Exercises

Introduction and practical details

Ilaria Luise - Sofia Vallecorsa | SCS 17th October 2024

Setup a kaggle account

https://www.kaggle.com/

kaggle Competitions Datasets Models Code Discussions Courses ••• Search **Sign In** **Register**

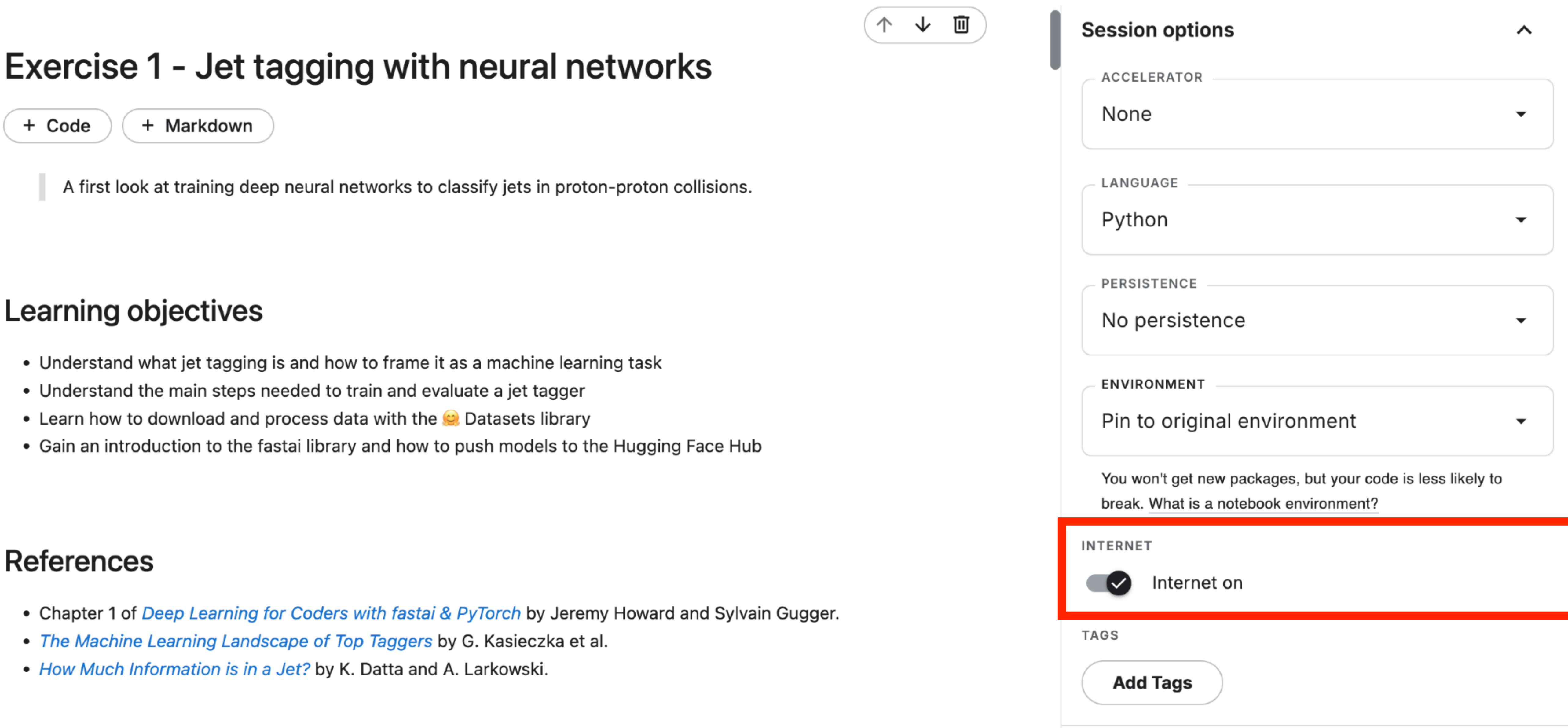
Level up with the largest AI & ML community

Join over 20M+ machine learners to share, stress test, and stay up-to-date on all the latest ML techniques and technologies. Discover a huge repository of community-published models, data & code for your next project.

Register with Google **Register with Email**

Enable internet connection

1) Go to session options
2) Verify your phone if you haven't done so
3) Enable internet on the node



(Optional) Weights and biases

1) Go to: https://wandb.ai/site

2) Login instructions: need to copy the API key when asked

Quickstart: Tracking your first run in Weights & Biases

Weights & Biases' tools make it easy for you to quickly track experiments, visualize results, spot regressions, and more. Simply put, Weights & Biases enables you to build better models faster and easily share findings with colleagues.

Visualize your model training with Python / Pytorch or **Open in Colab**

1. Set up the wandb library

Install the CLI and Python library for interacting with the Weights and Biases API.

```
pip install wandb
```

Next, log in and paste your API key when prompted.

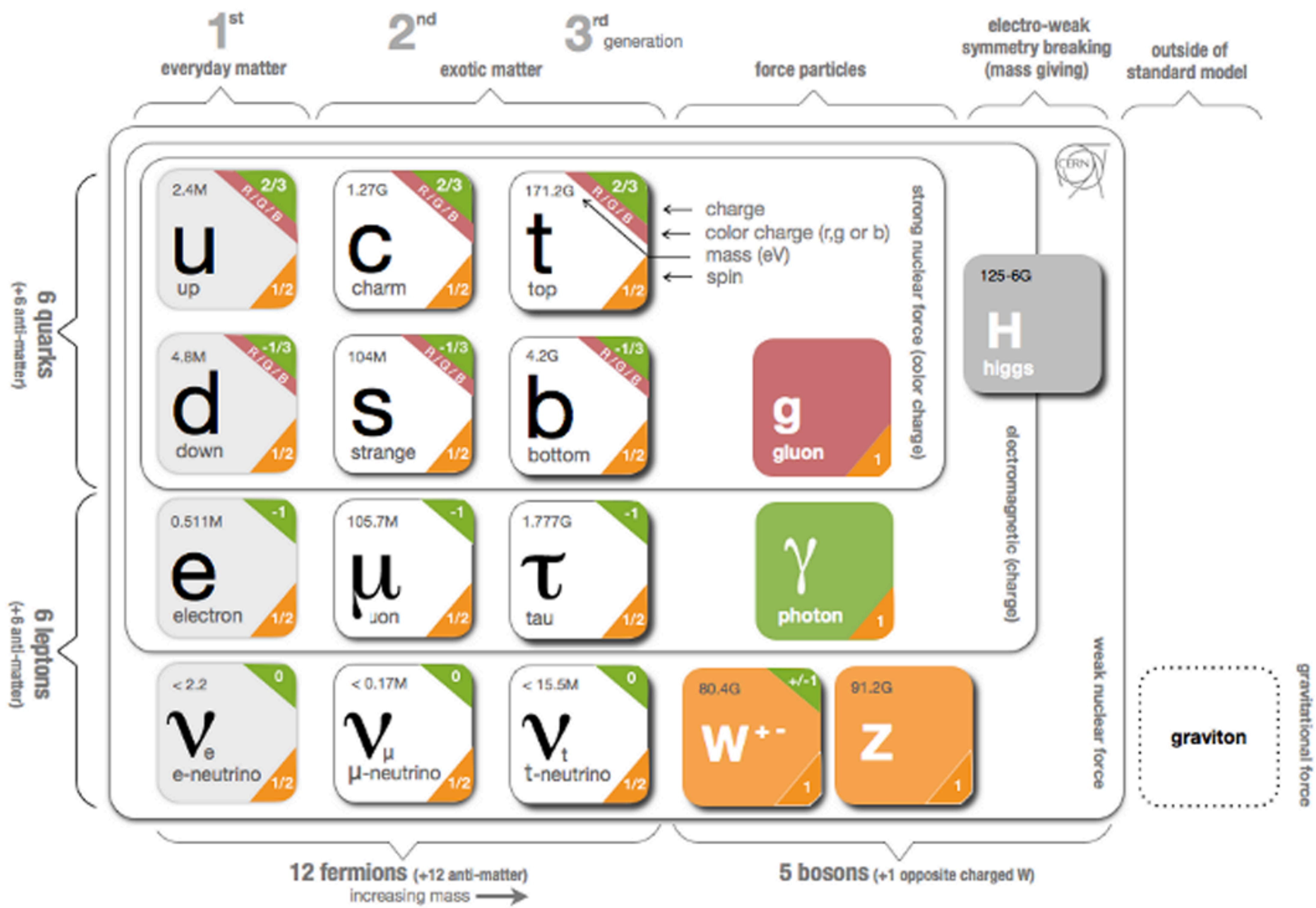
```
wandb login
```

Particle Tagging

What is it and why is it important?

Ilaria Luise - Sofia Vallecorsa | SCS 17th October 2024

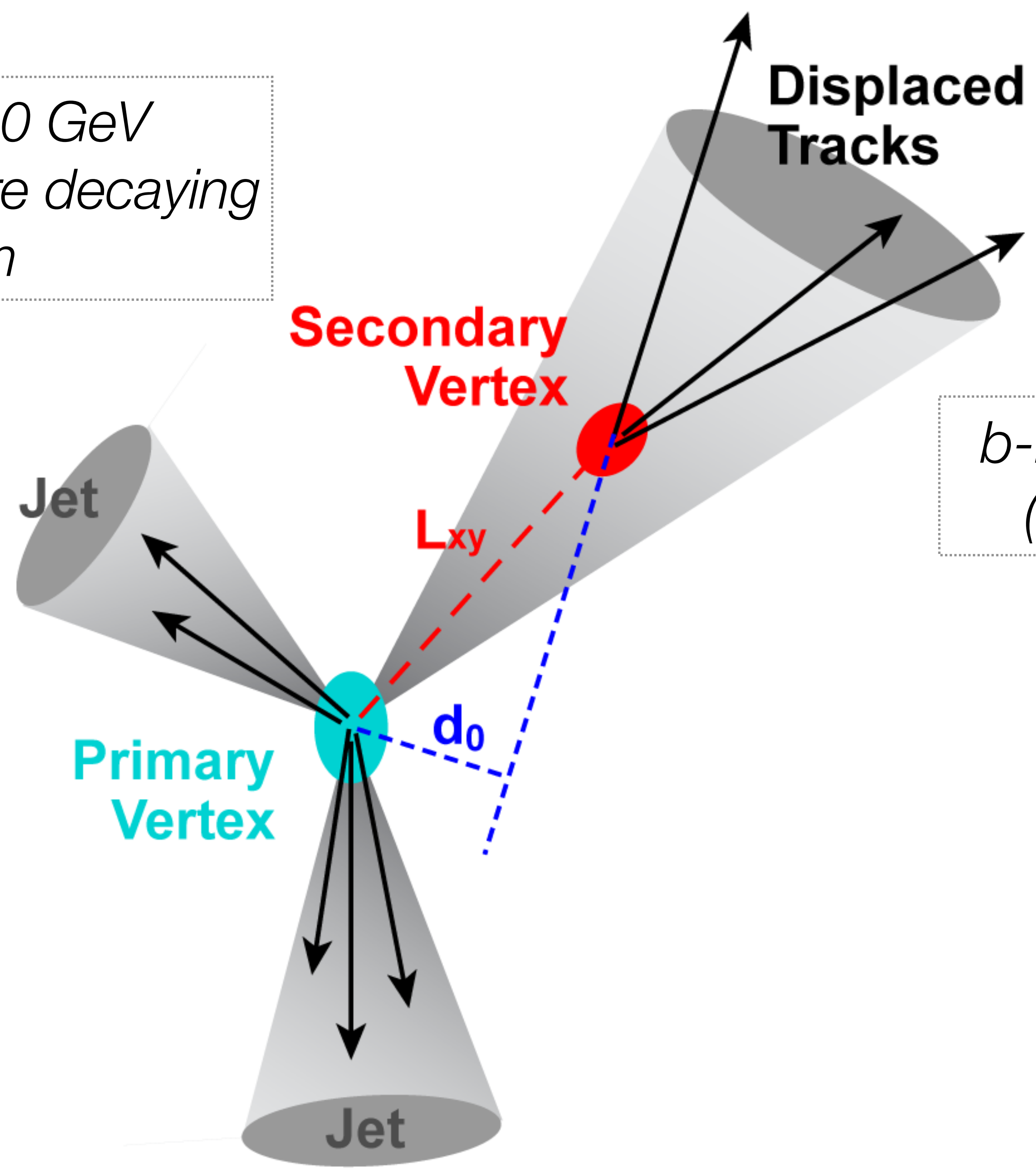
The standard model



Flavour tagging

b-tagging: Separate b-jets from light (u,d,s,g) and c-jets using specific b-hadron properties

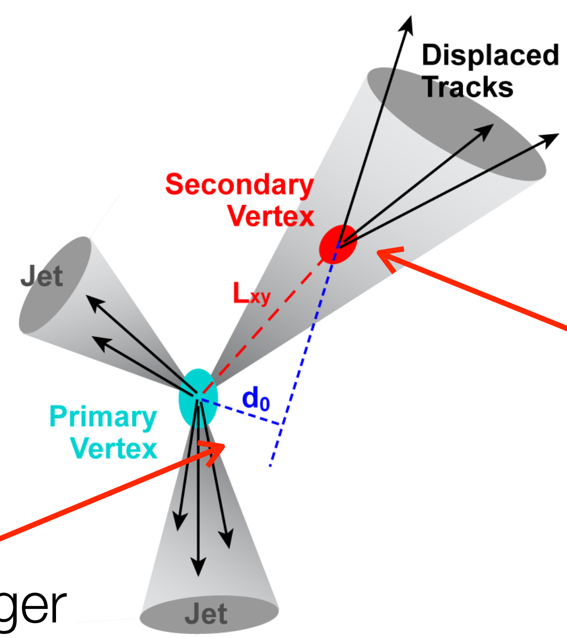
b-hadron @ 50 GeV
flies few mm before decaying
$c\tau$~3mm



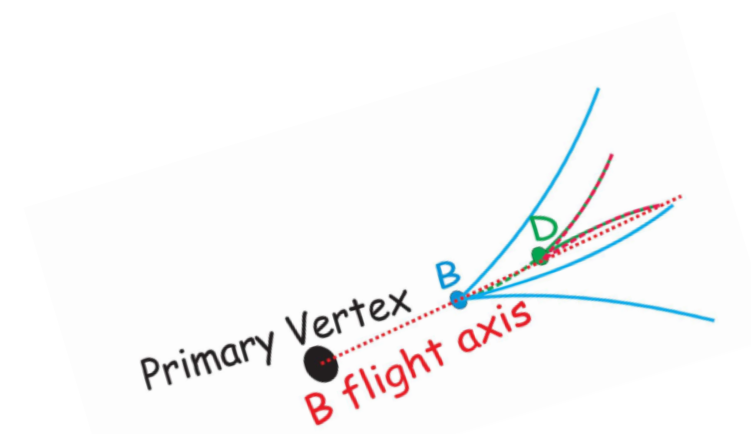
b-hadrons decay likely (85%) in c-hadrons

Flavour tagging

b-tagging: Separate b-jets from light (u,d,s,g) and c-jets using specific b-hadron properties



2 Secondary Vertex Reconstruction
SV1



1 Impact parameter tagger (IP significance)
IP2D,IP3D

3 Reconstruct hadron decay chain (B→D→…)
JetFitter

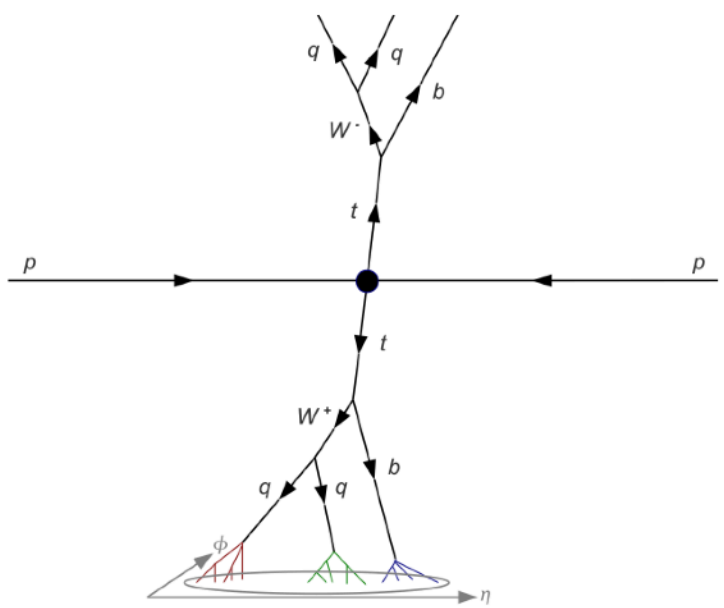
Features combined with Machine learning techniques

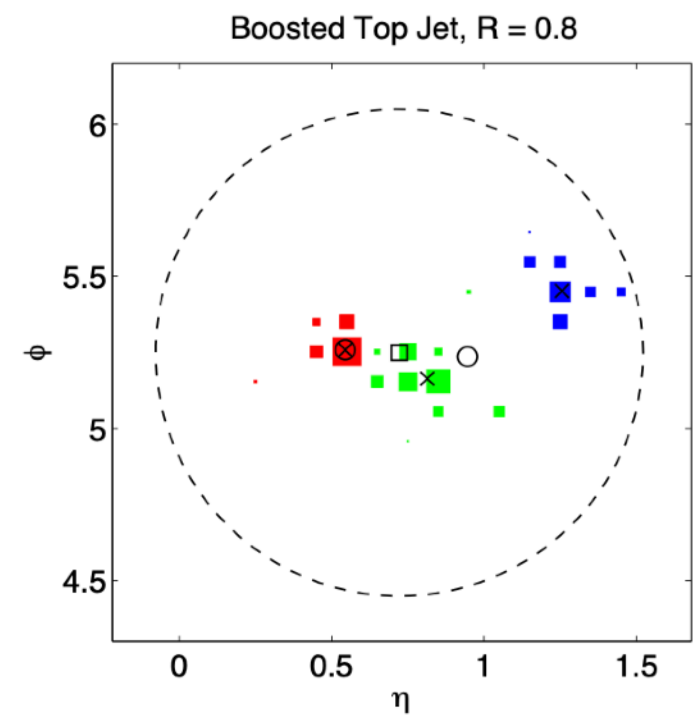
ParticleNet (CMS)

GN1 (single jet) - GN2X (boosted) (ATLAS)

Top tagging

top-tagging: identify boosted top-jets topologies

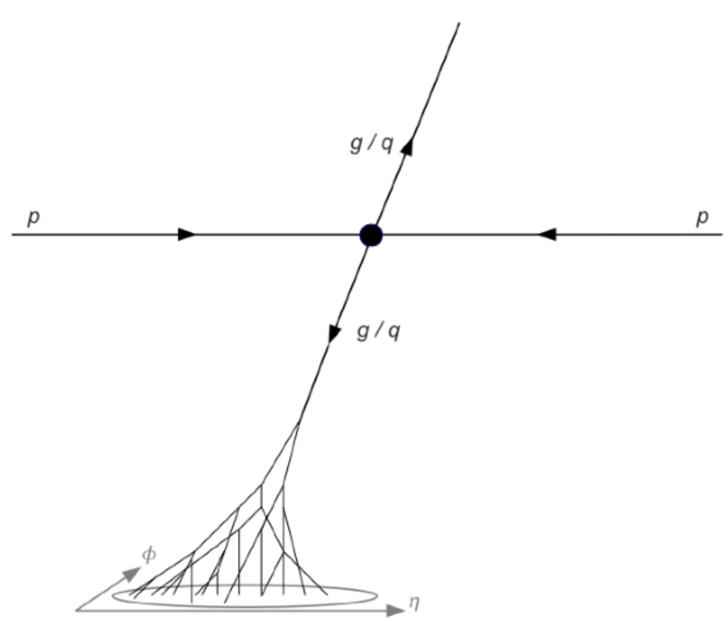


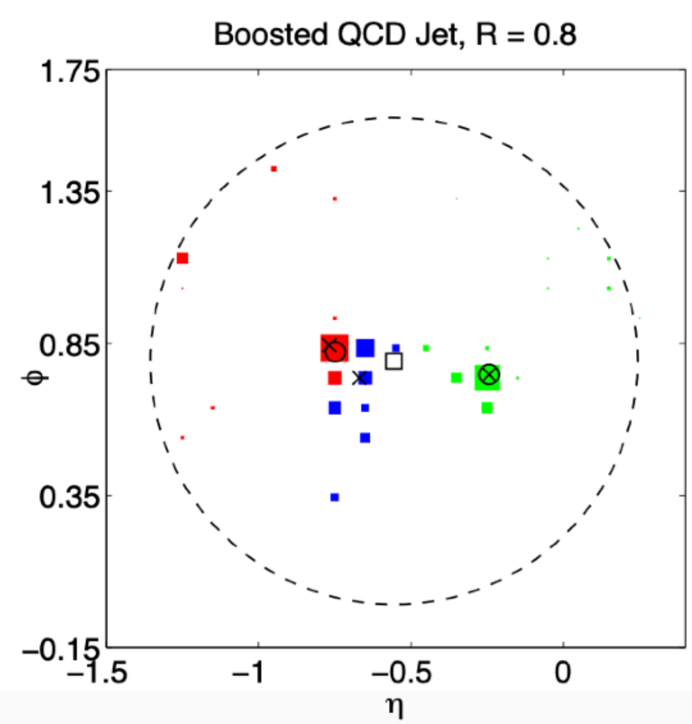


n-subjectiness

(a)

(b)

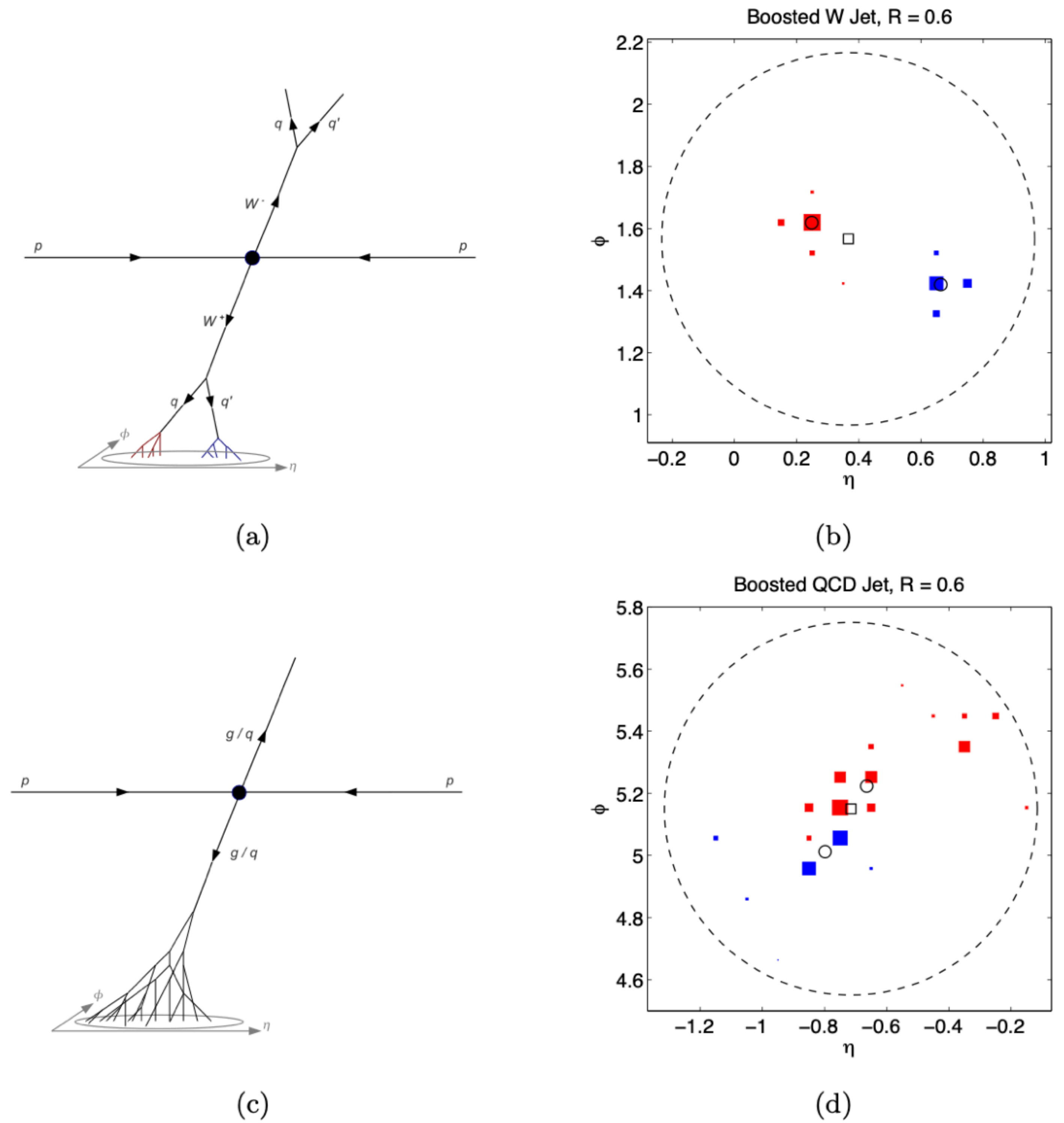




Top tagging

Other backgrounds

n-subjectiness



ATLAS: top taggers

Tagger	Number of parameters	Inference time
hlDNN	133,381	3 ms
DNN	876,641	3 ms
EFN	959,251	4 ms
PFN	754,501	3 ms
ResNet 50	1,499,585	20 ms
ParticleNet	764,887	143 ms

https://arxiv.org/pdf/2407.20127

Accuracies

Tagger	AUC	ACC	ε_{bkg}^{-1} @ $\varepsilon_{sig} = 0.5$	ε_{bkg}^{-1} @ $\varepsilon_{sig} = 0.8$
ResNet 50	0.872 ± 0.006	0.787 ± 0.006	18.4 ± 1.1	4.63 ± 0.2
EFN	0.894 ± 0.001	0.810 ± 0.001	23.8 ± 0.5	5.74 ± 0.07
hlDNN	0.9374 ± 0.0001	0.8628 ± 0.0002	47.2 ± 0.4	10.36 ± 0.03
DNN	0.9447 ± 0.0004	0.8715 ± 0.0008	73.0 ± 1.3	12.5 ± 0.1
PFN	0.9502 ± 0.0004	0.878 ± 0.001	92.7 ± 1.8	14.6 ± 0.2
ParticleNet	0.9614 ± 0.0005	0.895 ± 0.001	155.8 ± 3.8	20.6 ± 0.4

https://arxiv.org/pdf/2407.20127

Performance

	AUC	Acc	$1/\epsilon_B$ ($\epsilon_S = 0.3$) single	mean	median	#Param
CNN [16]	0.981	0.930	914±14	995±15	975±18	610k
ResNeXt [31]	0.984	0.936	1122±47	1270±28	1286±31	1.46M
TopoDNN [18]	0.972	0.916	295±5	382± 5	378 ± 8	59k
Multi-body N-subjettiness 6 [24]	0.979	0.922	792±18	798±12	808±13	57k
Multi-body N-subjettiness 8 [24]	0.981	0.929	867±15	918±20	926±18	58k
TreeNiN [43]	0.982	0.933	1025±11	1202±23	1188±24	34k
P-CNN	0.980	0.930	732±24	845±13	834±14	348k
ParticleNet [47]	0.985	0.938	1298±46	1412±45	1393±41	498k
LBN [19]	0.981	0.931	836±17	859±67	966±20	705k
LoLa [22]	0.980	0.929	722±17	768±11	765±11	127k
LDA [54]	0.955	0.892	151±0.4	151.5±0.5	151.7±0.4	184k
Energy Flow Polynomials [21]	0.980	0.932	384			1k
Energy Flow Network [23]	0.979	0.927	633±31	729±13	726±11	82k
Particle Flow Network [23]	0.982	0.932	891±18	1063±21	1052±29	82k
GoaT	0.985	0.939	1368±140		1549±208	35k

You are going to train your own CNN based tagger:
How does it compare to these ones?

Let's get started:

https://iluise.github.io/dl4phys/intro.html