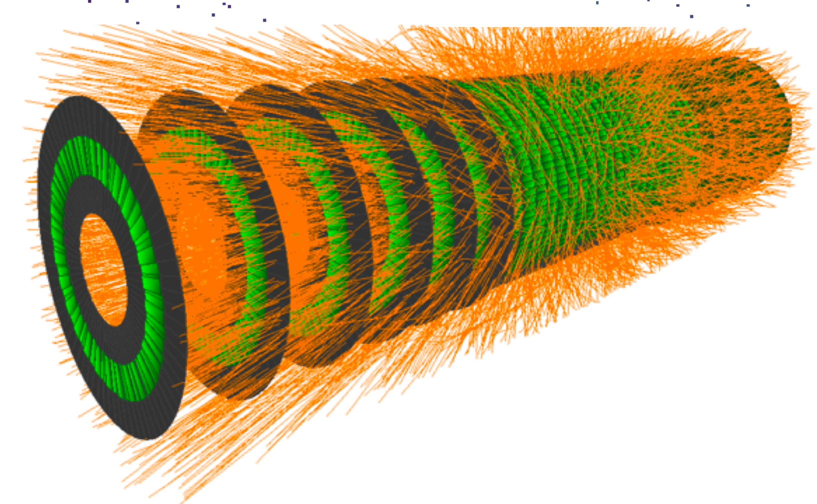
REFERENCE DATA SETS FOR HEP + ML

- Engage ML community for interesting, realistic tasks in experimental HEP

REFERENCE DATA SETS FOR HEP + ML

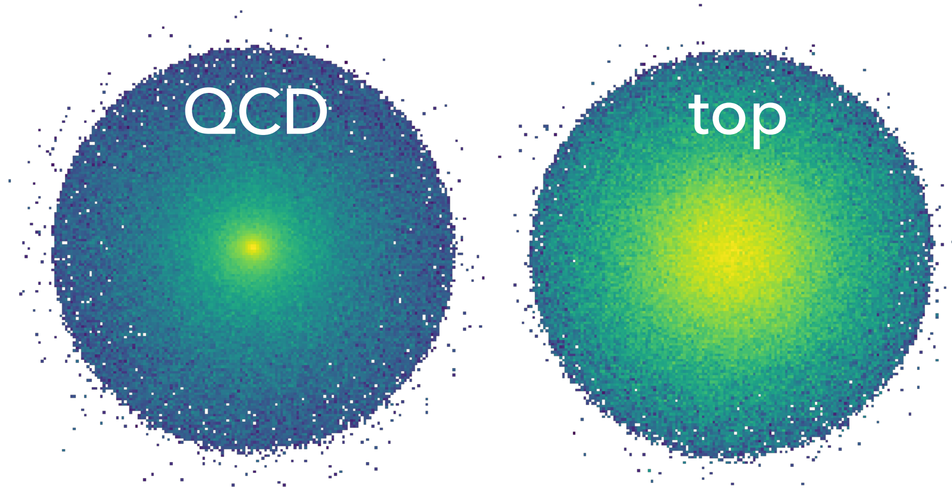
- Engage ML community for interesting, realistic tasks in experimental HEP
- Calls at ML4Jets and Connecting the Dots workshops for more public HEP data sets with real detector simulation for ML applications
  - Example: data set for top tagging based on Pythia+Delphes
  - Example: data set for tracking based on ACTS (kaggle TrackML challenge)



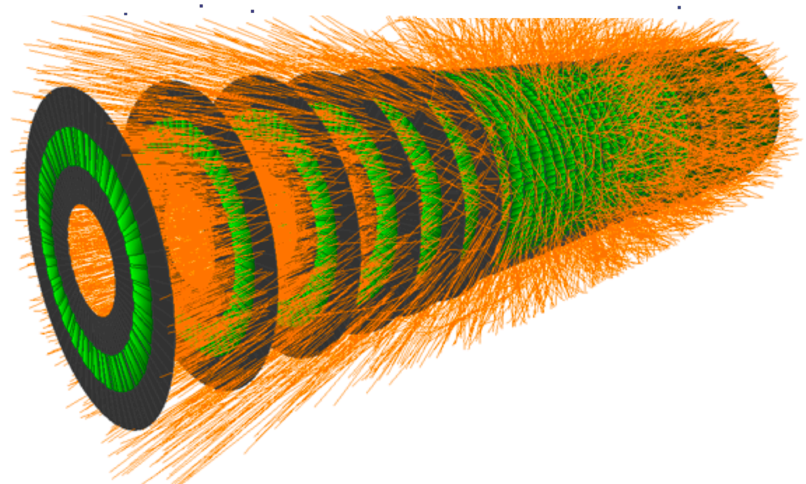


REFERENCE DATA SETS FOR HEP + ML

- Engage ML community for interesting, realistic tasks in experimental HEP
- Calls at ML4Jets and Connecting the Dots workshops for more public HEP data sets with real detector simulation for ML applications
  - Example: data set for top tagging based on Pythia+Delphes



  - Example: data set for tracking based on ACTS (kaggle TrackML challenge)



- Can **CMS open data** fill this role for many ML applications?

opendata CERN

http://opendata.cern.ch/

About

Explore more than **1 petabyte** of open data from particle physics!

Start typing... | Search

search examples: collision datasets, keywords:education, energy:7TeV

Explore

datasets
software
environments
documentation

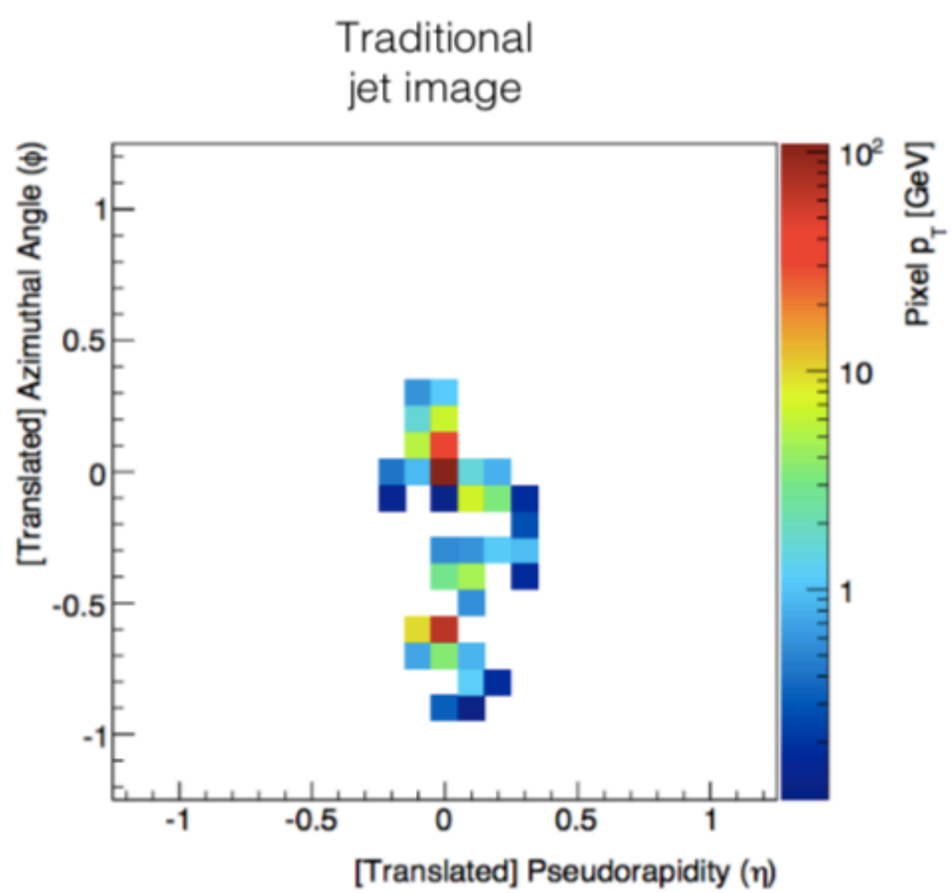
Focus on

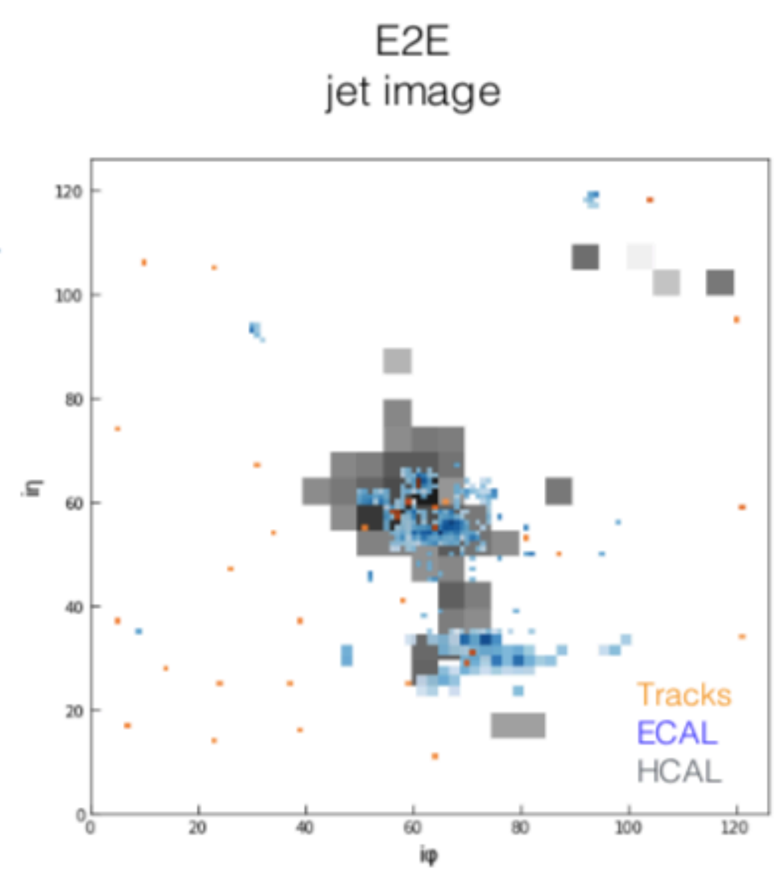
ATLAS
ALICE
CMS
LHCb
OPERA

Get started

ML USING OPEN DATA & SIMULATION

- Open data & simulation is also useful for ML-focused studies, e.g. kaggle ATLAS $H \to \tau\tau$ challenge: http://opendata.cern.ch/record/328
- Most existing efforts based on reducing AOD/MINIAOD samples
  - Requires CMS domain knowledge, CMS software, …
- Convolutional neural networks image-based event classification [arXiv:1708.07034]
- End-to-end physics event classification [arXiv:1807.11916]
- End-to-end jet classification of quarks and gluons [arXiv:1902.08276]





opendata CERN http://opendata-dev.cern.ch/ About

Explore more than **1 petabyte** of open data from particle physics!

Start typing... Search

search examples: collision datasets, keywords:education, energy:7TeV

Explore

datasets
software
environments
documentation

Focus on

ATLAS
ALICE
CMS
LHCb
OPERA
Data Science

Get started

opendata CERN

http://opendata-dev.cern.ch/

About

Explore more than **1 petabyte** of open data from particle physics!

Start typing... Search

search examples: collision datasets, keywords:education, energy:7TeV

Explore

datasets
software
environments
documentation

Focus on

ATLAS
ALICE
CMS
LHCb
OPERA
Data Science

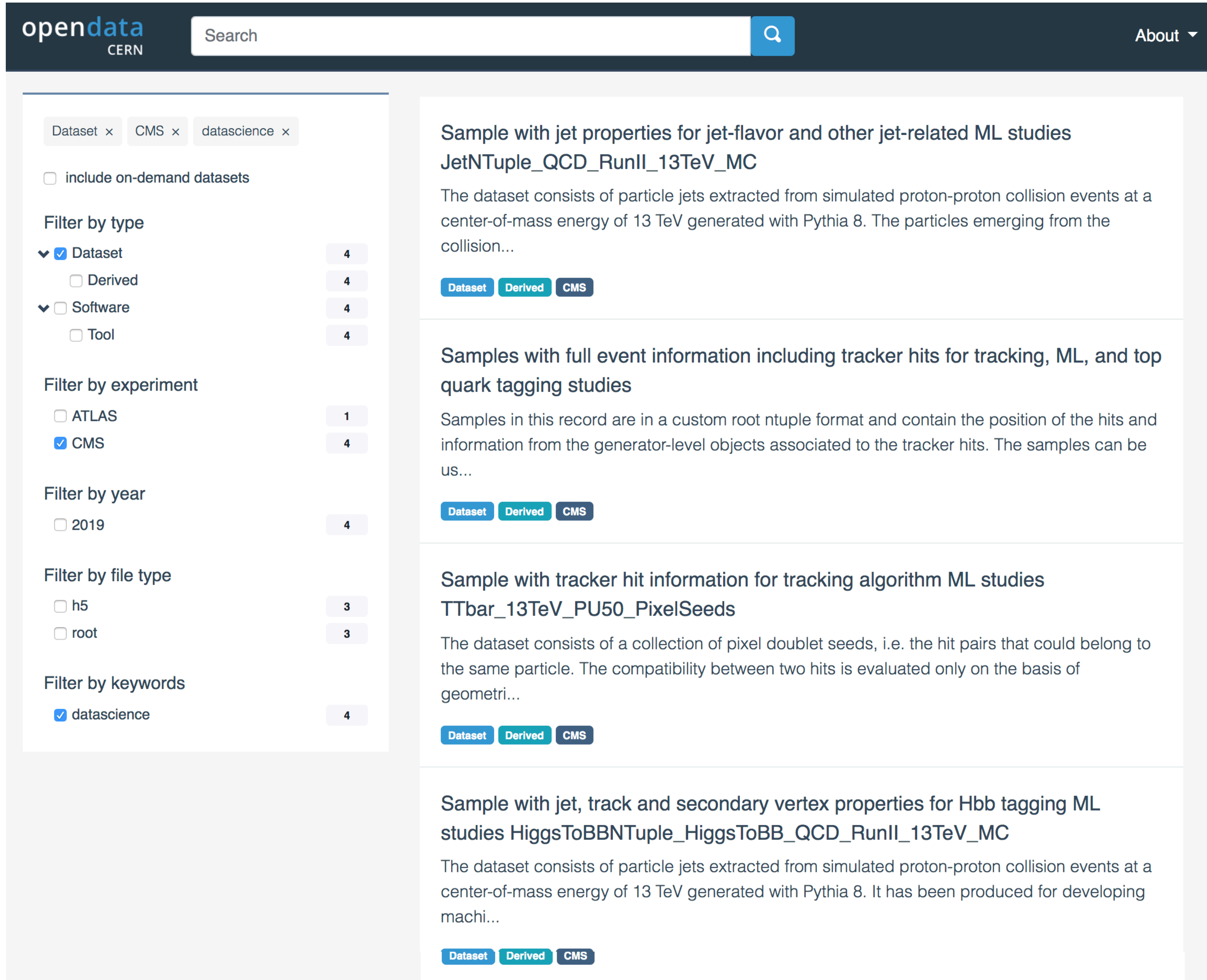
Get started

ALICE ATLAS CMS LHCb OPERA CERN

ML-DEDICATED OPEN DATA RELEASE

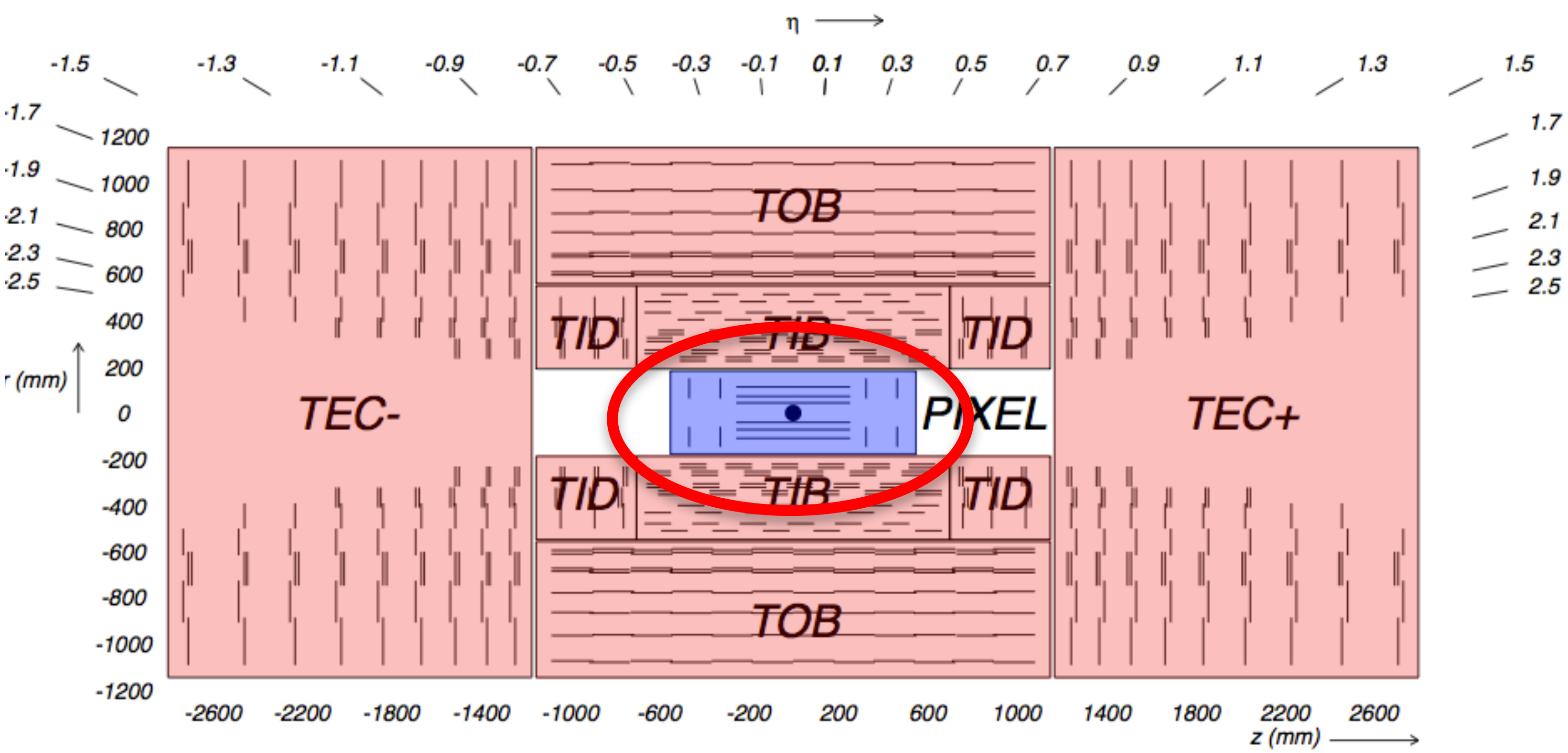
http://opendata-dev.web.cern.ch/search?keywords=datascience&experiment=CMS&type=Dataset

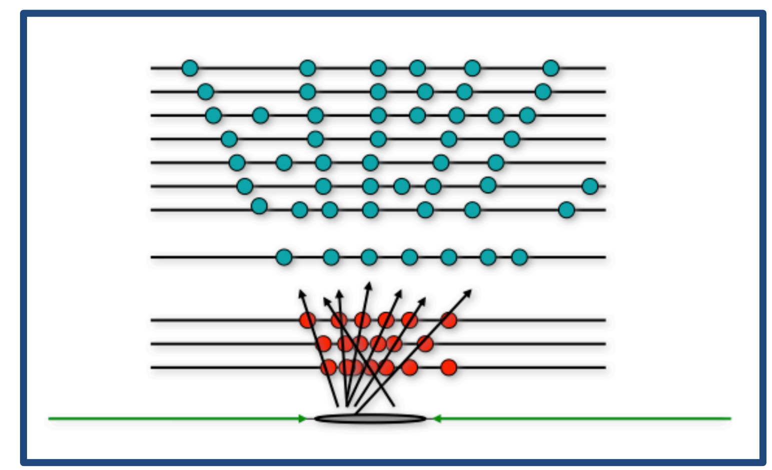
- 4 ***derived*** datasets from official 2016 CMS simulation (ROOT & HDF5)
  - Jet flavor studies
  - Top tagging
  - **Pixel tracking studies**
  - **H(bb) tagging**



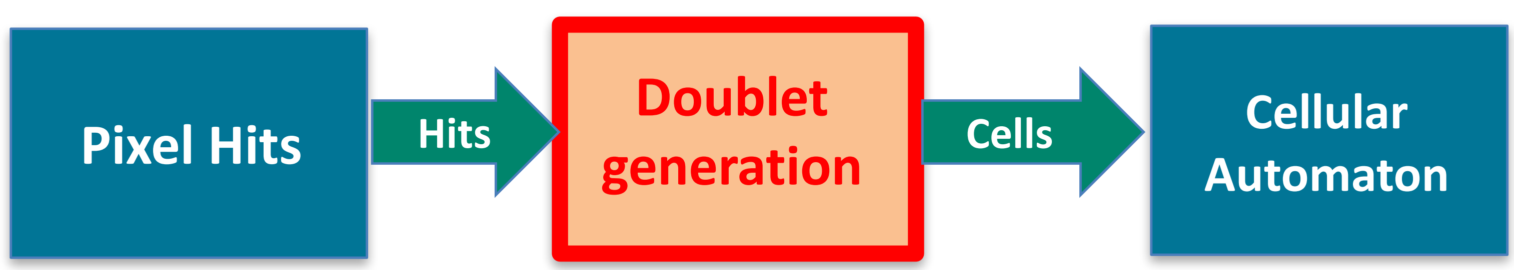
TASK: PIXEL TRACKING

- Early stage of tracking: generation of **pixel hit doublets** (seeds for tracks)





Seeding



- Doublet generation: bottleneck due to combinatorial background!
- $O(10^5)$ fake doublets produced with $O(10^3)$ true doublets

PIXEL TRACKING DATASET

opendata CERN

Search

About

Sample with tracker hit information for tracking algorithm ML studies TTbar_13TeV_PU50_PixelSeeds

Di Florio, Adriano; Pantaleo, Felice; Pierini, Maurizio;

Dataset | Derived | Datascience | CMS | CERN-LHC | Parent Dataset: TTToHadronic_TuneCP5_13TeV-powheg-pythia8 in FEVTDEBUGHLT format for LHC Phase2 studies

Description

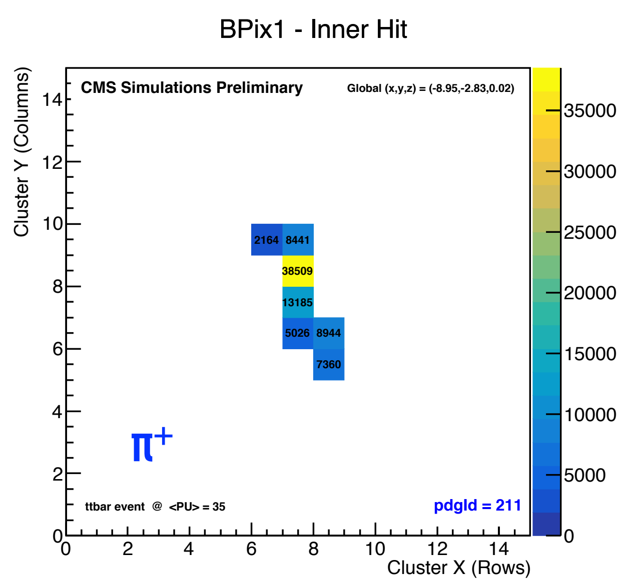
The dataset consists of a collection of pixel doublet seeds, i.e. the hit pairs that could belong to the same particle. The compatibility between two hits is evaluated only on the basis of geometrical considerations, such as cuts in η, ϕ and r. These doublets define the building blocks for further tracks. Each doublet is charcaterised by a set of features, such as its coordinates and the charge released in ther Pixel detector, and the pixel cluster shape, projected on 2D histogram.

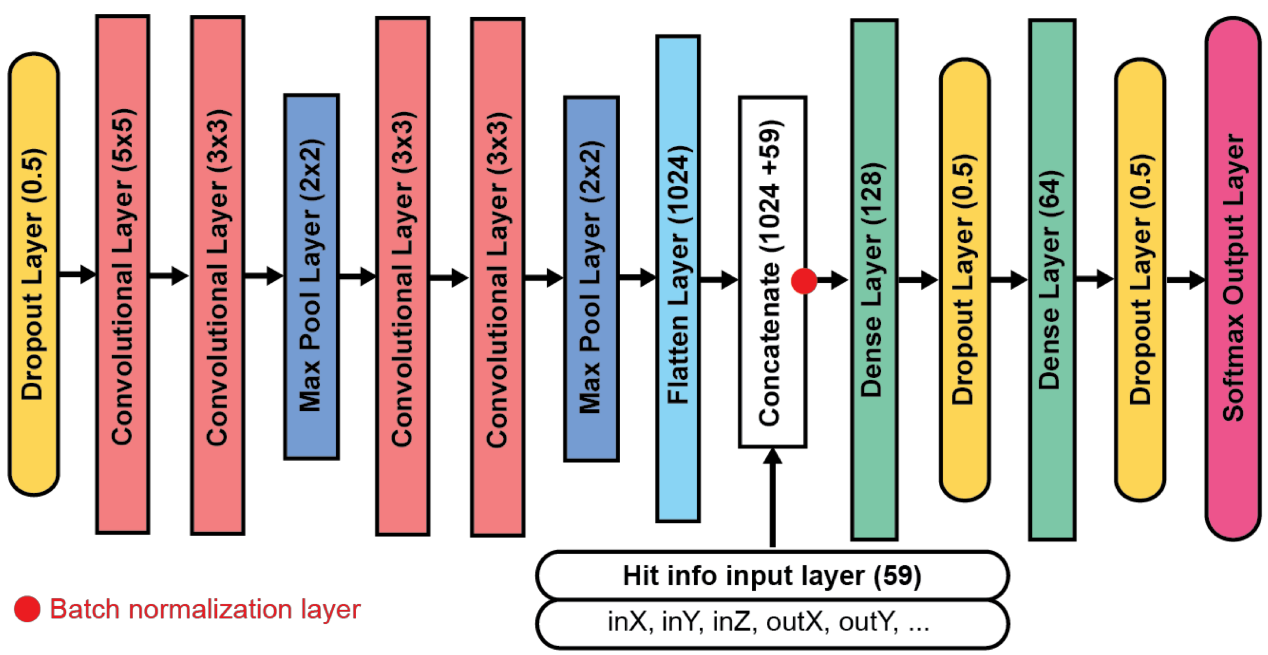
These data can be used in one of the first steps of the track finding workflow, which is the creation of track seeds, i.e. compatible pairs of hits from different detector layers, that are subsequently fed to to higher level pattern recognition steps. However the set of compatible hit pairs is highly affected by combinatorial background resulting in the next steps of the tracking algorithm to process a significant fraction of fake doublets.

- Derived dataset (HDF5): http://opendata-dev.web.cern.ch/record/12320
- 3547 files, 200 GB, 650 million total entries (doublet pixel hits)
    - Doublet features, e.g. coordinates, charge released in pixel detector, and the pixel cluster shape (2D histogram)

EXAMPLE: CNN FOR DOUBLET GENERATION

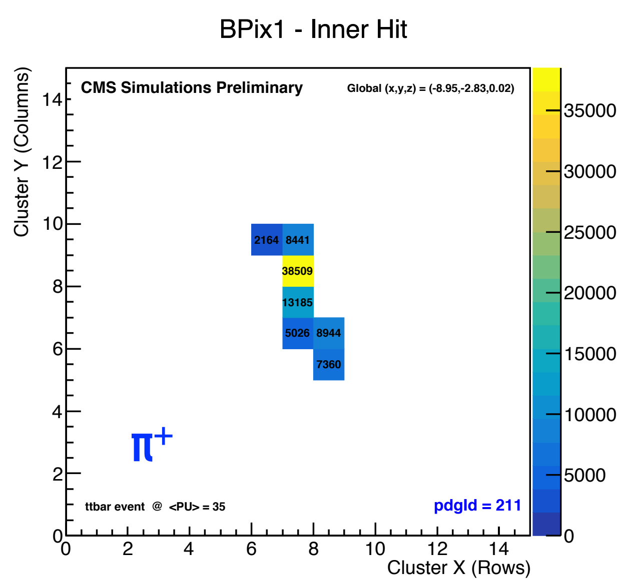
- Convolutional neural networks can be used identify good doublets

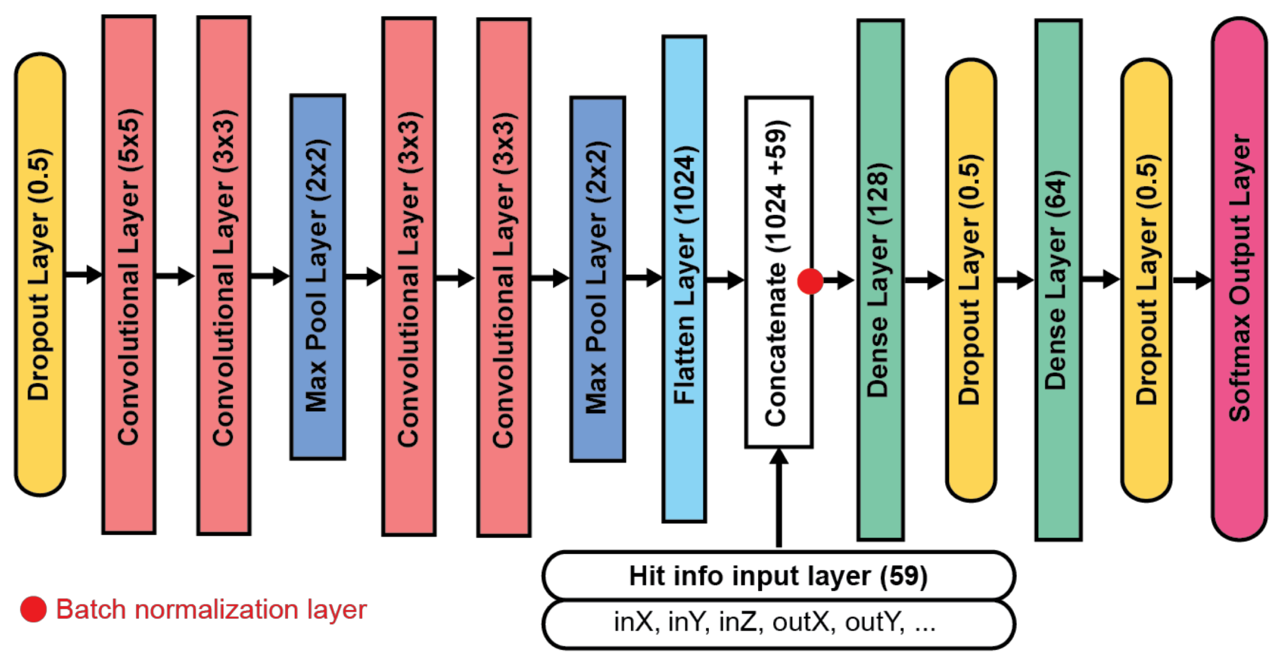


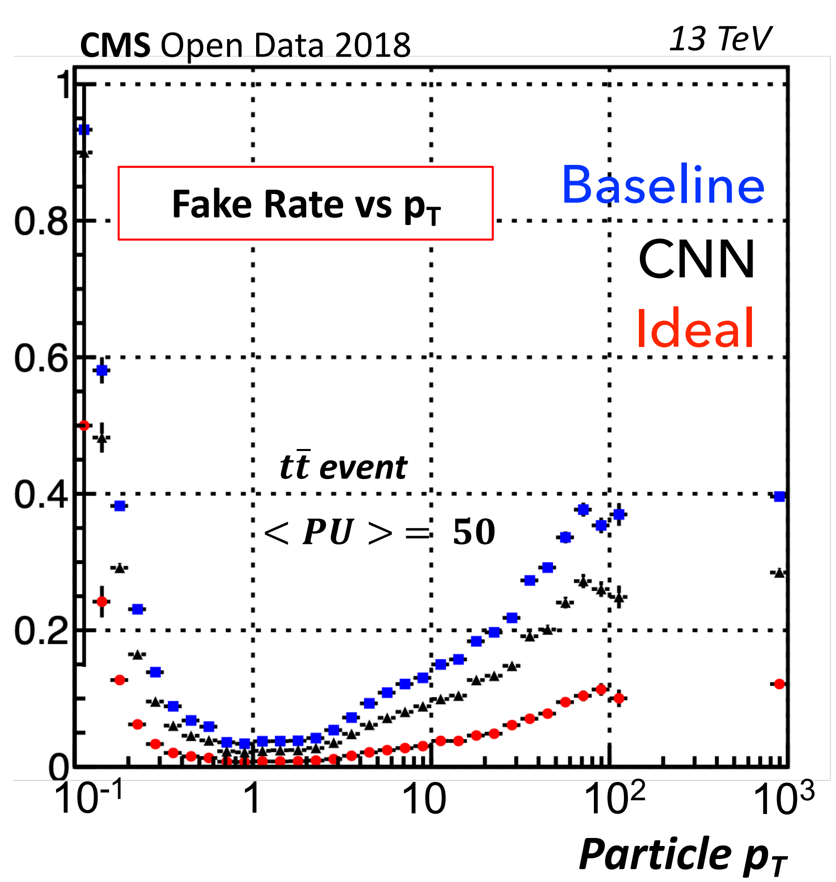


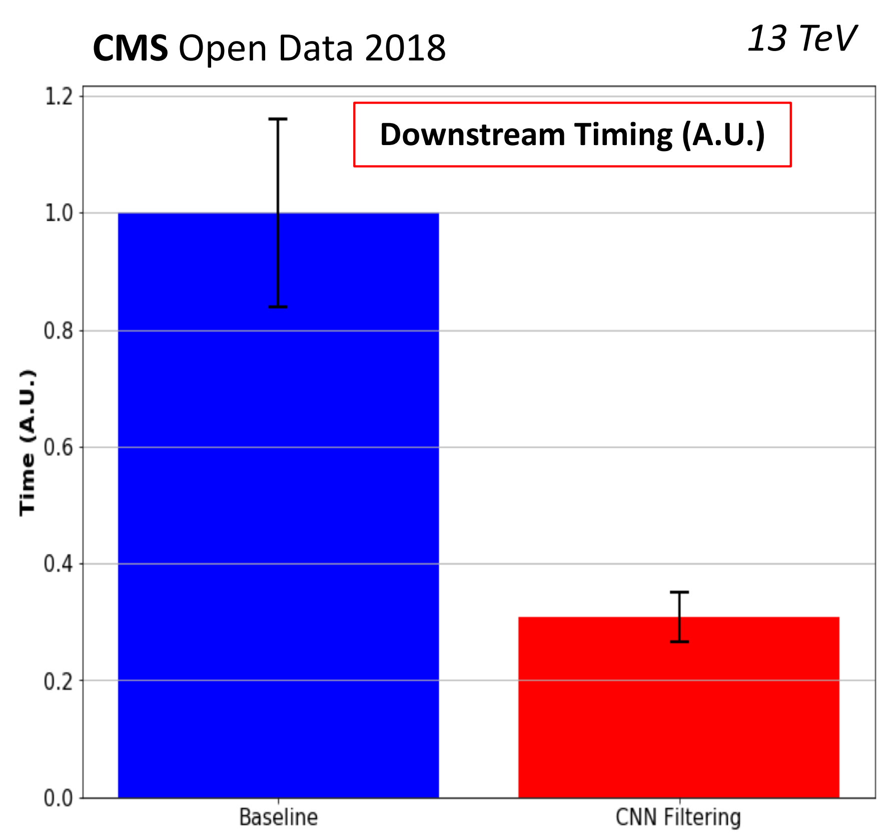
EXAMPLE: CNN FOR DOUBLET GENERATION

- Convolutional neural networks can be used identify good doublets
- Better fake rate and less CPU time





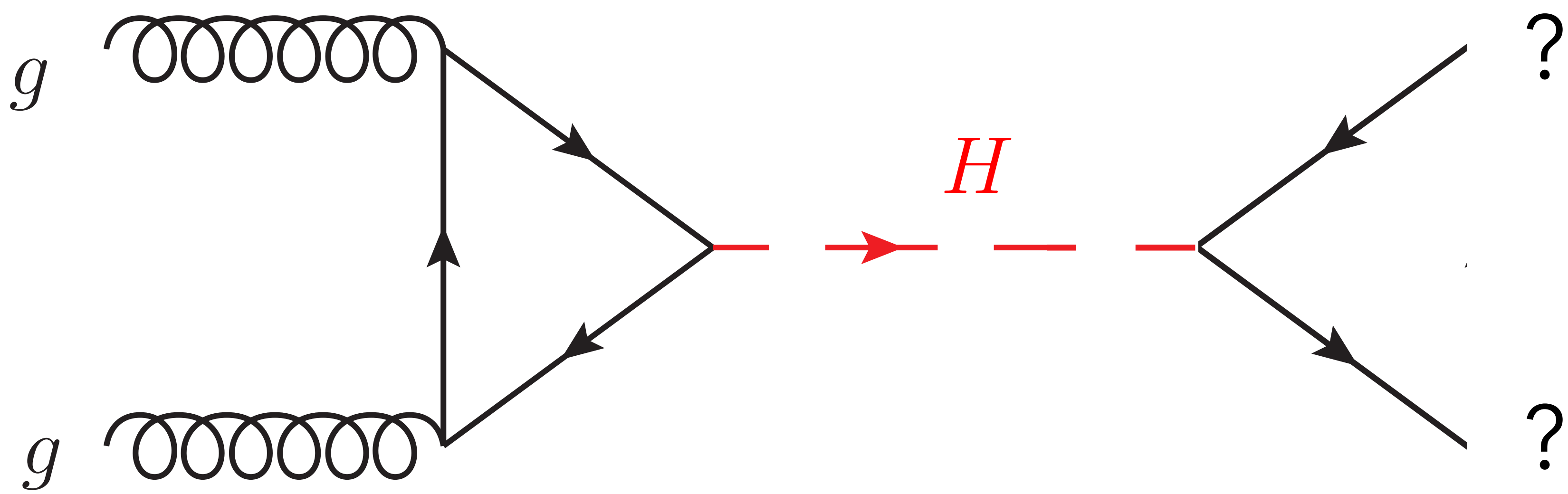




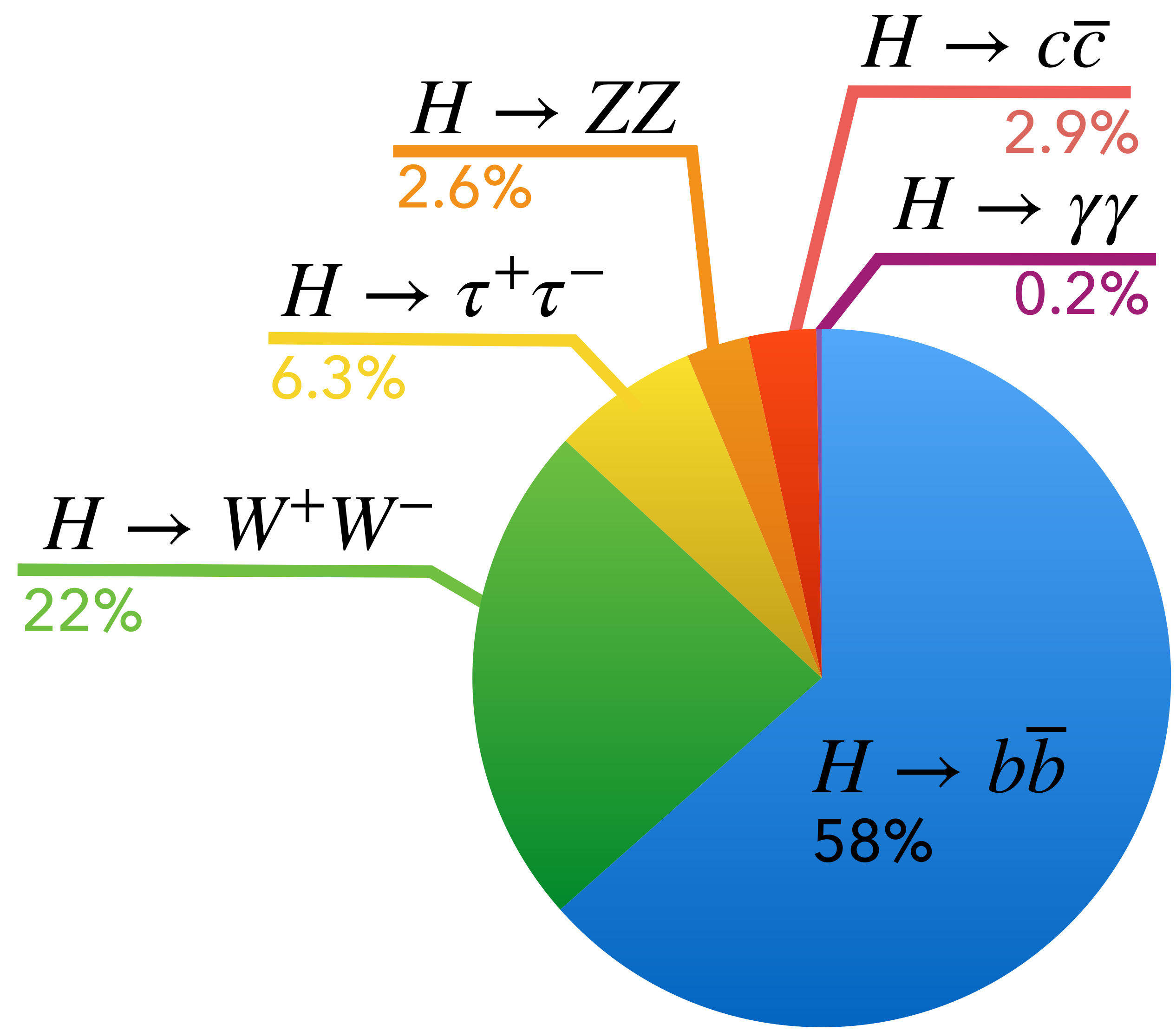
TASK: TAGGING H(BB) JETS

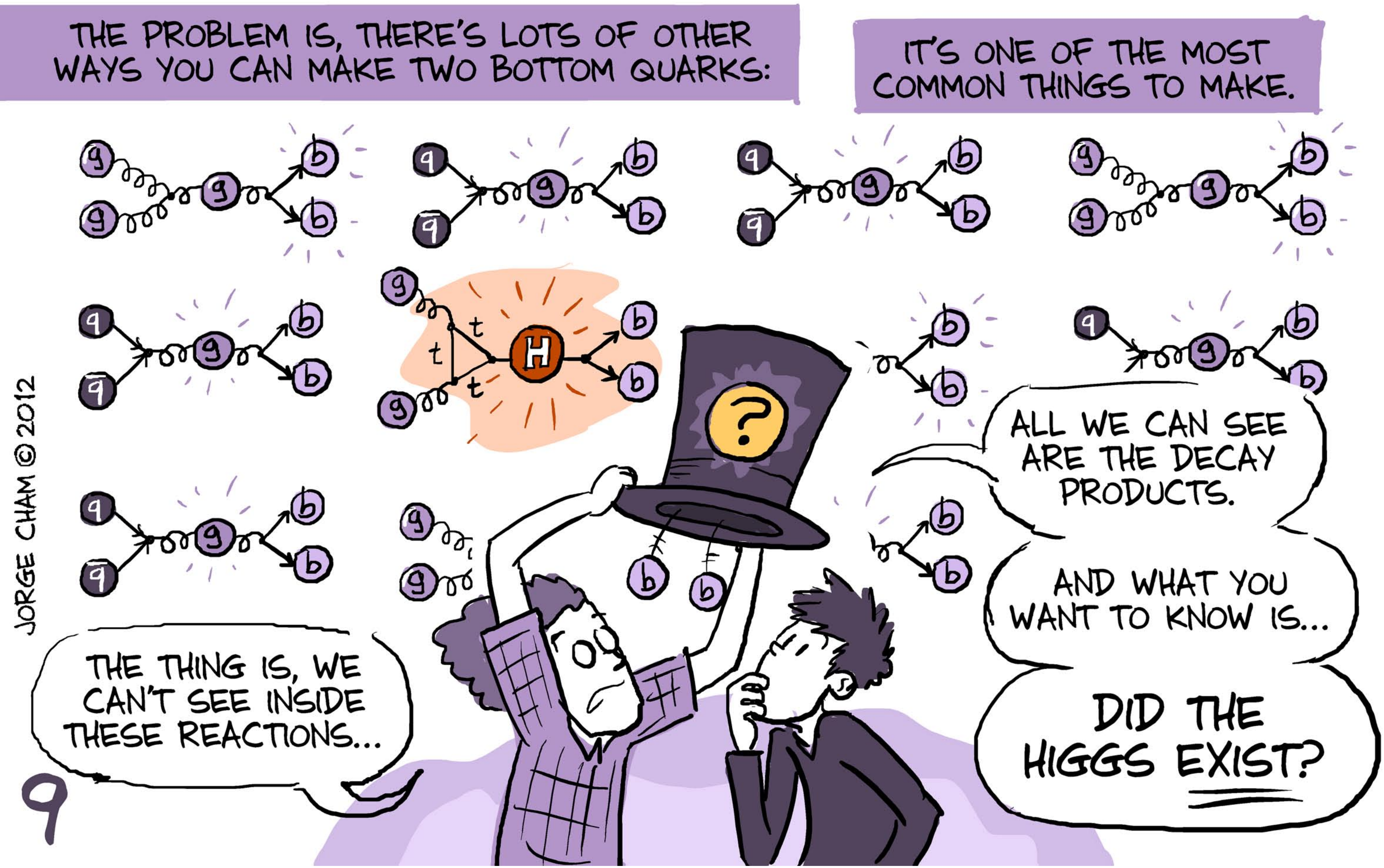
produce detect

(87%)



- Biggest Higgs decay mode is H→bb
- ggF high-p_T Higgs production may be sensitive to new physics
- Large QCD background…



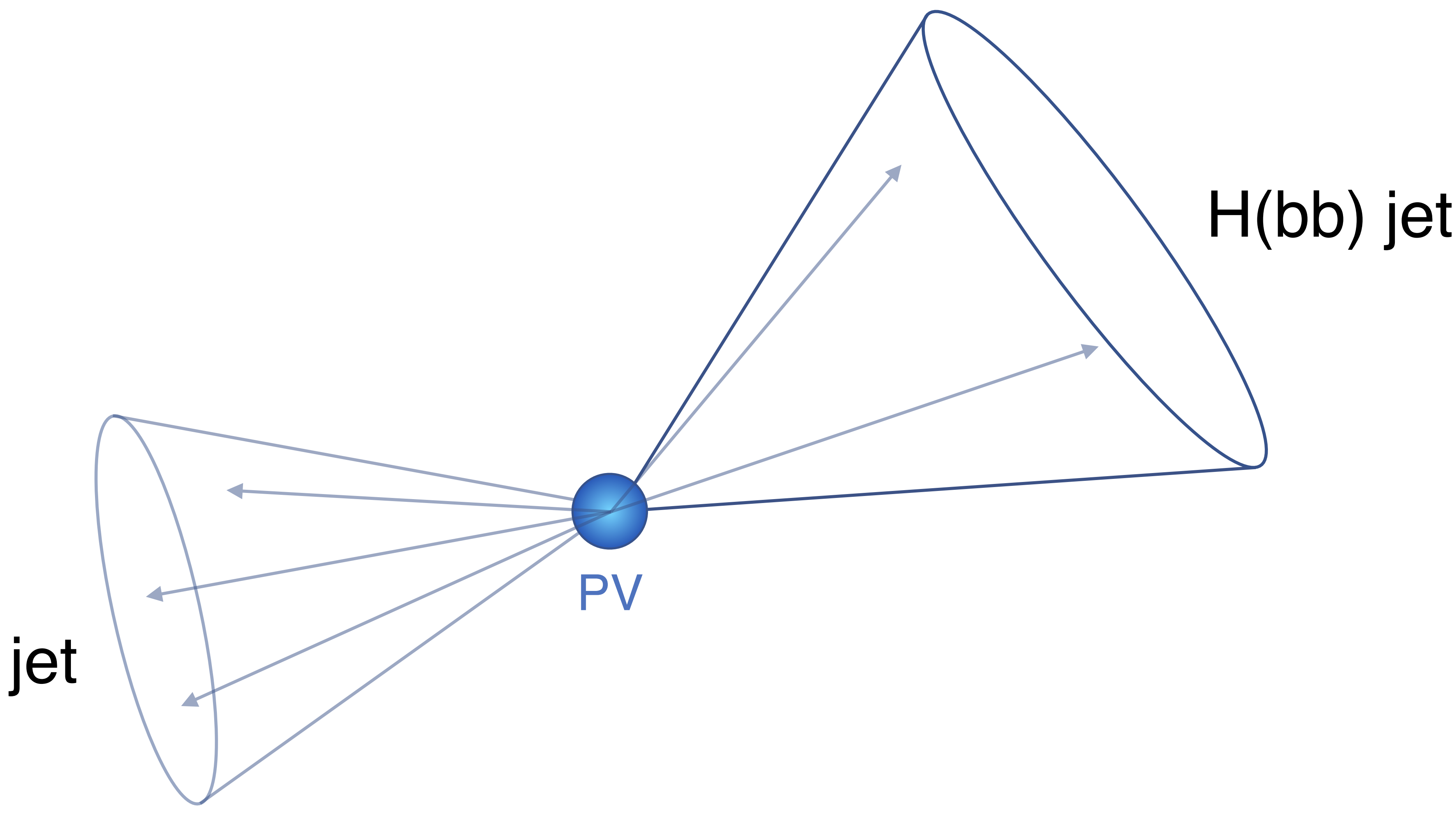


THE PROBLEM IS, THERE'S LOTS OF OTHER WAYS YOU CAN MAKE TWO BOTTOM QUARKS:
IT'S ONE OF THE MOST COMMON THINGS TO MAKE.
THE THING IS, WE CAN'T SEE INSIDE THESE REACTIONS...
ALL WE CAN SEE ARE THE DECAY PRODUCTS.
AND WHAT YOU WANT TO KNOW IS...
DID THE HIGGS EXIST?
JORGE CHAM © 2012
9

TASK: TAGGING H(BB) JETS

b hadrons have long lifetimes:
travel O(mm) before decay!

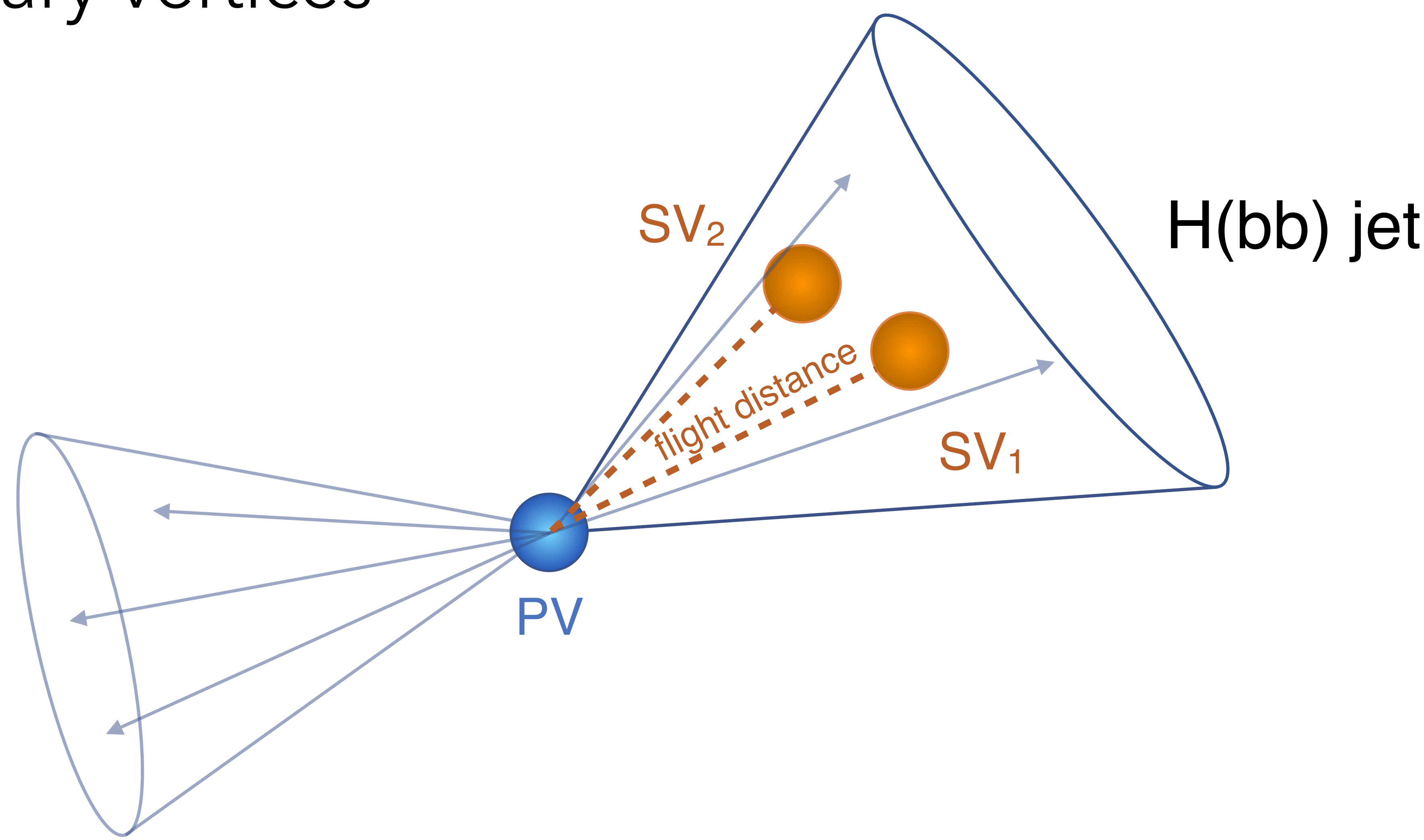
- Handles:



TASK: TAGGING H(BB) JETS

b hadrons have long lifetimes:
travel O(mm) before decay!

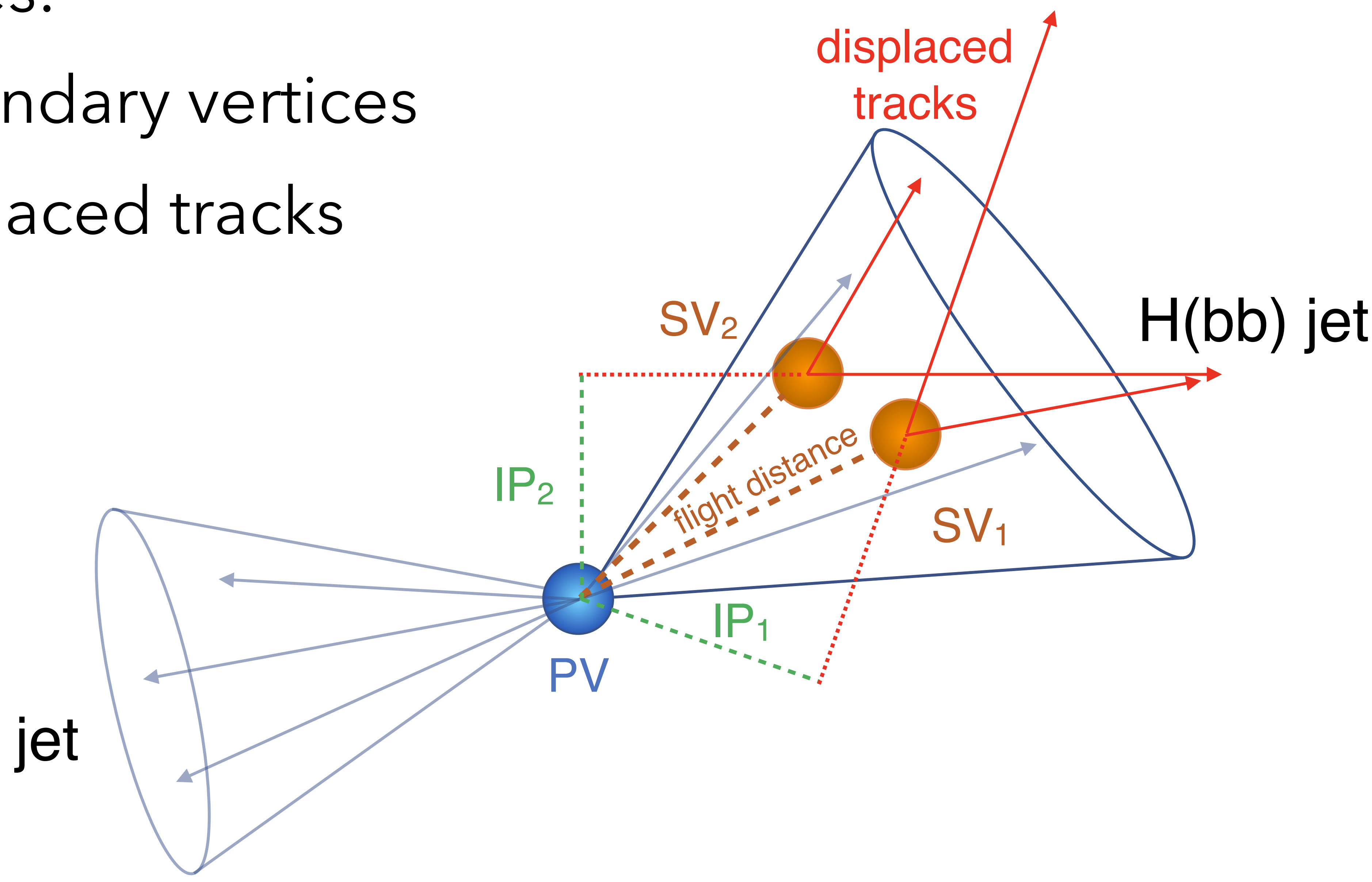
- Handles:
  - secondary vertices



TASK: TAGGING H(BB) JETS

b hadrons have long lifetimes:
travel O(mm) before decay!

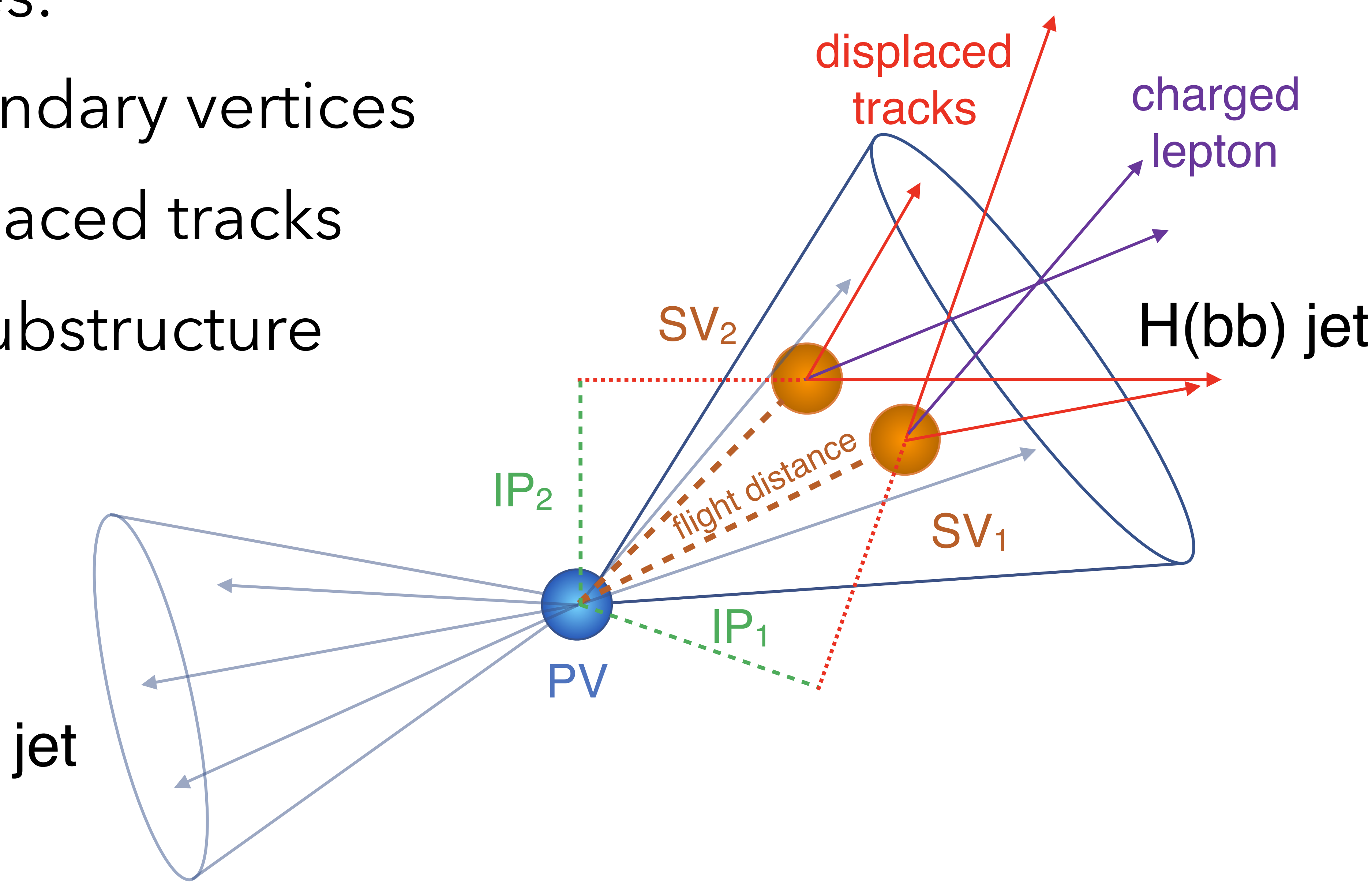
- Handles:
  - secondary vertices
  - displaced tracks



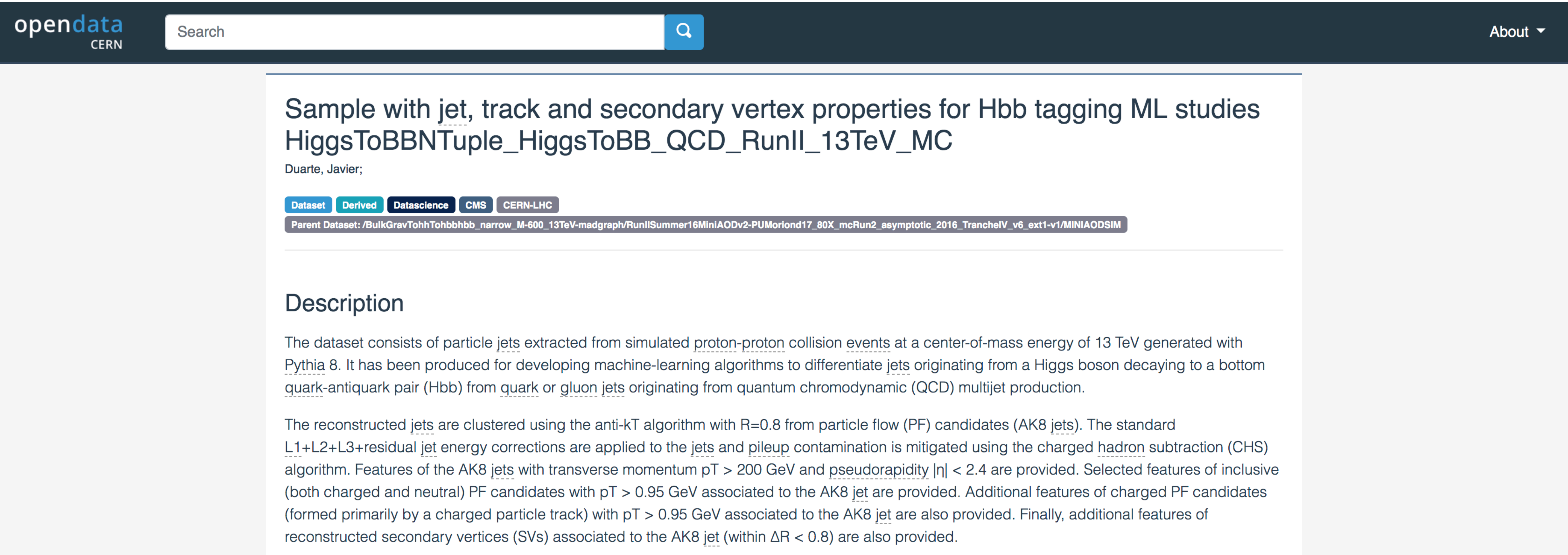
TASK: TAGGING H(BB) JETS

b hadrons have long lifetimes:
travel O(mm) before decay!

- Handles:
  - secondary vertices
  - displaced tracks
  - jet substructure



H(BB) DATASET



opendata CERN

Search

About

Sample with jet, track and secondary vertex properties for Hbb tagging ML studies HiggsToBBNTuple_HiggsToBB_QCD_RunII_13TeV_MC

Duarte, Javier;

Dataset Derived Datascience CMS CERN-LHC

Parent Dataset: /BulkGravTohhTohbbhbb_narrow_M-600_13TeV-madgraph/RunIISummer16MiniAODv2-PUMoriond17_80X_mcRun2_asymptotic_2016_TrancheIV_v6_ext1-v1/MINIAODSIM

Description

The dataset consists of particle jets extracted from simulated proton-proton collision events at a center-of-mass energy of 13 TeV generated with Pythia 8. It has been produced for developing machine-learning algorithms to differentiate jets originating from a Higgs boson decaying to a bottom quark-antiquark pair (Hbb) from quark or gluon jets originating from quantum chromodynamic (QCD) multijet production.

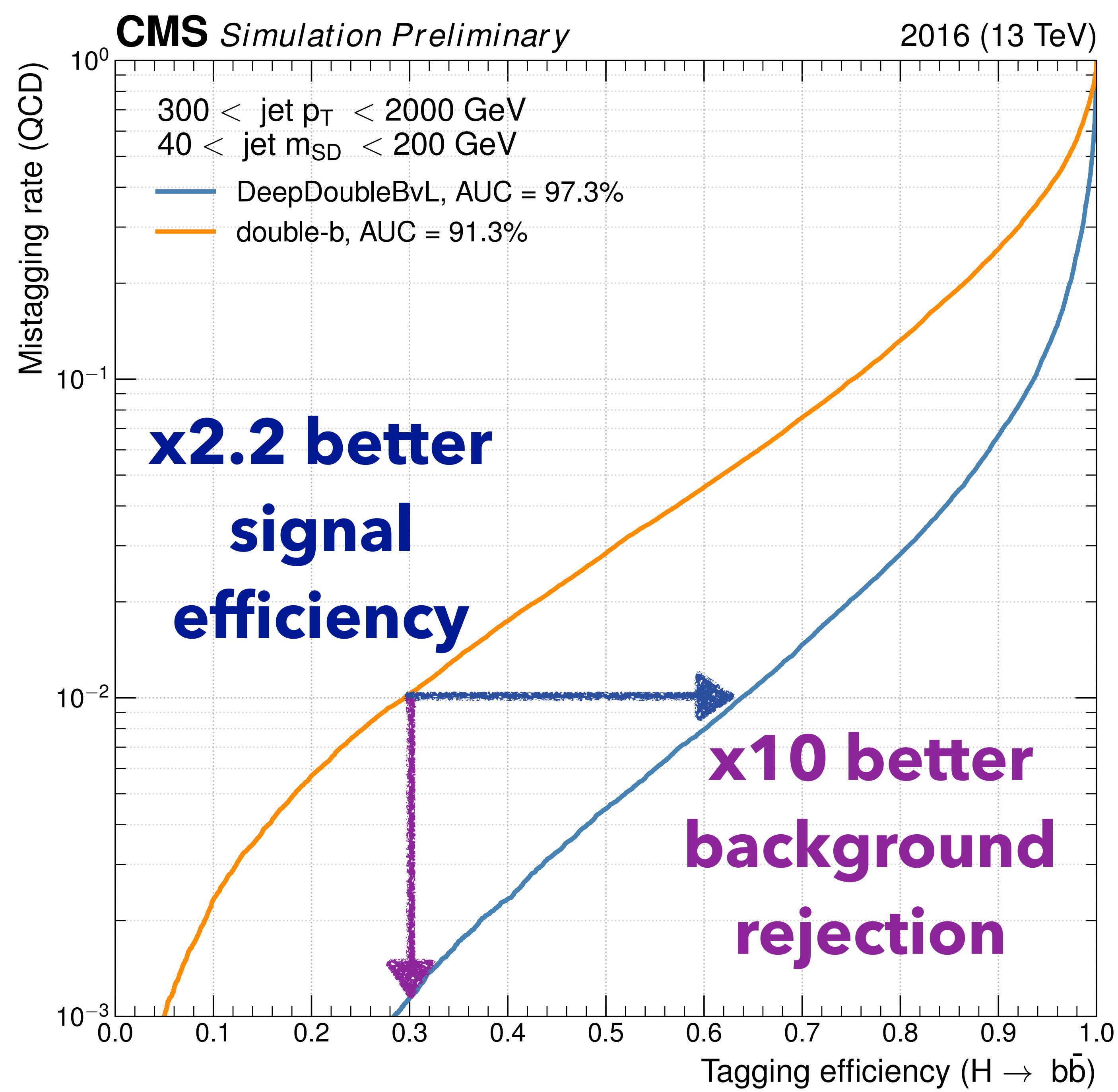
The reconstructed jets are clustered using the anti-kT algorithm with R=0.8 from particle flow (PF) candidates (AK8 jets). The standard L1+L2+L3+residual jet energy corrections are applied to the jets and pileup contamination is mitigated using the charged hadron subtraction (CHS) algorithm. Features of the AK8 jets with transverse momentum pT > 200 GeV and pseudorapidity |η| < 2.4 are provided. Selected features of inclusive (both charged and neutral) PF candidates with pT > 0.95 GeV associated to the AK8 jet are provided. Additional features of charged PF candidates (formed primarily by a charged particle track) with pT > 0.95 GeV associated to the AK8 jet are also provided. Finally, additional features of reconstructed secondary vertices (SVs) associated to the AK8 jet (within ΔR < 0.8) are also provided.

- Derived datasets (ROOT & HDF5): http://opendata-dev.web.cern.ch/record/12102
- 182 files, 245 GB, 18 million total entries (jets)
  - event features, e.g. MET, ρ (average density)
  - jet features, e.g. mass, p_T, N-subjettiness variables
  - particle candidate features, e.g. p_T, η, **ϕ** (*for up to 100 particles*)
  - charged particle / track features, e.g. impact parameter (*for up to 60 tracks*)
  - secondary vertex features, e.g. flight distance (*for up to 5 vertices*)

CMS DEEP DOUBLE-B TAGGING

[JINST 13 (2018) P05011](#)
[DP-2018/033](#)

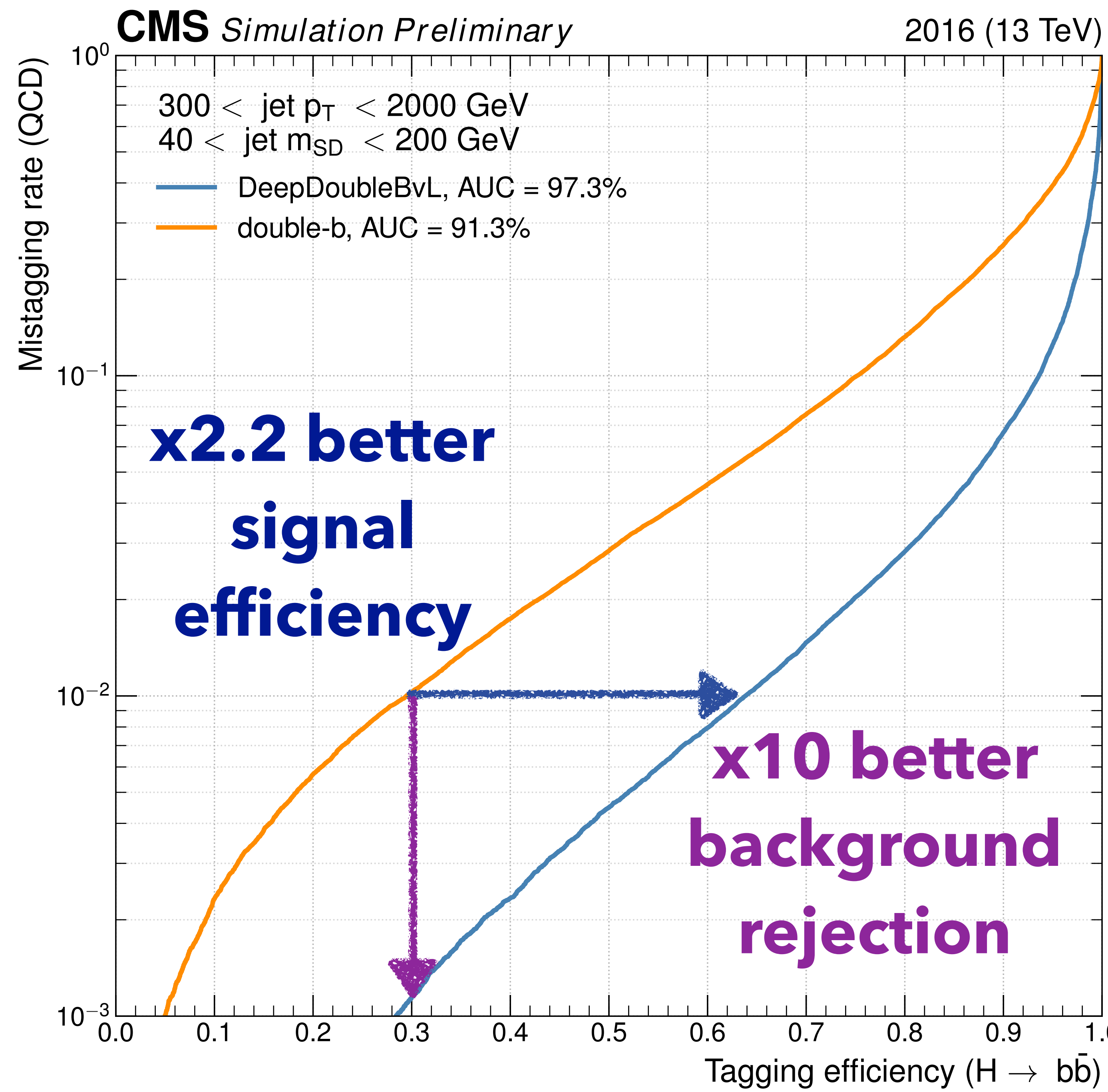
- Deeper neural network with track and secondary vertex inputs in convolutional (1D CNN) and recurrent (GRU) network
  - Large performance gain over previous algorithm (BDT with high-level features)



CMS DEEP DOUBLE-B TAGGING

JINST 13 (2018) P05011
[DP-2018/033](DP-2018/033)

- Deeper neural network with track and secondary vertex inputs in convolutional (1D CNN) and recurrent (GRU) network
  - Large performance gain over previous algorithm (BDT with high-level features)
- New approaches?
  - Deep 2D CNNs (ResNet-50), interaction/graph neural networks, …

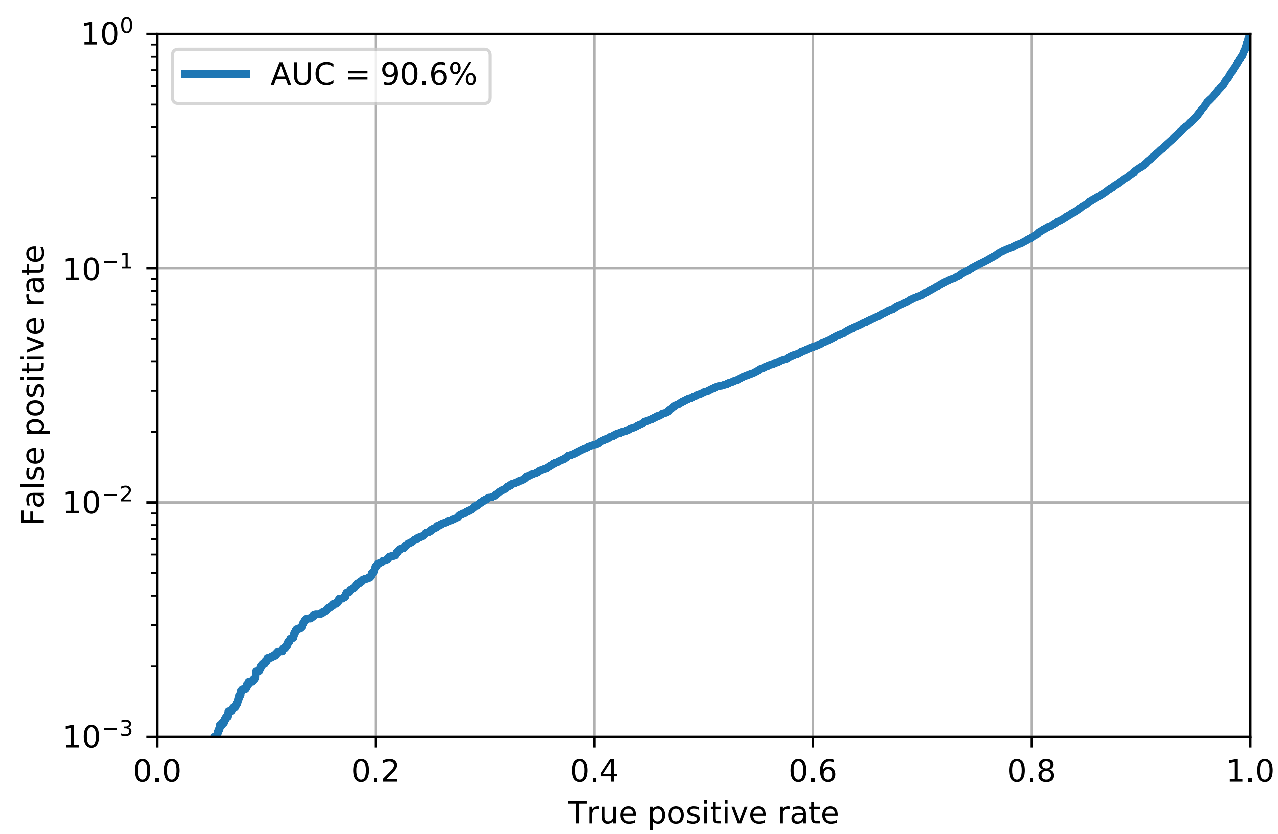


EXAMPLE: SIMPLE NEURAL NETWORK TRAINING

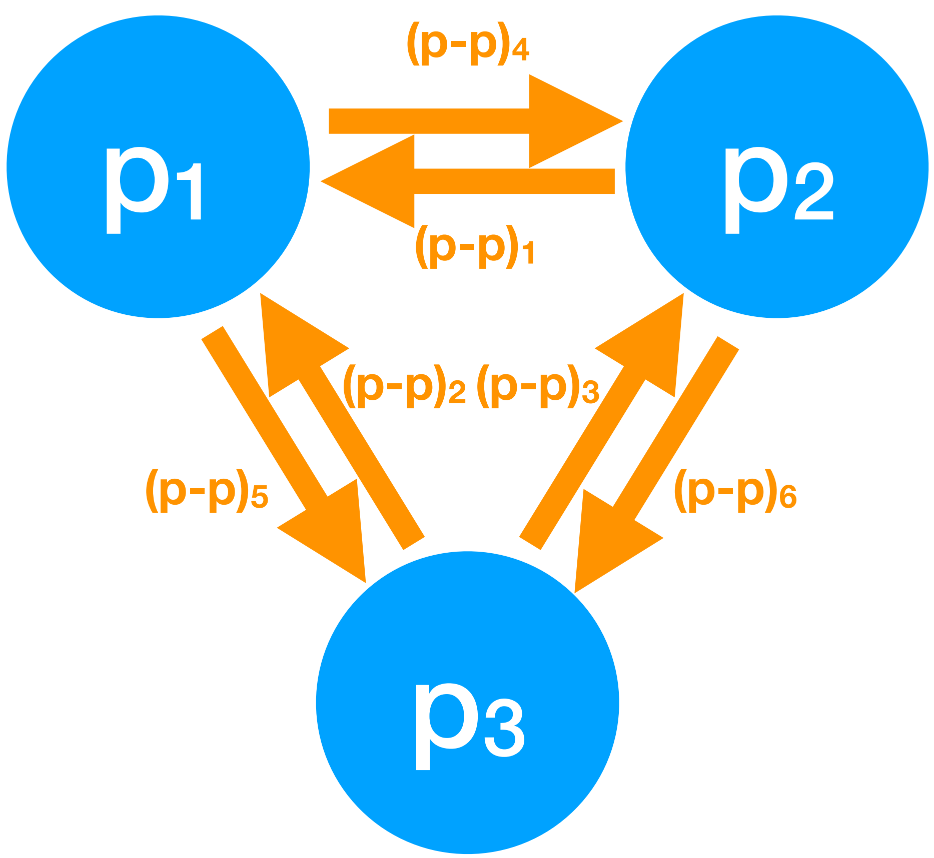
https://github.com/cernopendata-datascience/HiggsToBBMachineLearning

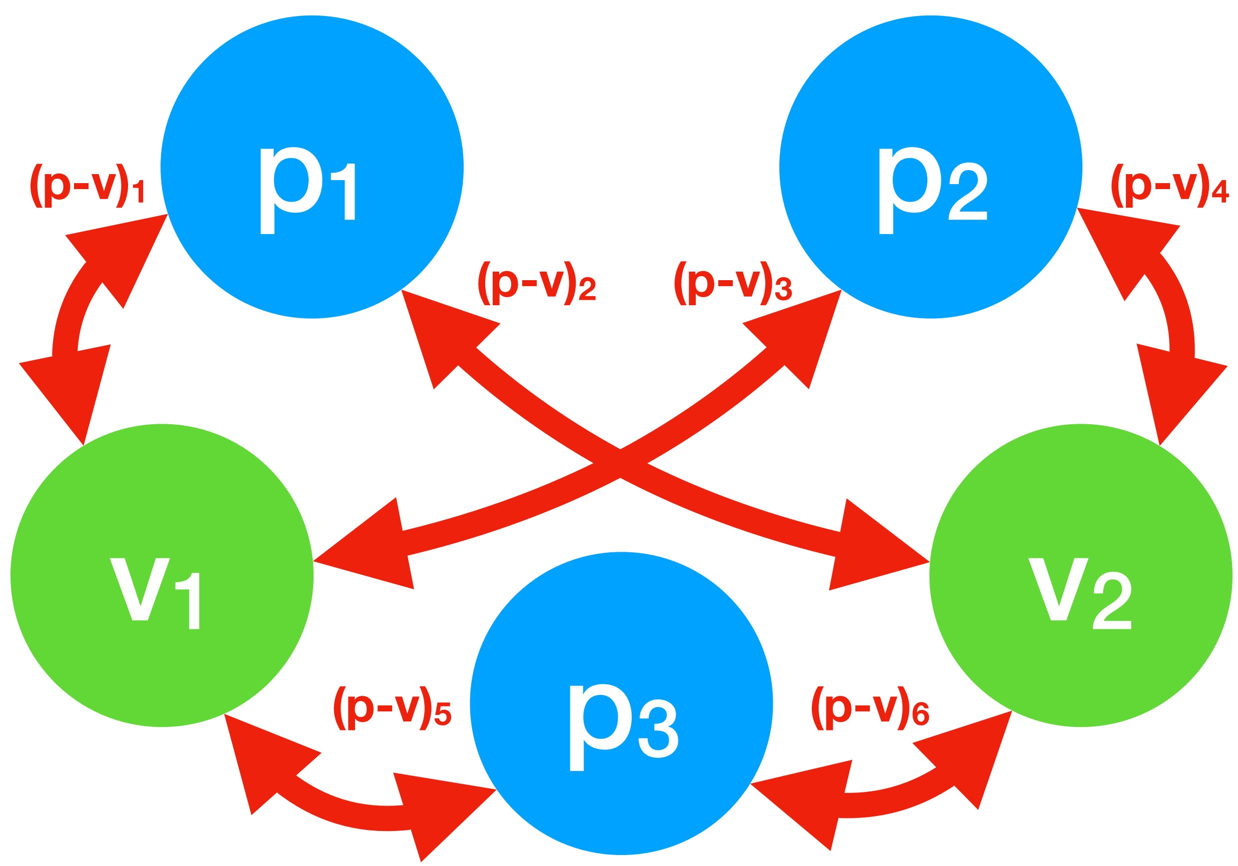
- Train fully connected neural network with high level features in ~30 lines of code
- Similar performance to CMS double-b tagger (BDT) with 1 training file

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input (InputLayer)           (None, 27)                0
_________________________________________________________________
bn_1 (BatchNormalization)    (None, 27)                108
_________________________________________________________________
dense_1 (Dense)              (None, 64)                1792
_________________________________________________________________
dense_2 (Dense)              (None, 32)                2080
_________________________________________________________________
dense_3 (Dense)              (None, 32)                1056
_________________________________________________________________
output (Dense)               (None, 2)                 66
=================================================================
Total params: 5,102
Trainable params: 5,048
Non-trainable params: 54
_________________________________________________________________
```



- Architectures like **interaction networks** can learn representations of **particle-particle** and **particle-vertex** interactions to better identify H→bb jets





SUMMARY

- Many ML efforts underway using CERN open data

SUMMARY

- Many ML efforts underway using CERN open data
- New CMS ML-focused release should make open data more accessible to ML enthusiasts / data scientists
  - Not just a set of derived files, full provenance (original files, open source ntuple code)!
  - Simpler to collaborate within and outside of HEP

A VISION OF THE FUTURE

- LHC has a looming “big data” challenge
- Collaborations with data scientists / ML academics may help to solve our algorithmic and computing challenges

JAVIER DUARTE

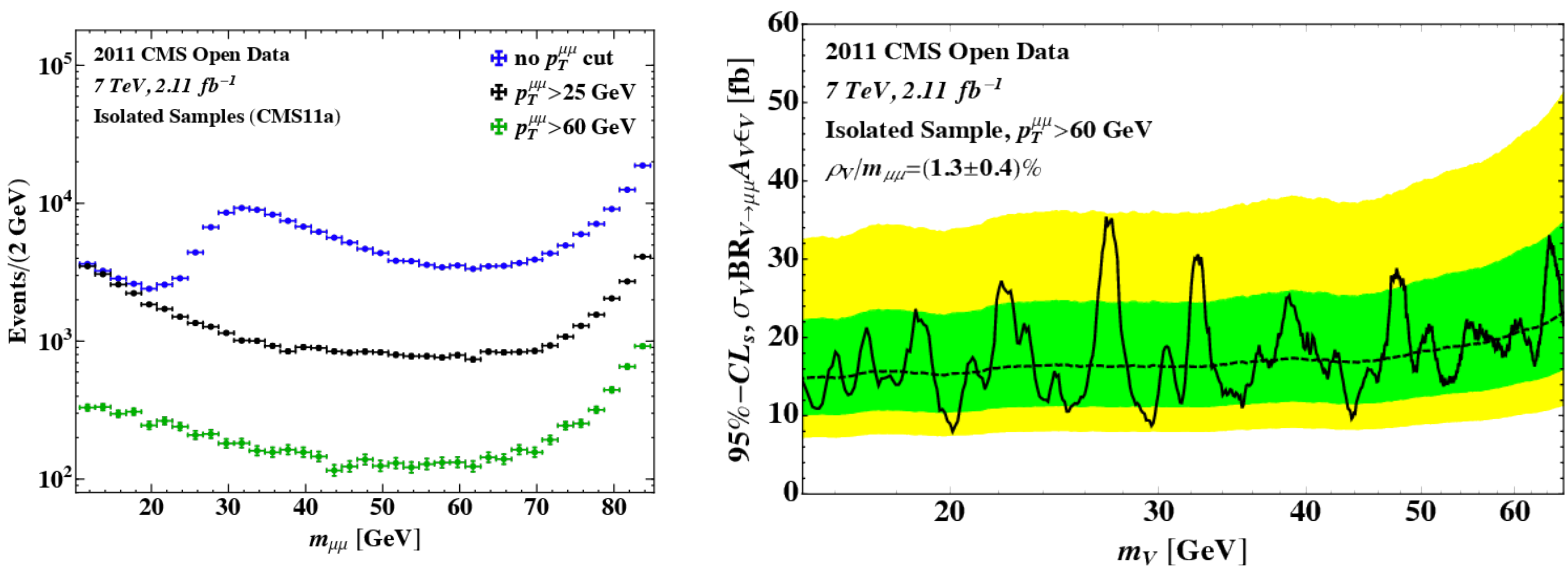
MAY 21, 2019

LHCP, PUEBLA, MEXICO

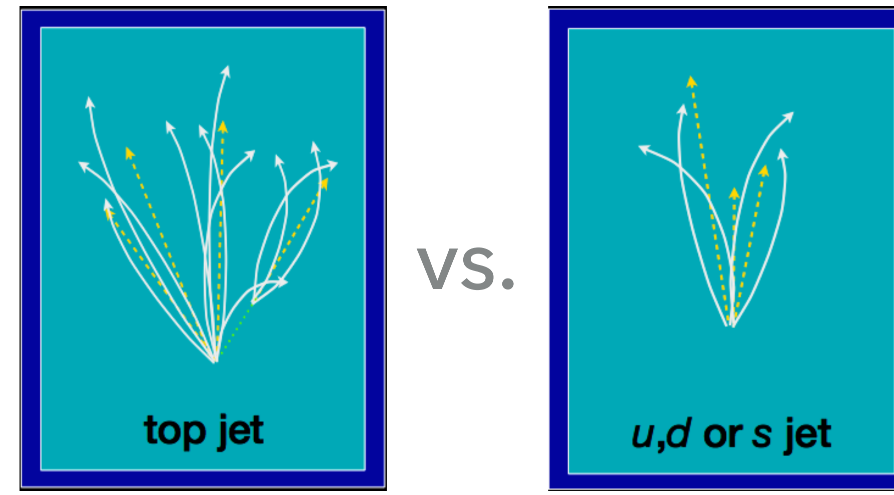
BACKUP

PHYSICS ANALYSIS USING OPEN DATA

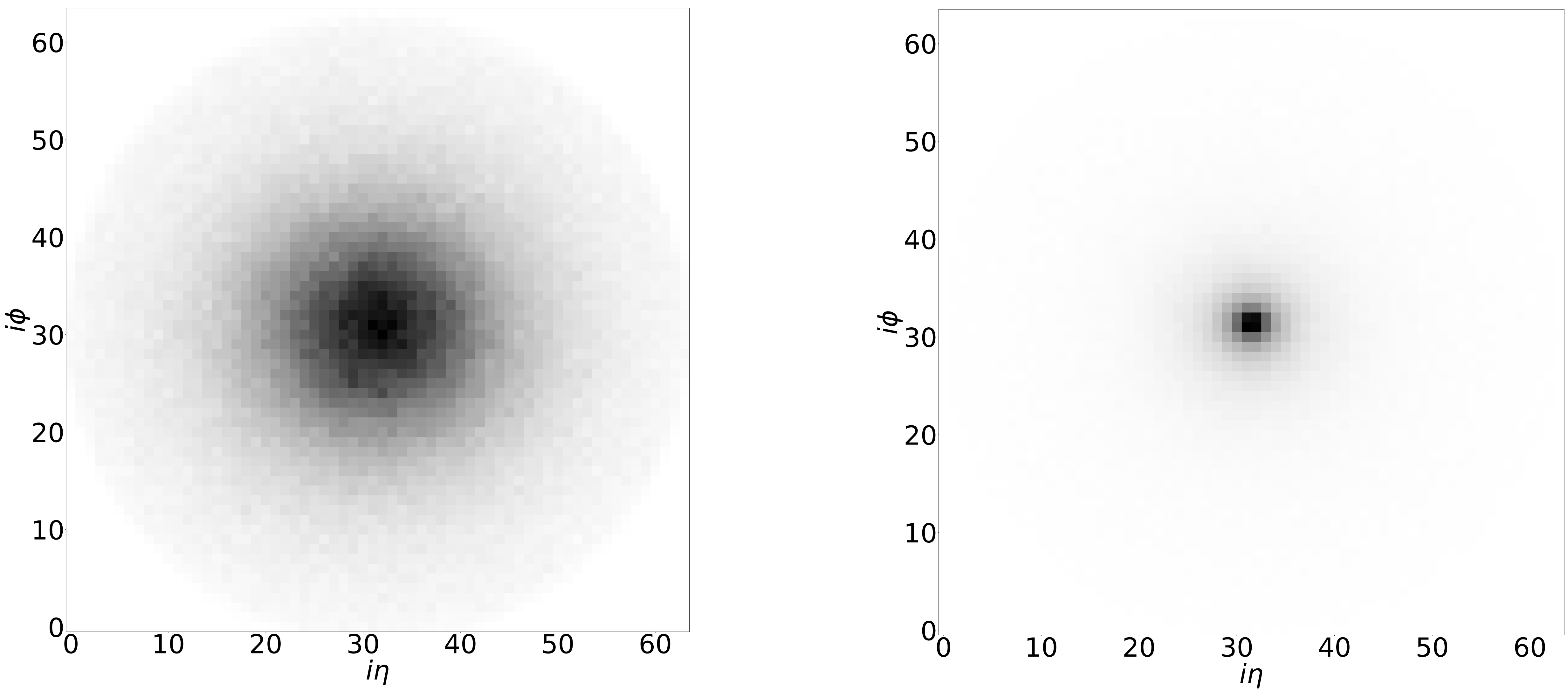
- Open data has been used to perform new searches and measurements
- Searches for non-standard sources of parity violation in jets [arXiv:1904.11195]
- Searches for high-p_T dimuon resonances [arXiv:1902.04222]



- Re-train ResNet-50 to identify the origin of jets
- Inputs are jet images = pixelated versions of calorimeter hits in 2D (η, Φ)



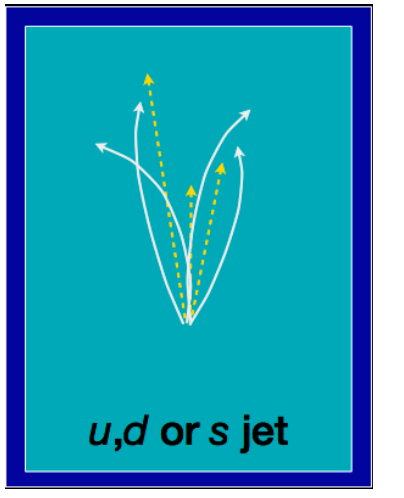
Note: averaged over 10k jets; 1 jet gives a ***sparse*** image



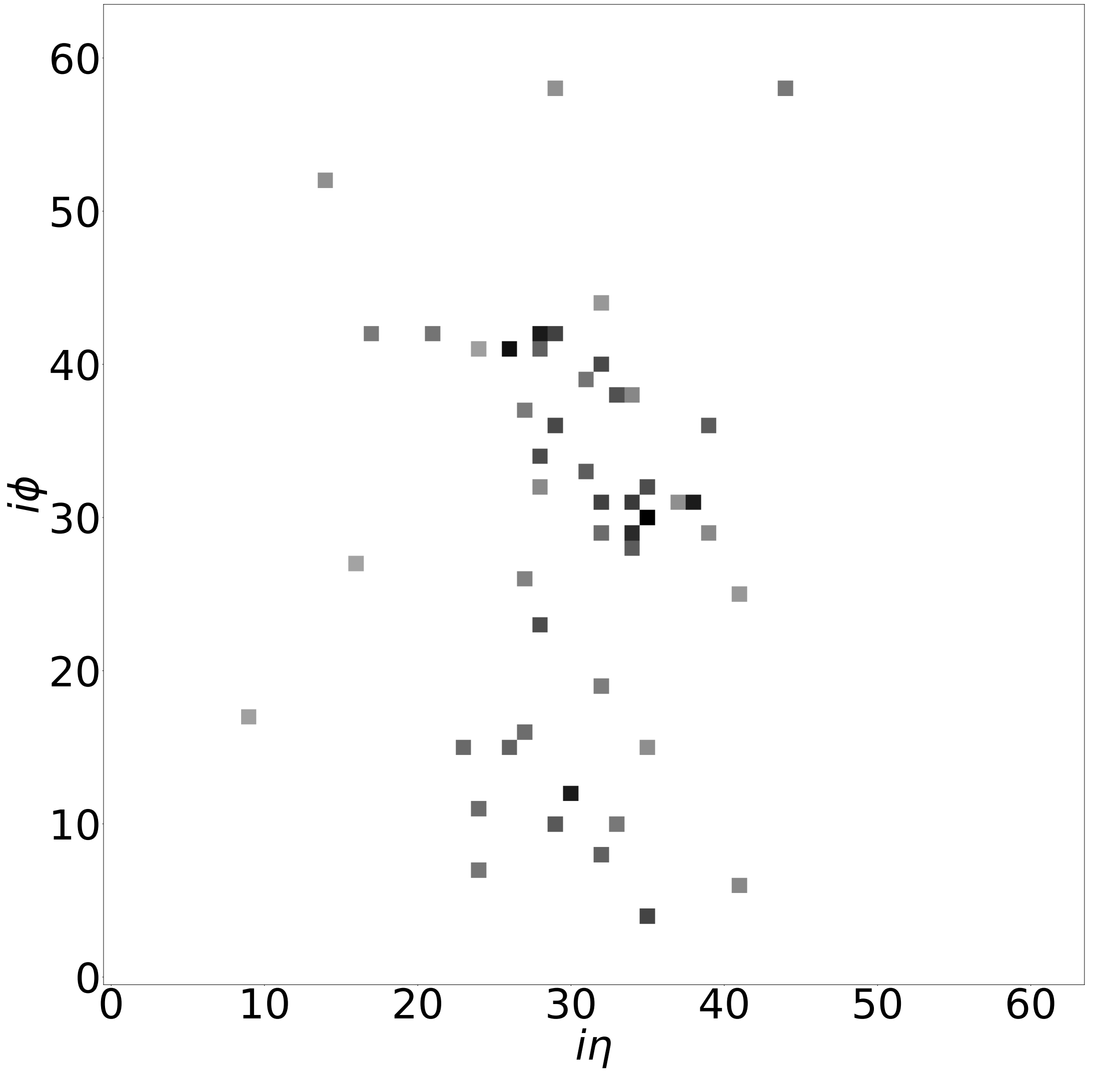
JET VISION

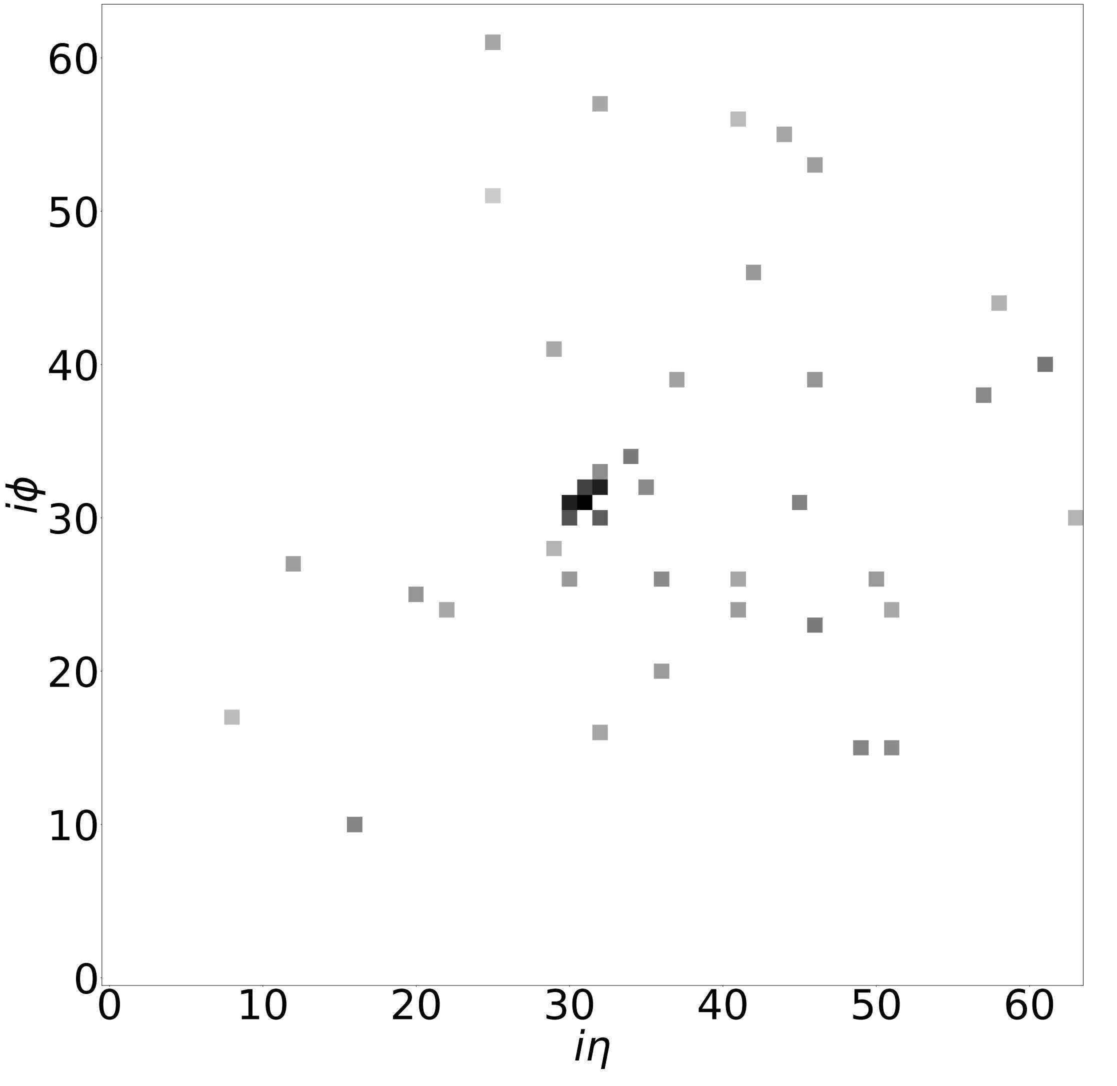
- Re-train ResNet-50 to identify the origin of jets
- Inputs are jet images = pixelated versions of calorimeter hits in 2D (η, Φ)

vs.



Note: averaged over 10k jets; 1 jet gives a ***sparse*** image





THE LARGE HADRON COLLIDER

THE LARGE HADRON COLLIDER

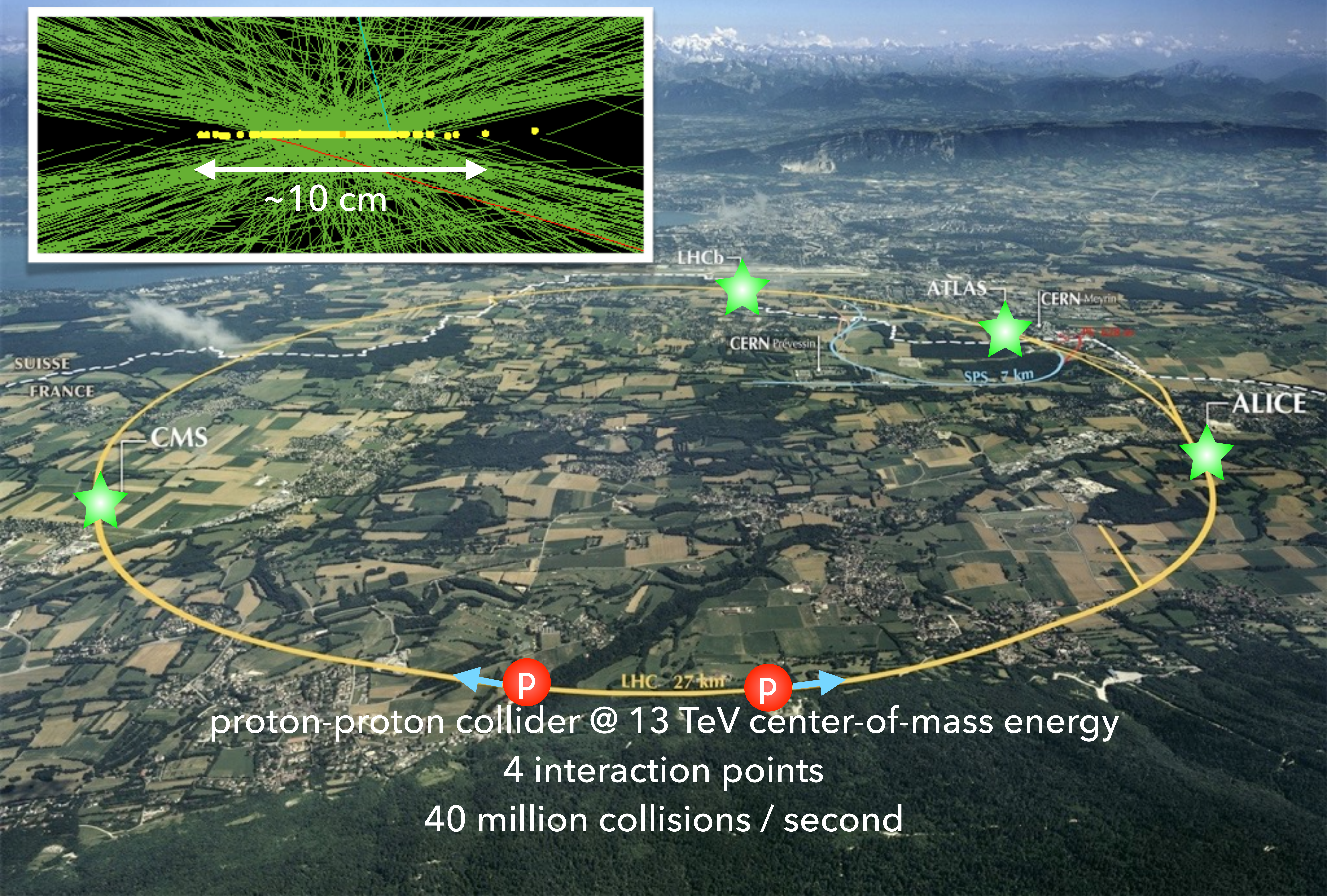
proton-proton collider @ 13 TeV center-of-mass energy

THE LARGE HADRON COLLIDER

proton-proton collider @ 13 TeV center-of-mass energy

4 interaction points

THE LARGE HADRON COLLIDER

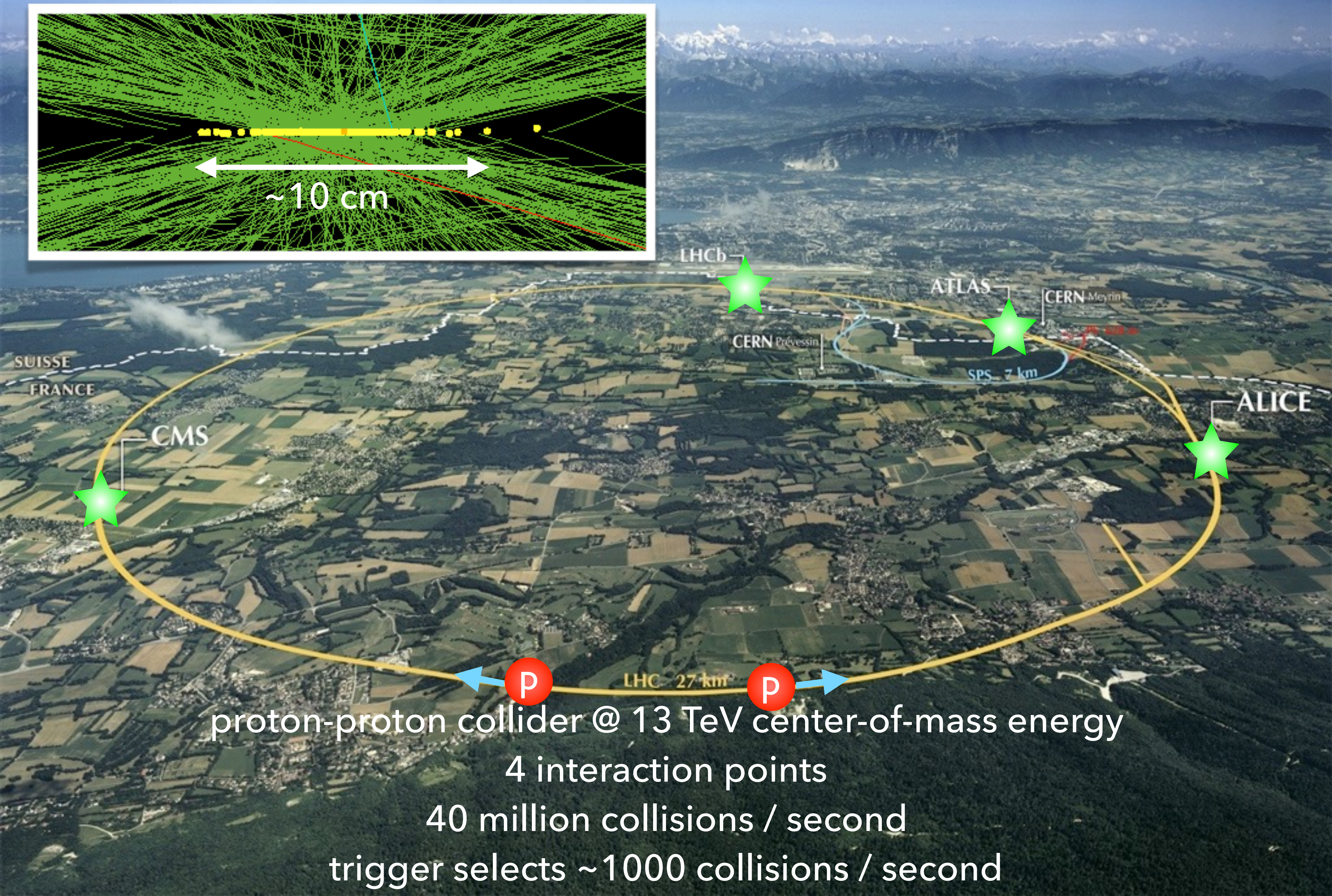


proton-proton collider @ 13 TeV center-of-mass energy

4 interaction points

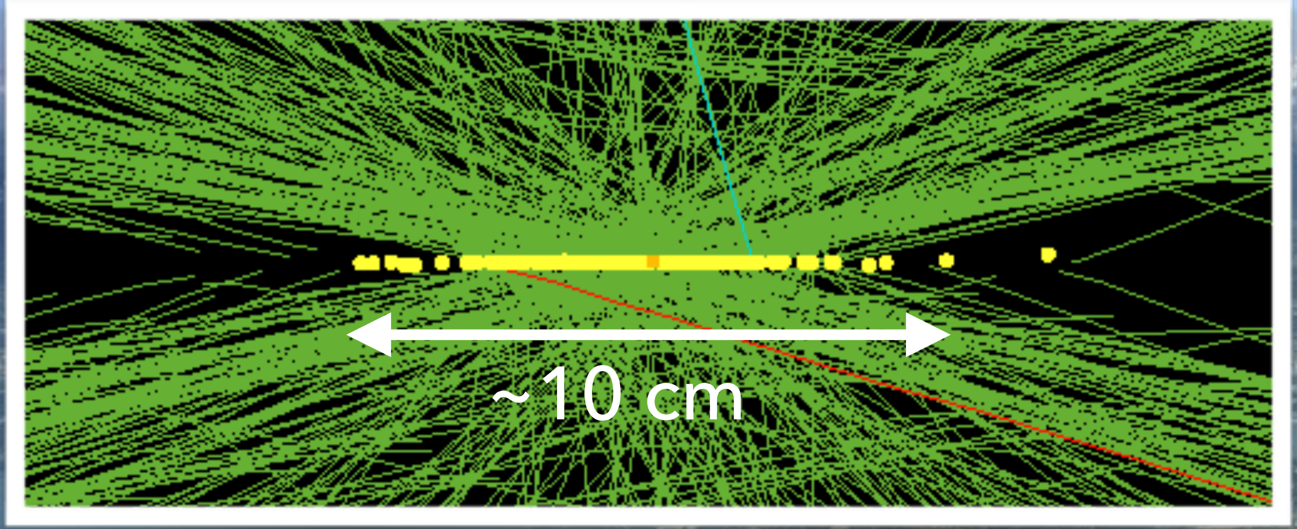
40 million collisions / second

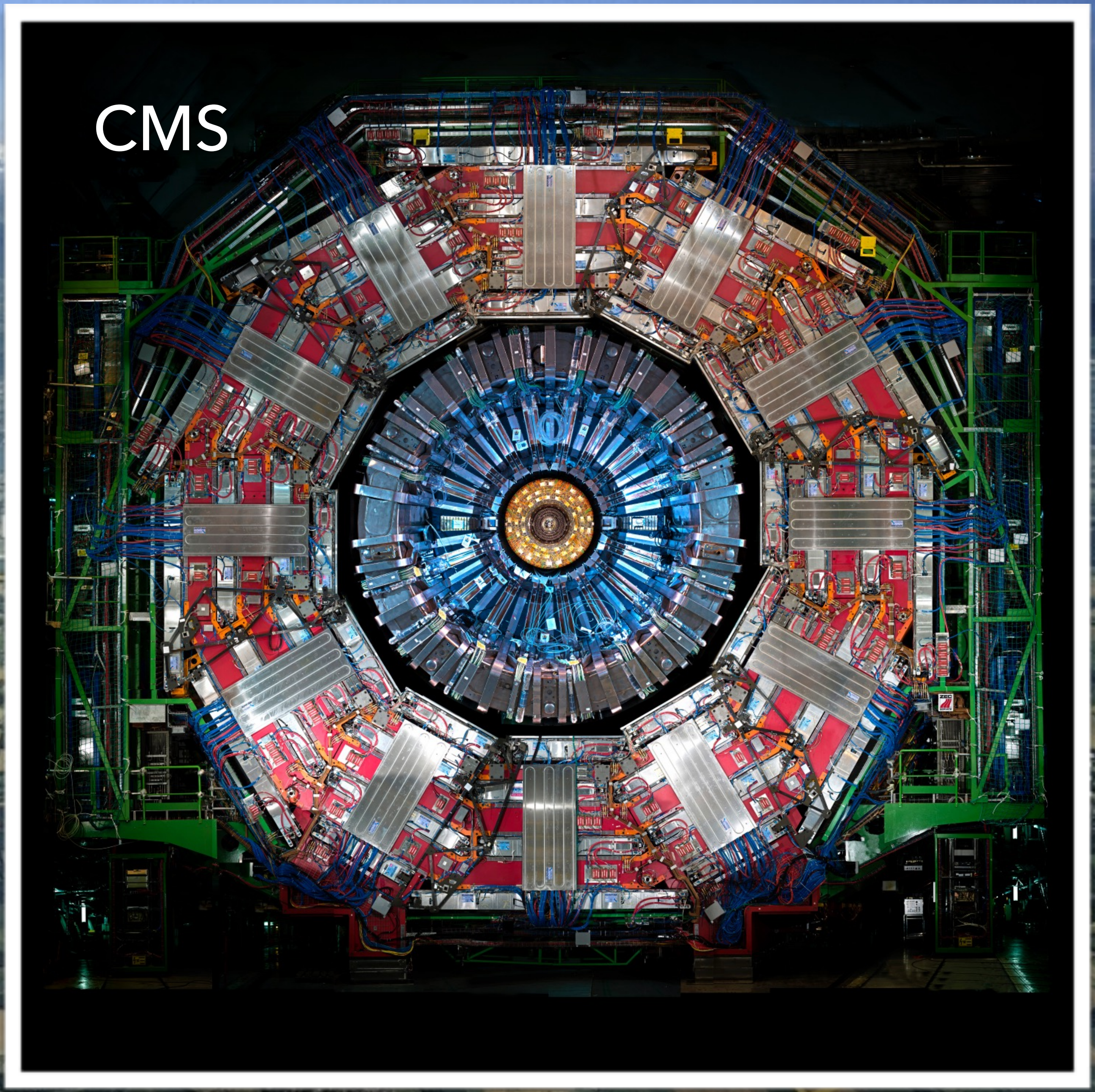
THE LARGE HADRON COLLIDER



proton-proton collider @ 13 TeV center-of-mass energy
4 interaction points
40 million collisions / second
trigger selects ~1000 collisions / second

THE LARGE HADRON COLLIDER





proton-proton collider @ 13 TeV center-of-mass energy

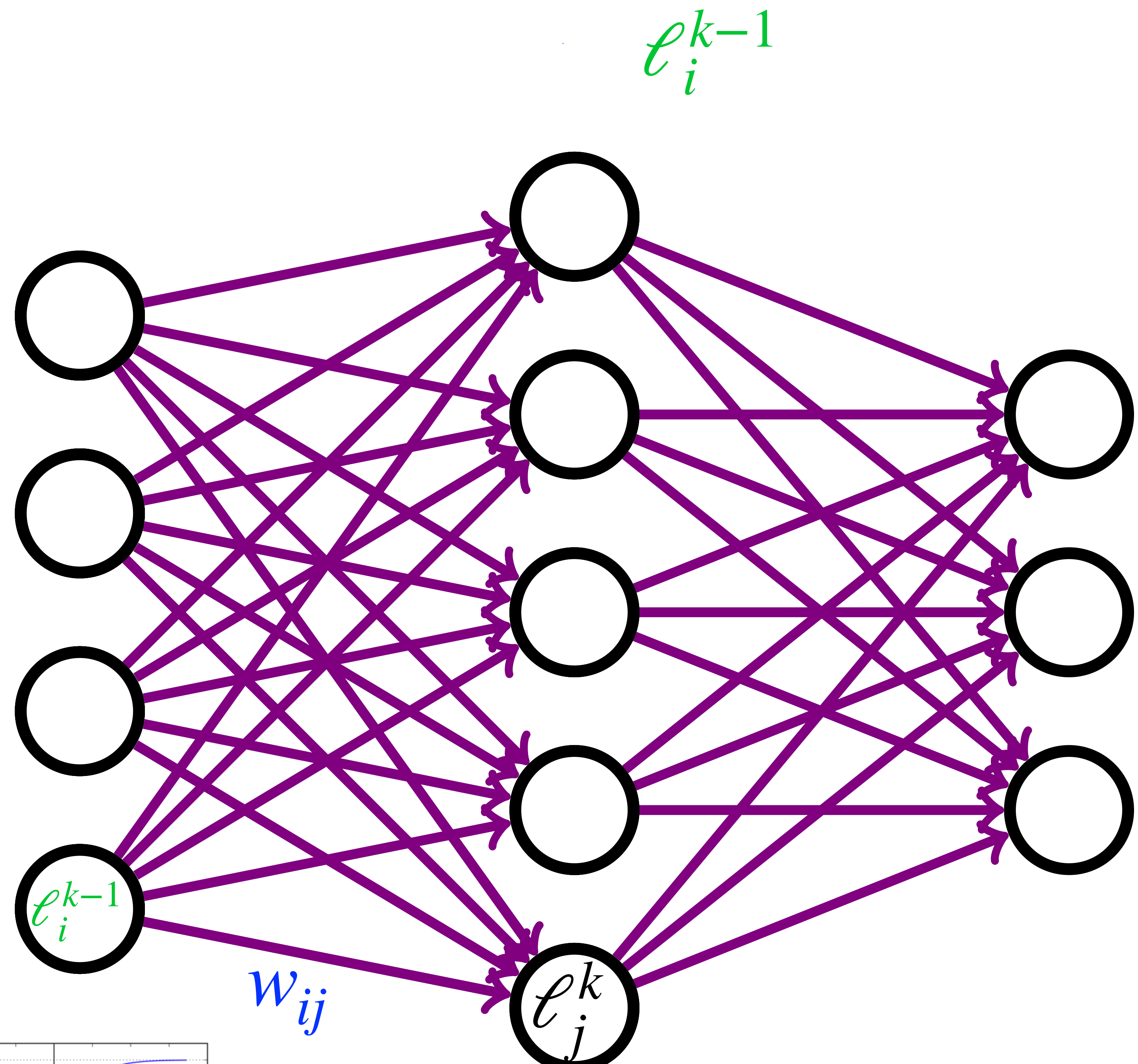
4 interaction points

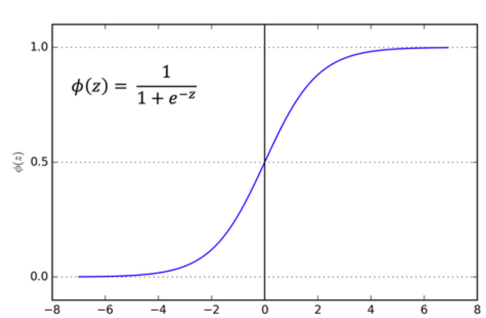
40 million collisions / second

trigger selects ~1000 collisions / second

NEURAL NETWORK (RECAP)

- Classic fully connected architecture
- Each **input** multiplied by a **weight**
- **Weighted** values are summed, **bias** is added
- Nonlinear **activation function** is applied
- Trained by varying the **parameters** to minimize a loss function (quantifies how many mistakes the network makes)





A sufficiently "wide" neural network can approximate any function!

- **Step 0**: Define the problem (choice of loss function)

- **Step 0**: Define the problem (choice of loss function)

$$L = -y\log(p) + (1-y)\log(1-p)$$

y = 0 (background) or 1 (signal)

p = output of our NN (probability of signal)

- **Step 0**: Define the problem (choice of loss function)

$$L = -y\log(p) + (1-y)\log(1-p)$$

y = 0 (background) or 1 (signal)

p = output of our NN (probability of signal)

if $p \sim y$, $L \sim 0$ (correct!)

if $p \sim 1-y$, $L \sim \infty$ (incorrect!)

- **Step 0**: Define the problem (choice of loss function)

$$L = -y\log(p) + (1-y)\log(1-p)$$

y = 0 (background) or 1 (signal)
p = output of our NN (probability of signal)

if $p \sim y$, $L \sim 0$ (correct!)
if $p \sim 1-y$, $L \sim \infty$ (incorrect!)

- Step 1: Acquire lots of labeled data and split into training and testing sets

- **Step 0**: Define the problem (choice of loss function)

$$L = -y\log(p) + (1-y)\log(1-p)$$

y = 0 (background) or 1 (signal)

p = output of our NN (probability of signal)

if $p \sim y$, $L \sim 0$ (correct!)

if $p \sim 1-y$, $L \sim \infty$ (incorrect!)

- Step 1: Acquire lots of labeled data and split into training and testing sets
- Step 2: Select input features

MACHINE LEARNING STEPS

- **Step 0**: Define the problem (choice of loss function)

$$L = -y\log(p) + (1-y)\log(1-p)$$

y = 0 (background) or 1 (signal)

p = output of our NN (probability of signal)

if $p \sim y$, $L \sim 0$ (correct!)

if $p \sim 1-y$, $L \sim \infty$ (incorrect!)

- Step 1: Acquire lots of labeled data and split into training and testing sets
- Step 2: Select input features
- Step 3: Explore/train different neural network architectures

MACHINE LEARNING STEPS

- **Step 0**: Define the problem (choice of loss function)

$$L = -y\log(p) + (1-y)\log(1-p)$$

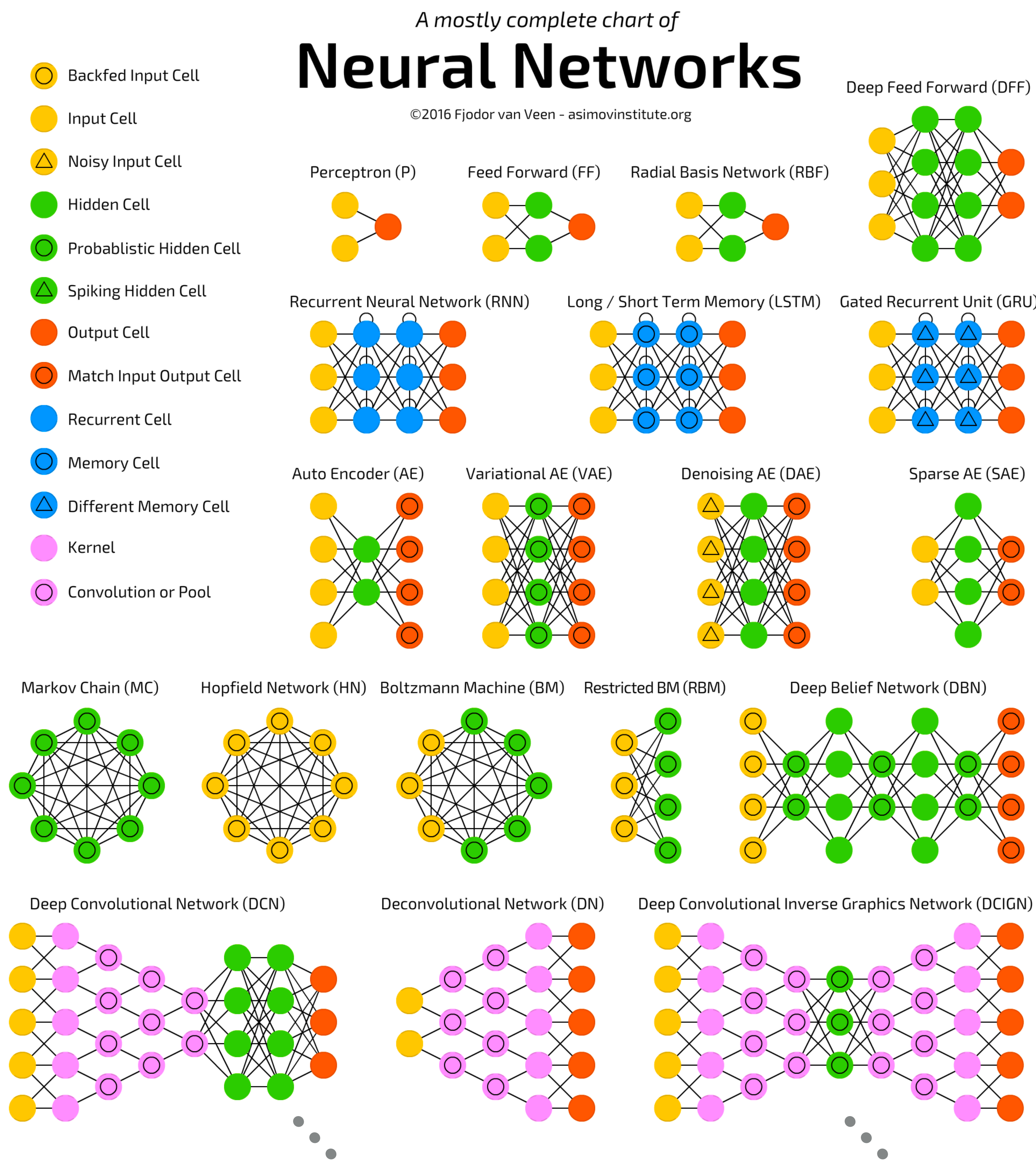
y = 0 (background) or 1 (signal)
p = output of our NN (probability of signal)

if $p \sim y$, $L \sim 0$ (correct!)
if $p \sim 1-y$, $L \sim \infty$ (incorrect!)

- Step 1: Acquire lots of labeled data and split into training and testing sets
- Step 2: Select input features
- Step 3: Explore/train different neural network architectures
- **Step 4**: Evaluate performance

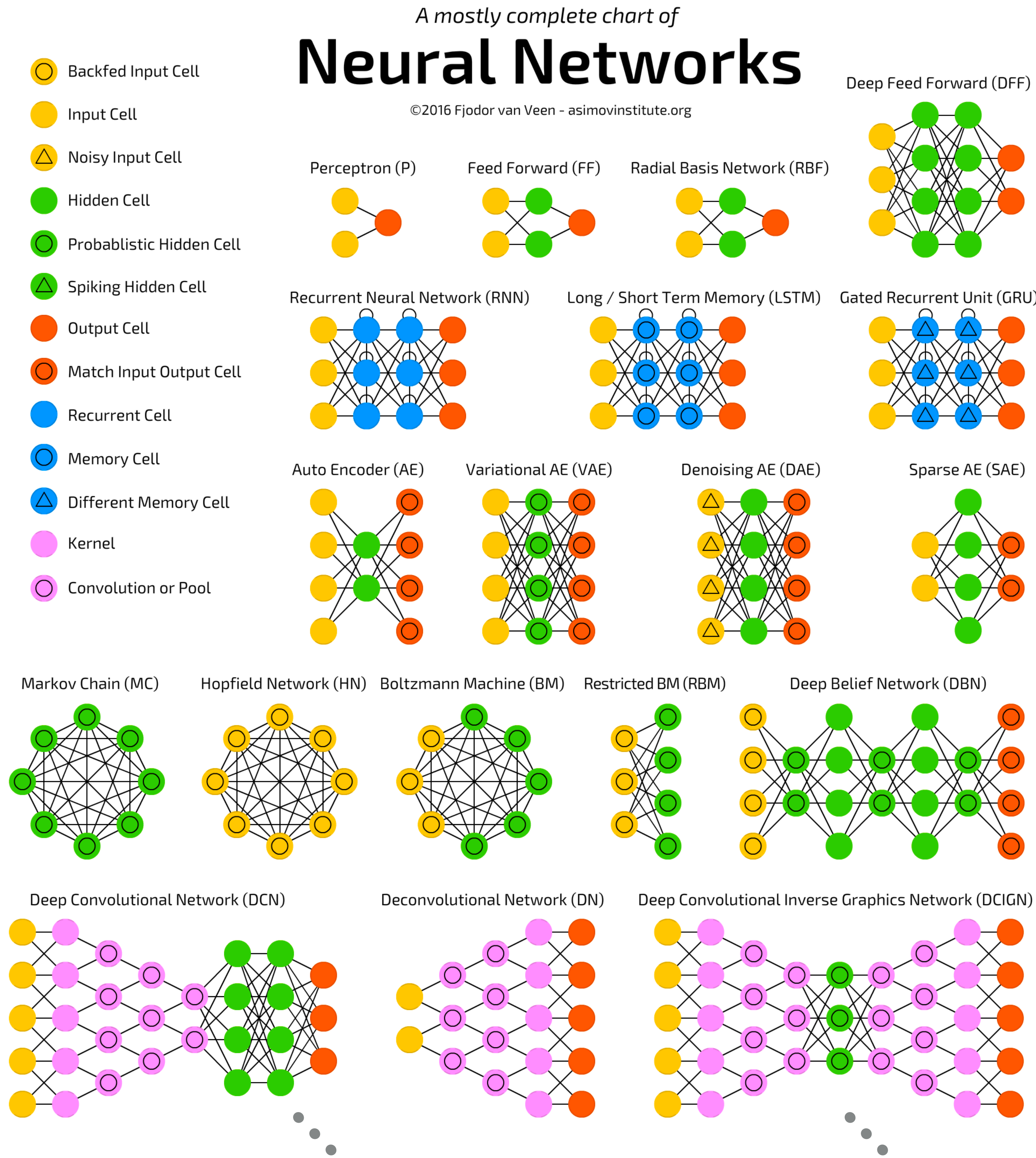
NEURAL NETWORK ZOO

- You have a task to accomplish, which can be represented as a smooth function from your inputs to the answer you want



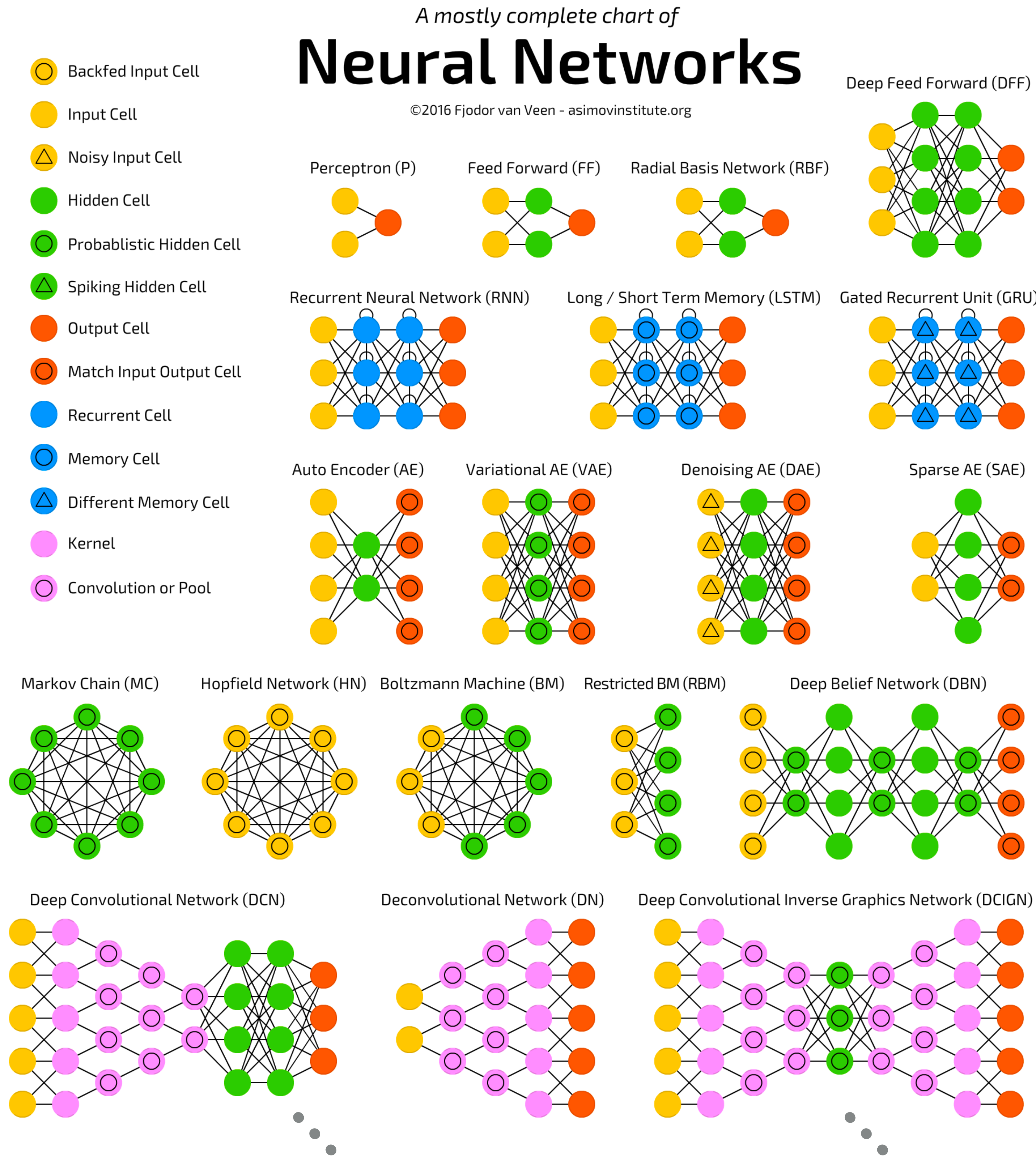
NEURAL NETWORK ZOO

- You have a task to accomplish, which can be represented as a smooth function from your inputs to the answer you want
  - Train an algorithm to learn an approximation of the optimal solution function (Machine Learning)



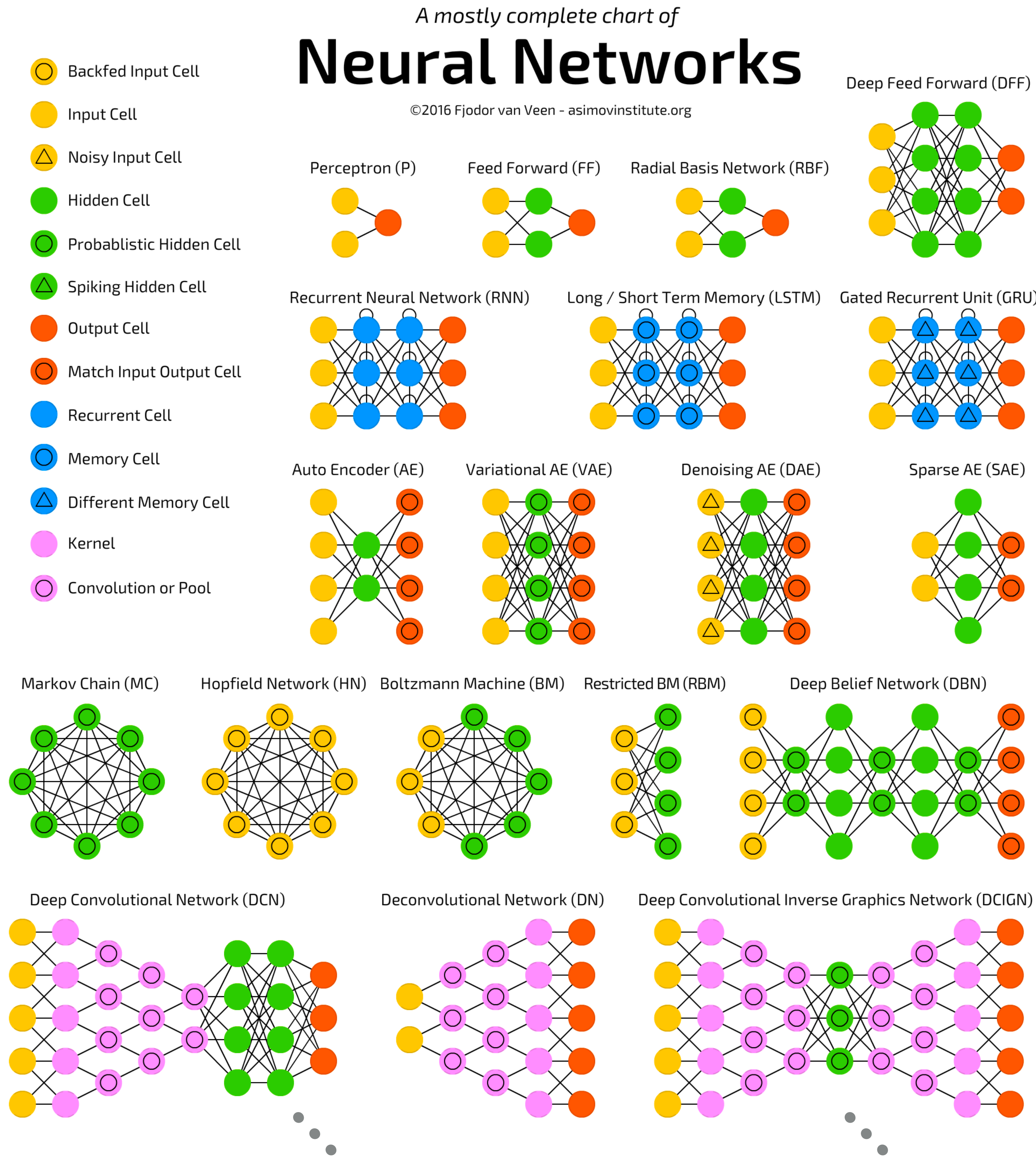
NEURAL NETWORK ZOO

- You have a task to accomplish, which can be represented as a smooth function from your inputs to the answer you want
    - Train an algorithm to learn an approximation of the optimal solution function (Machine Learning)
- NNs are the best ML solution on the market *today*



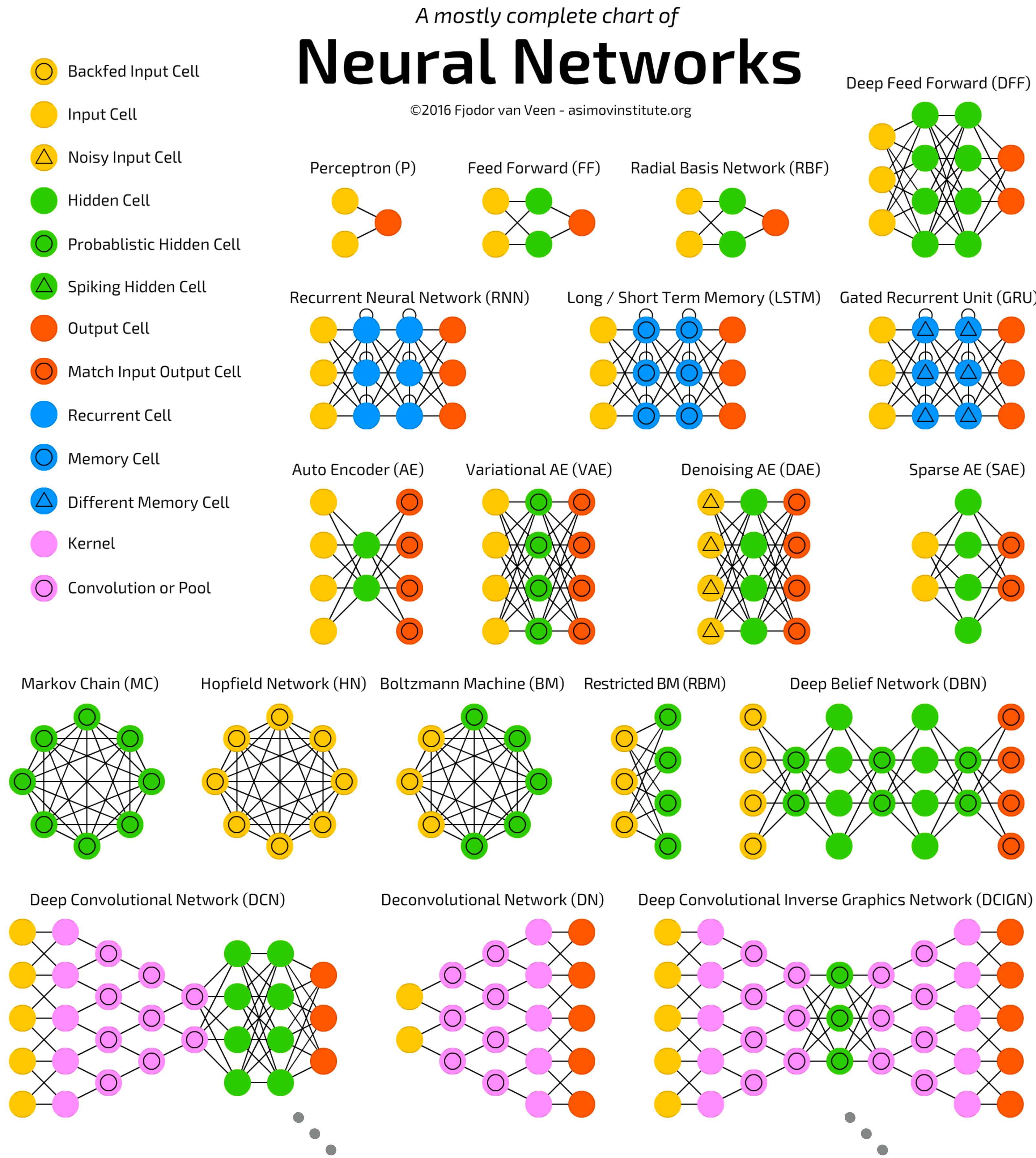
NEURAL NETWORK ZOO

- You have a task to accomplish, which can be represented as a smooth function from your inputs to the answer you want
  - Train an algorithm to learn an approximation of the optimal solution function (Machine Learning)
- NNs are the best ML solution on the market *today*
  - Each node performs a math operation on the input



NEURAL NETWORK ZOO

- You have a task to accomplish, which can be represented as a smooth function from your inputs to the answer you want
  - Train an algorithm to learn an approximation of the optimal solution function (Machine Learning)
- NNs are the best ML solution on the market *today*
  - Each node performs a math operation on the input
  - Edges represent the flow of nodes' inputs & outputs



TRAINING

- A network is trained by specifying inputs, targets, and a loss function
  - Target is what the network should learn for that input, can be a “truth” label (supervised) or the input itself (unsupervised)
  - Loss function quantifies how many mistakes the network makes
- Training is the minimization of the loss function by varying the network parameters