SEARCHING FOR DARK MATTER WITH MACHINE LEARNING

SCOTT KRAVITZ

PHYS290E

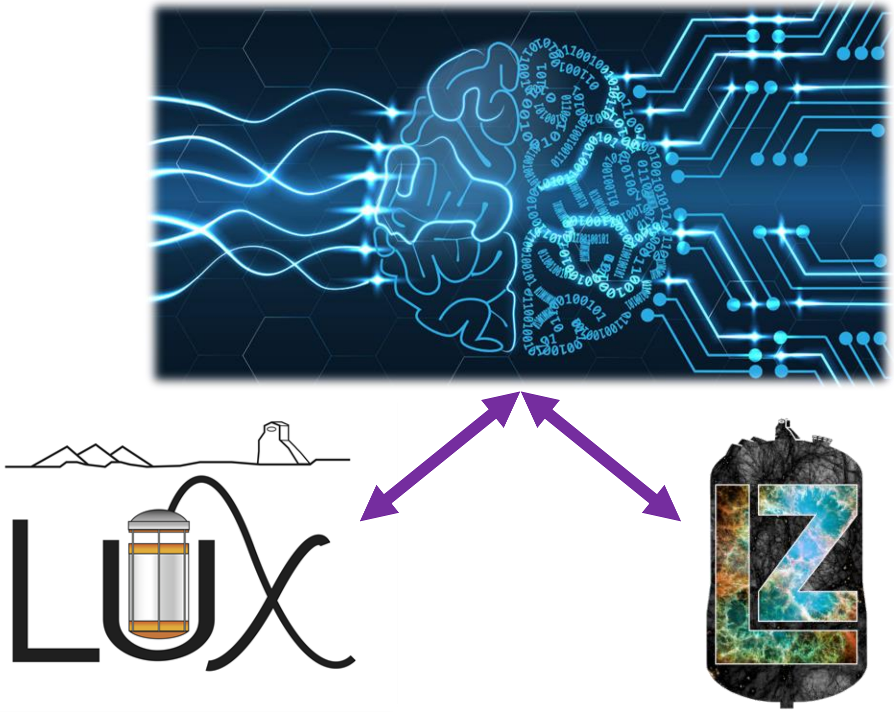
FEBRUARY 9, 2022

MY BACKGROUND

- Grad school: Stanford, EXO (neutrinoless DBD)
- Now a Chamberlain postdoc at LBL
- Lead LUX/LZ dark matter experiments' ML groups
- ML is relatively new in rare event searches!
    - Lots of room for exploration
    - ...also lots of suspicion of ML "black box"
- My nefarious agenda:
    - Make our (physicists') own ML techniques/apps
    - Emphasize interpretability, reliability, quantified uncertainties
    - Find a common language for problems/solutions across collaborations to maximize ML benefits

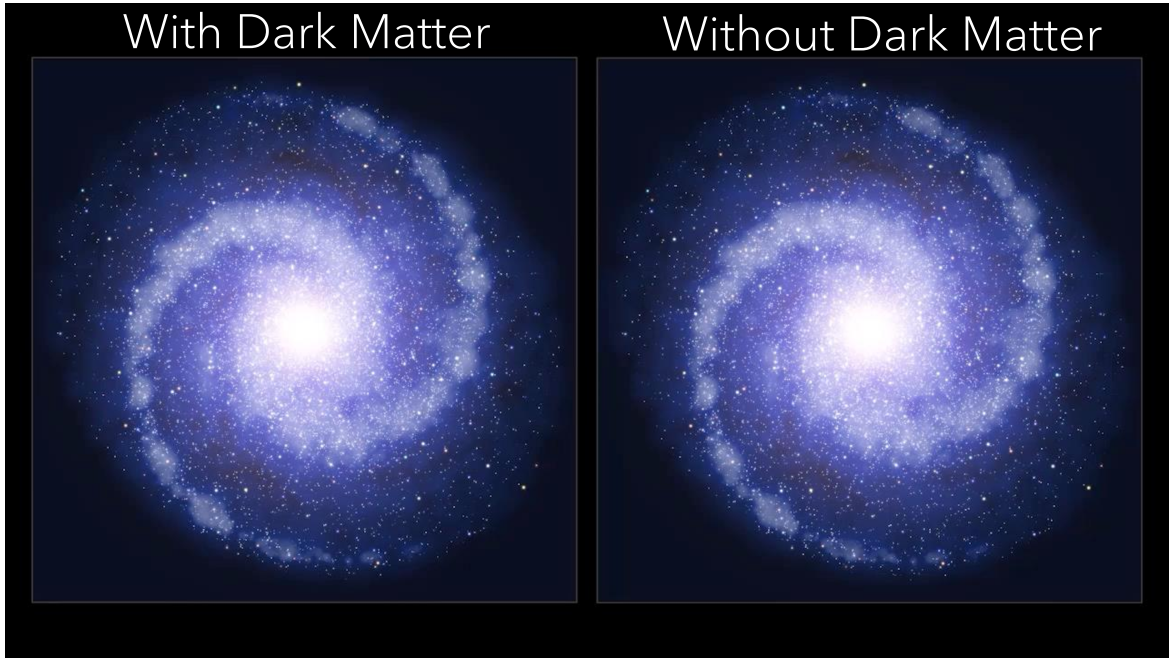
OUTLINE

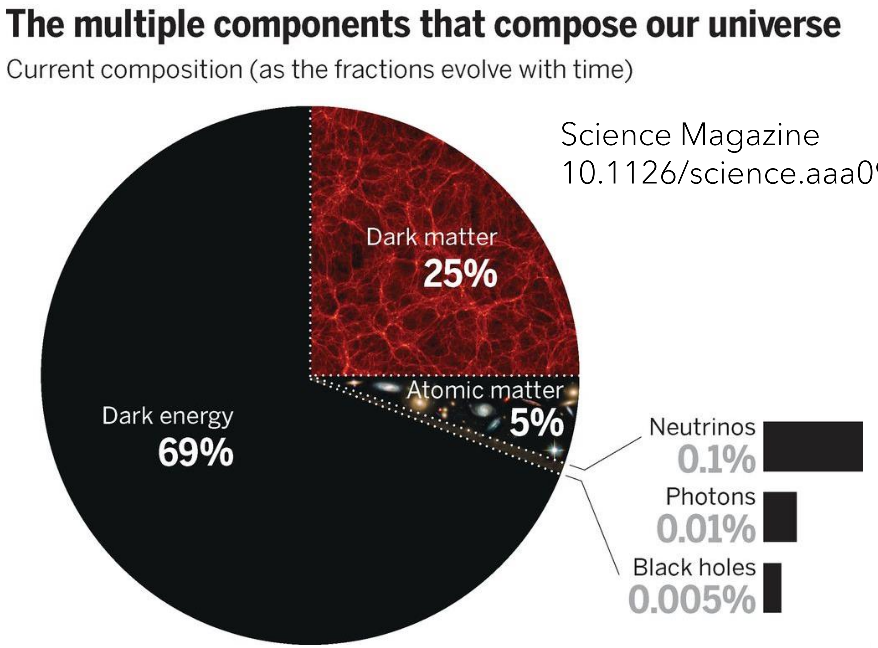
1. Dark matter and direct detection
2. The LUX and LZ dark matter experiments
3. Quick intro to machine learning
4. Improved DM analysis with physics-oriented machine learning
5. ML resources + tutorial overview



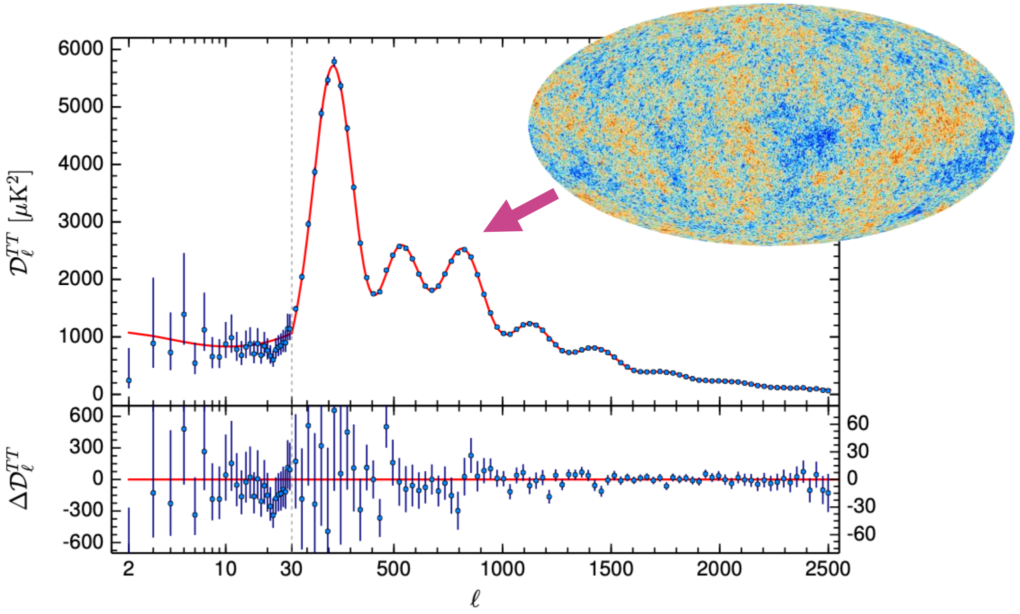
DARK MATTER

- Detected through gravitational effects
- Particle properties remain unknown!
- Range of candidate particle properties is staggering

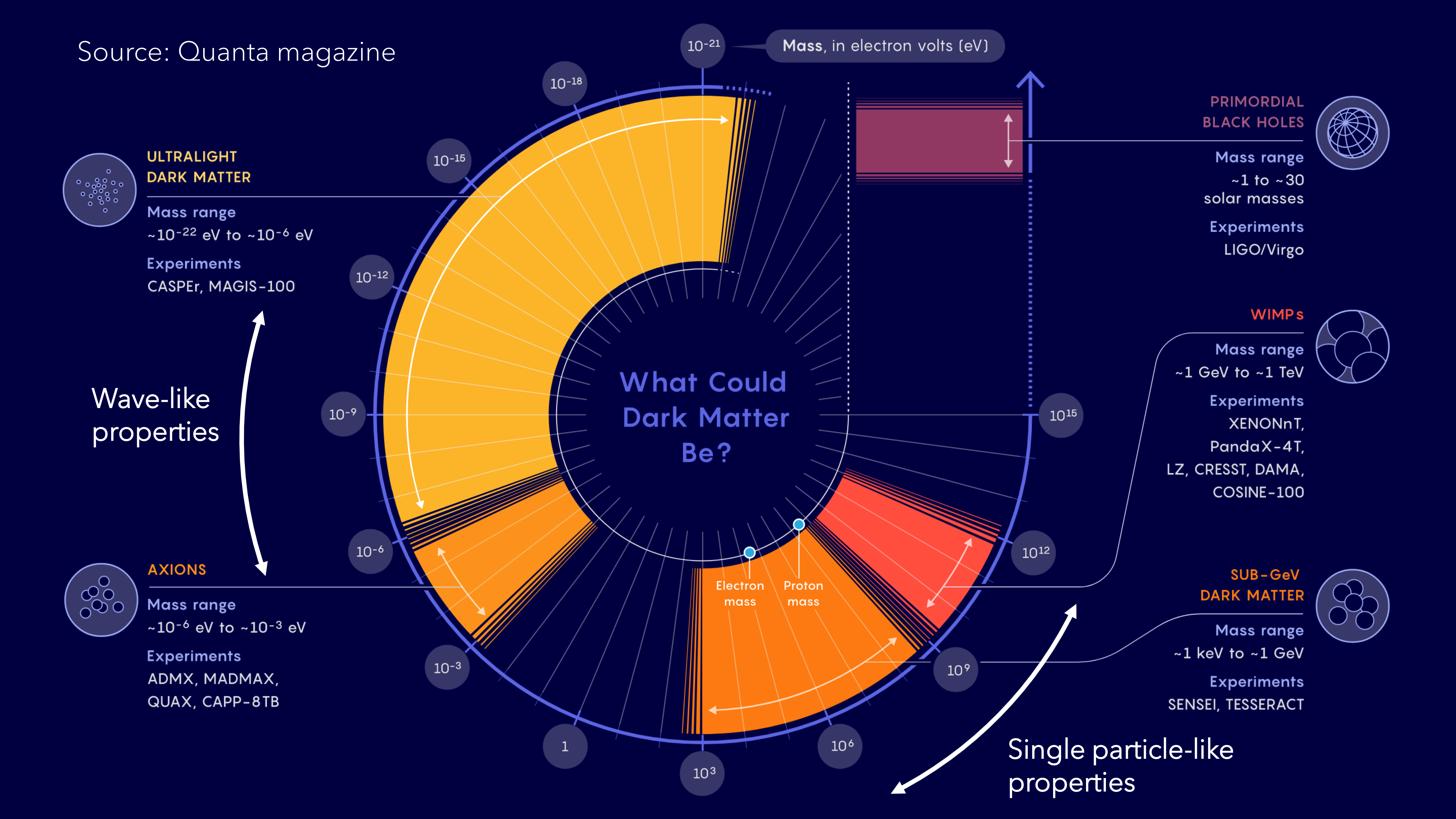




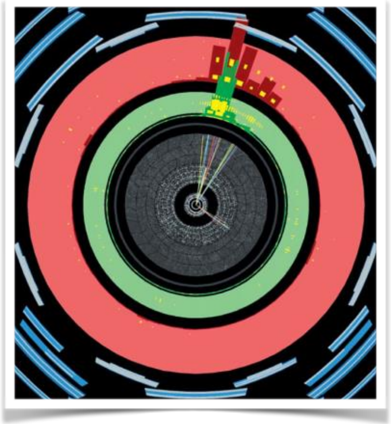
Science Magazine
10.1126/science.aaa0980

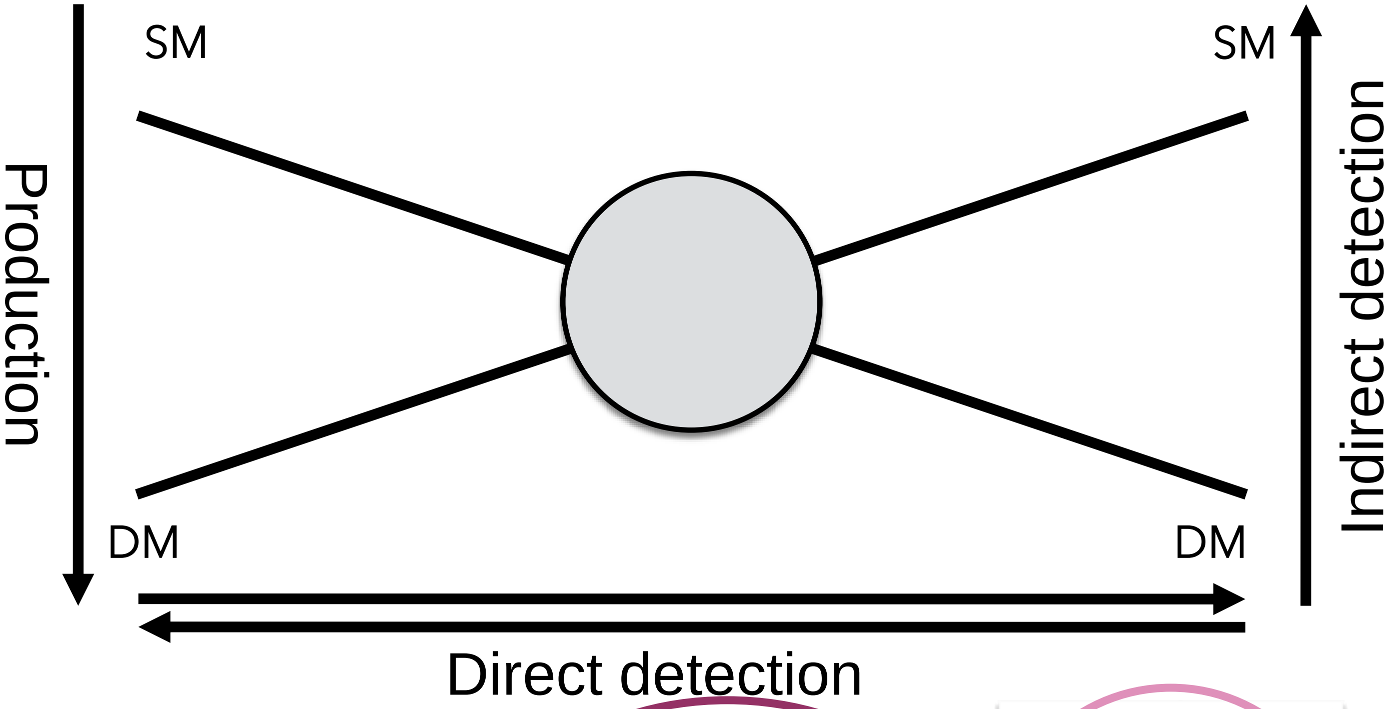


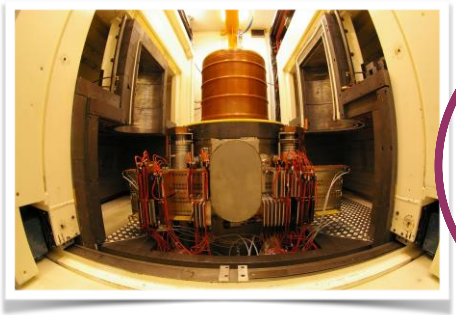
Planck Collab. Astron. Astrophys. 594 (2016) A13

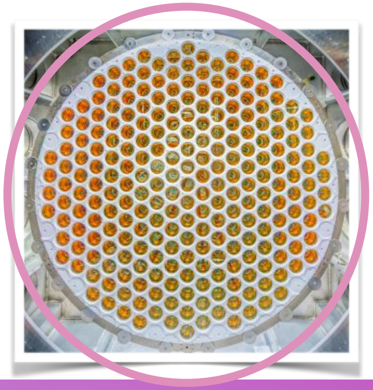

Source: Quanta magazine
Mass, in electron volts (eV)
What Could Dark Matter Be?
ULTRALIGHT DARK MATTER
Mass range
~10^-22 eV to ~10^-6 eV
Experiments
CASPEr, MAGIS-100
AXIONS
Mass range
~10^-6 eV to ~10^-3 eV
Experiments
ADMX, MADMAX, QUAX, CAPP-8TB
PRIMORDIAL BLACK HOLES
Mass range
~1 to ~30 solar masses
Experiments
LIGO/Virgo
WIMPs
Mass range
~1 GeV to ~1 TeV
Experiments
XENONnT, PandaX-4T, LZ, CRESST, DAMA, COSINE-100
SUB-GeV DARK MATTER
Mass range
~1 keV to ~1 GeV
Experiments
SENSEI, TESSERACT
Electron mass
Proton mass
Wave-like properties
Single particle-like properties
10^-21
10^-18
10^-15
10^-12
10^-9
10^-6
10^-3
1
10^3
10^6
10^9
10^12
10^15

DARK MATTER DETECTION STRATEGIES



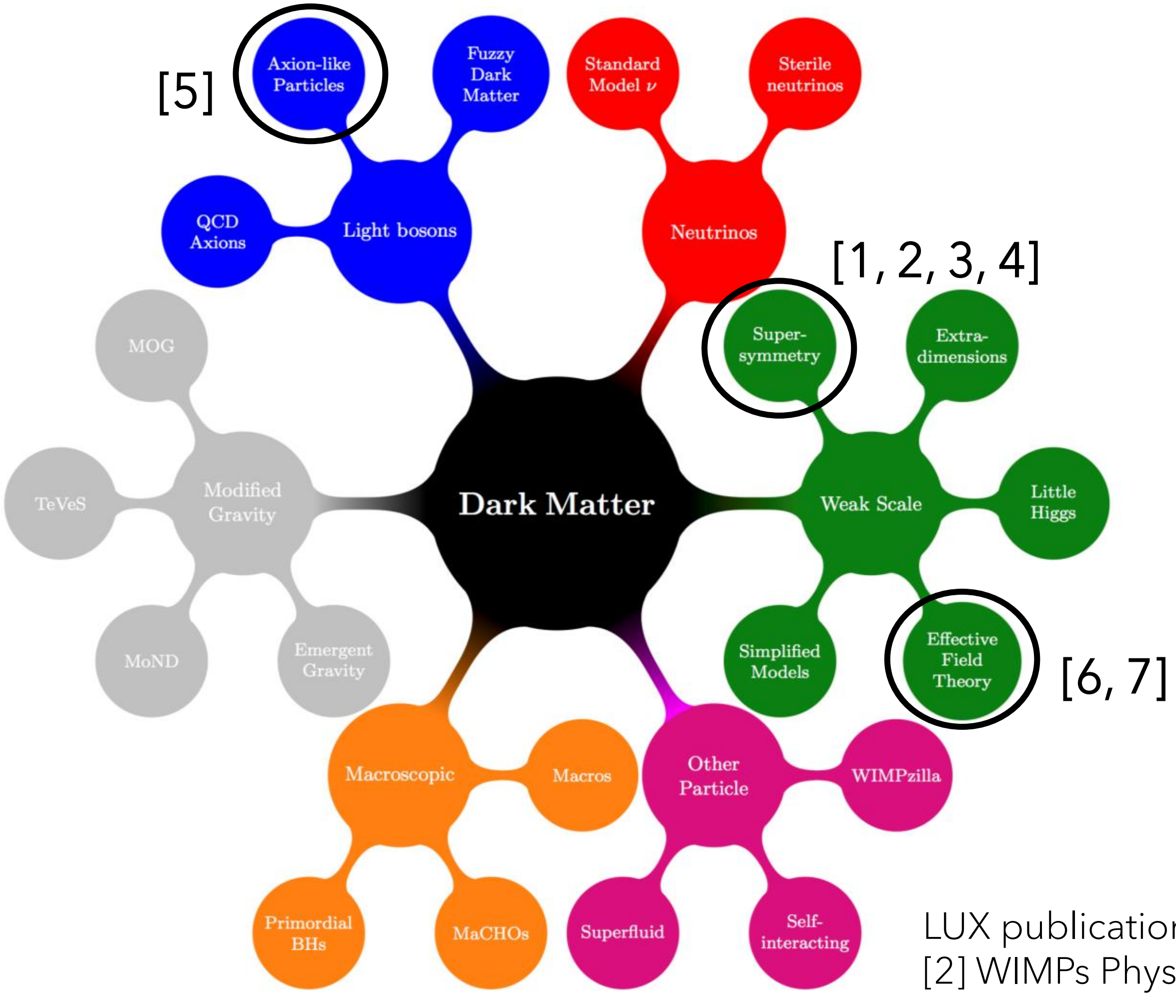


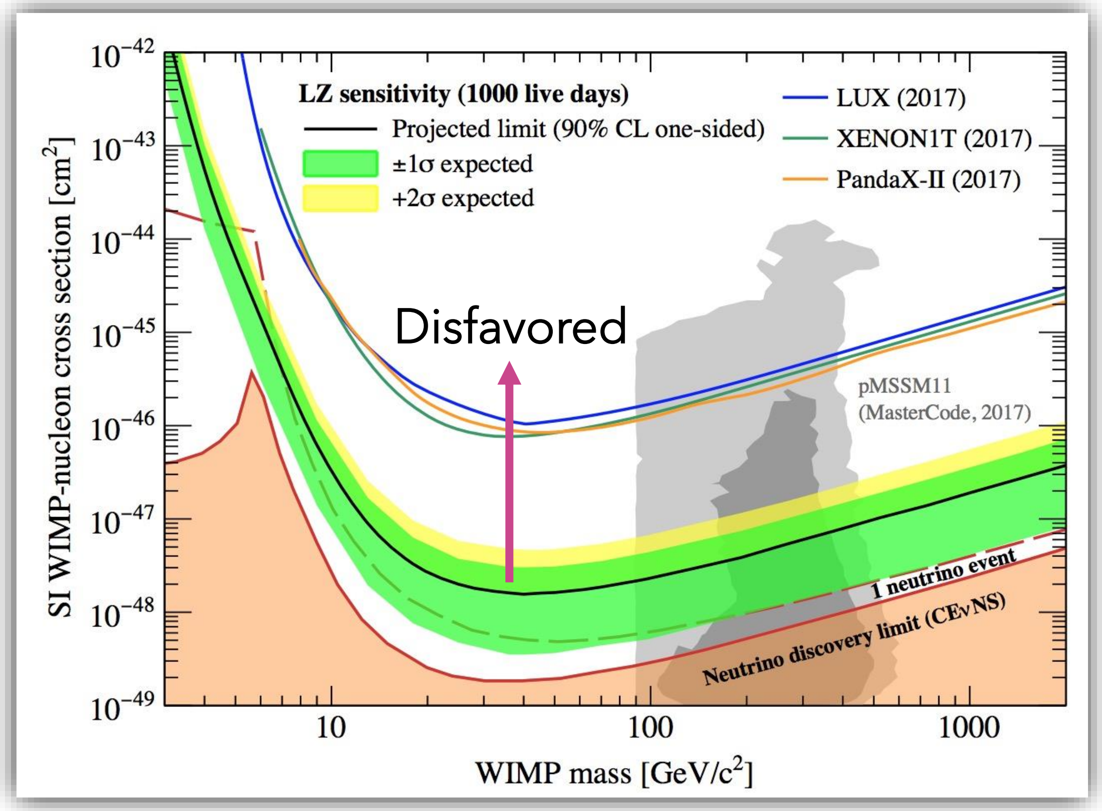




LZ

PROBING DM WITH DIRECT DETECTION





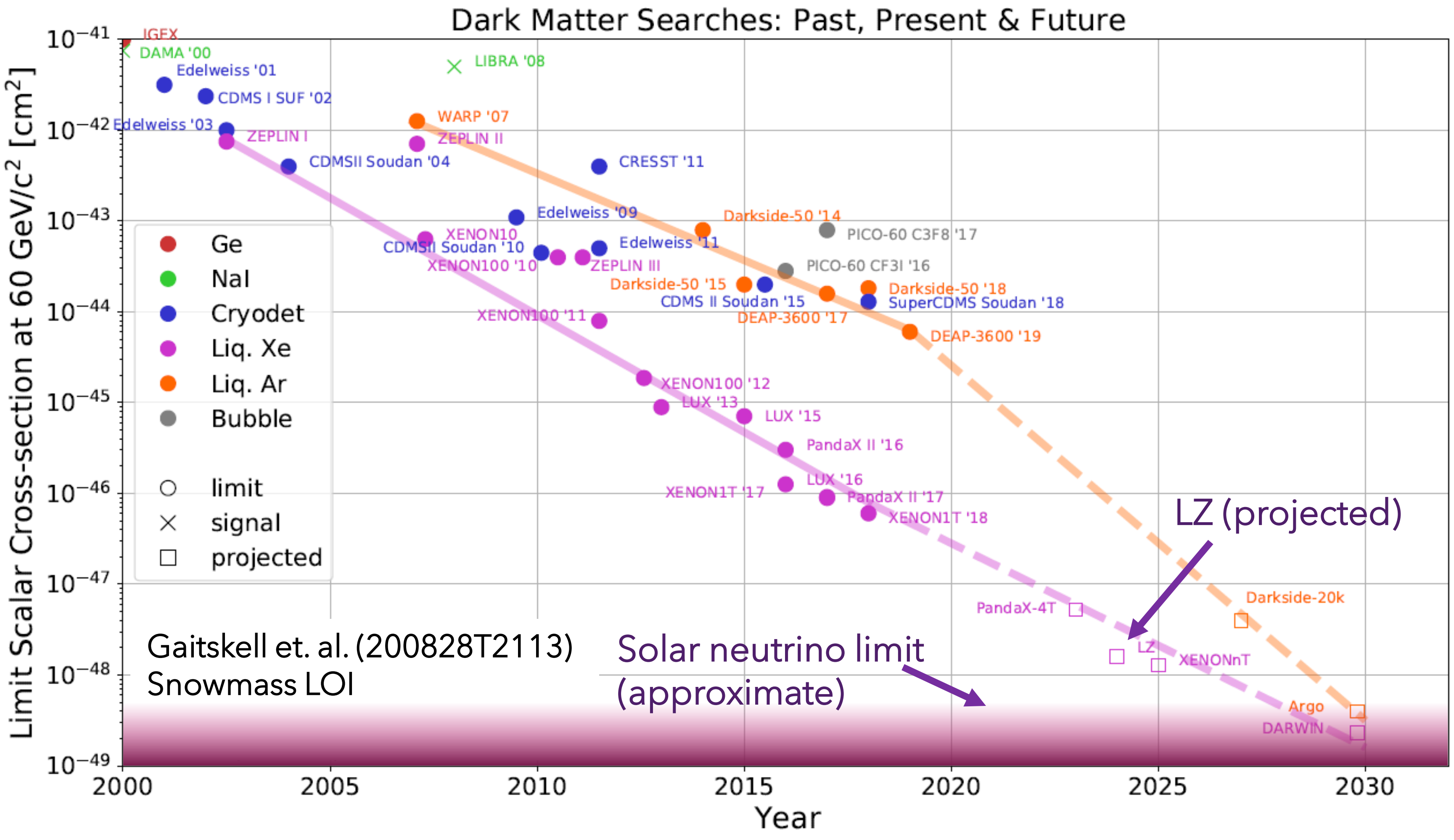
[1] Figure from LZ. PhysRevD.101.052002

LUX publications:
[2] WIMPs PhysRevLett.118.021303
[3] Mirror DM PhysRevD.101.012003
[4] Sub-GeV DM PhysRevLett.122.131301
[5] Axion-like particles PhysRevLett.118.261301
[6] EFT (2013) PhysRevD.103.122005
[7] EFT (2014-2016) PhysRevD.104.062005

Credit: Tim Tait (blog)

THE FUTURE OF DIRECT DETECTION

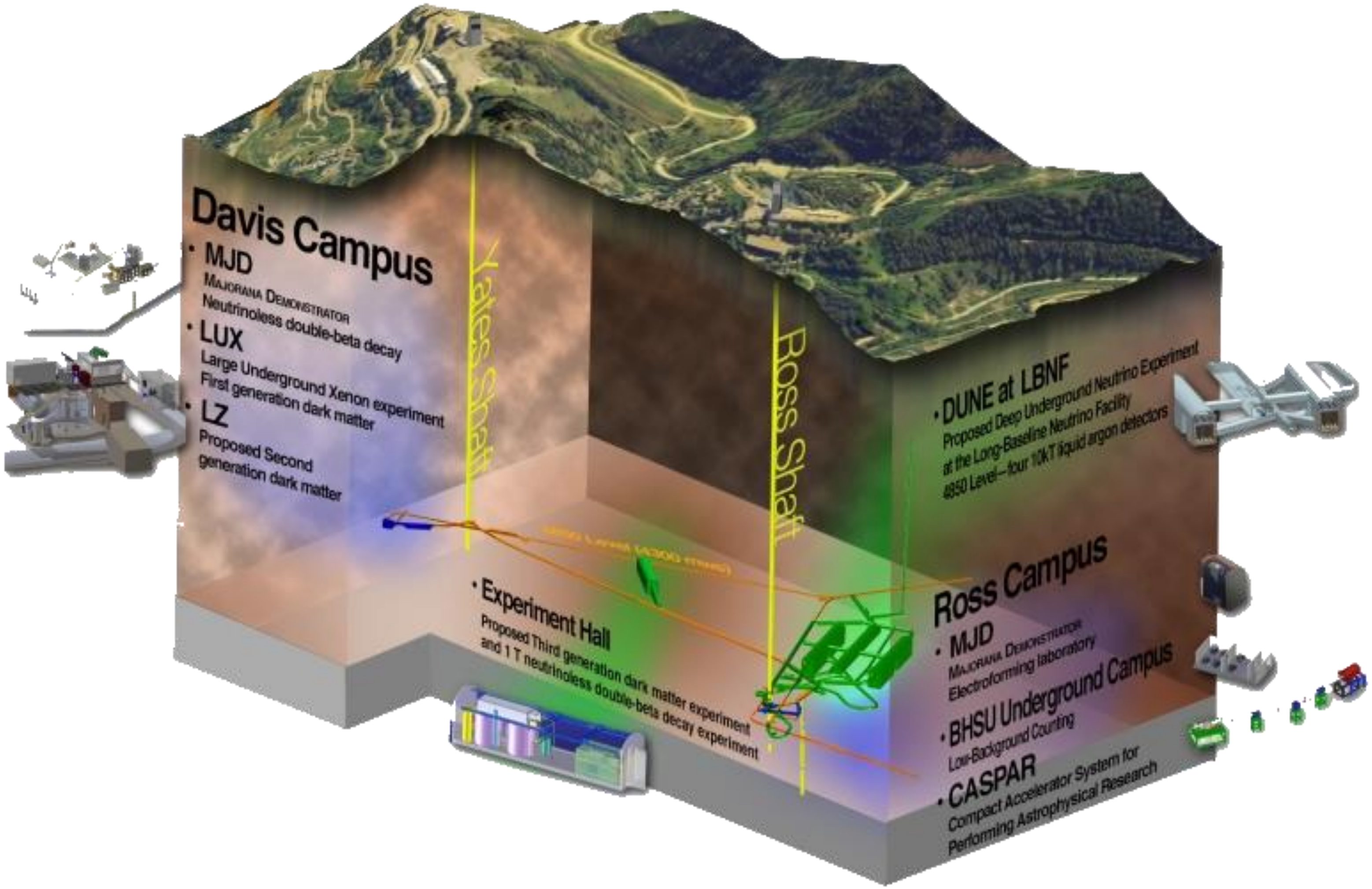
- Ultimate goal: detect DM or reach neutrino floor/fog
- Xe detectors leading the way for WIMP dark matter
- Simply increasing detector size likely insufficient!
- Must continue innovating from both detector design and data analysis angles




THE LUX AND LZ DARK MATTER EXPERIMENTS

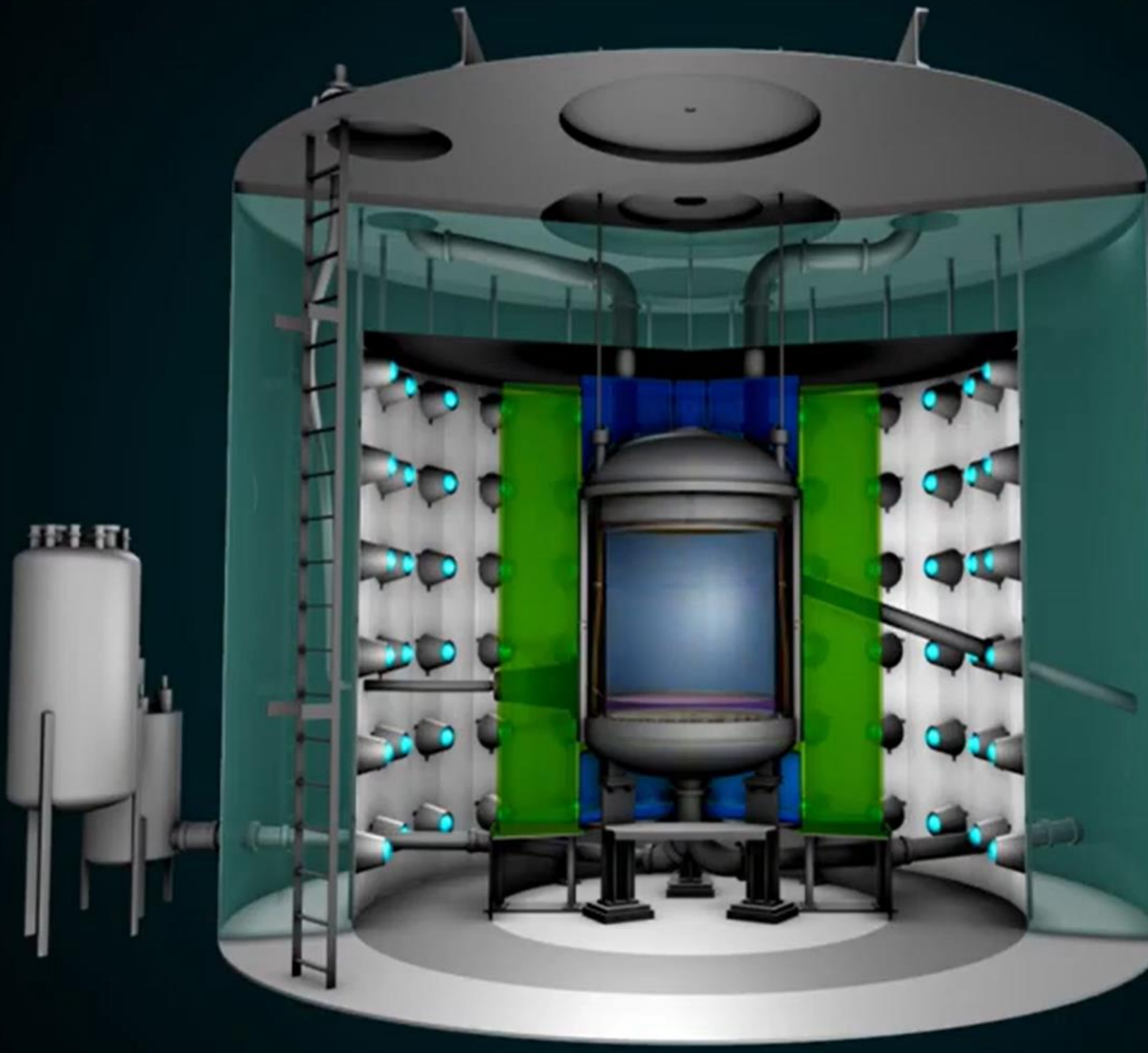
EXPERIMENTS AT SURF

- LUX and LZ are in Lead, SD
- Roughly 1 mile underground at the Sanford Underground Research Facility (SURF)
- Site of the Homestake gold mine, then the Homestake neutrino experiment (first to detect solar neutrinos)



@lzdarkmatter
https://lz.lbl.gov/

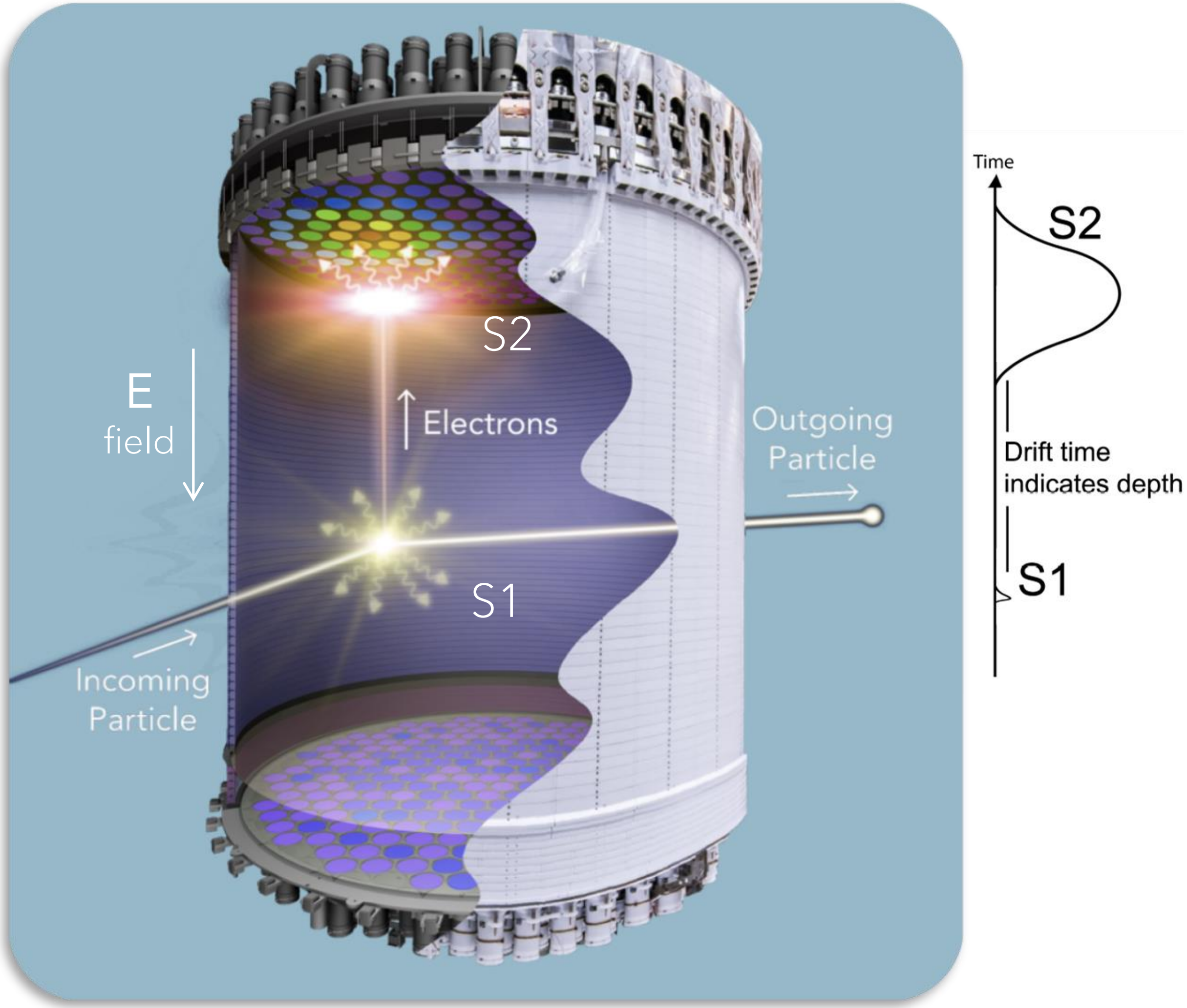
LUX AND LZ



LUX-ZEPLIN
(LZ)

DATA IN LUX AND LZ

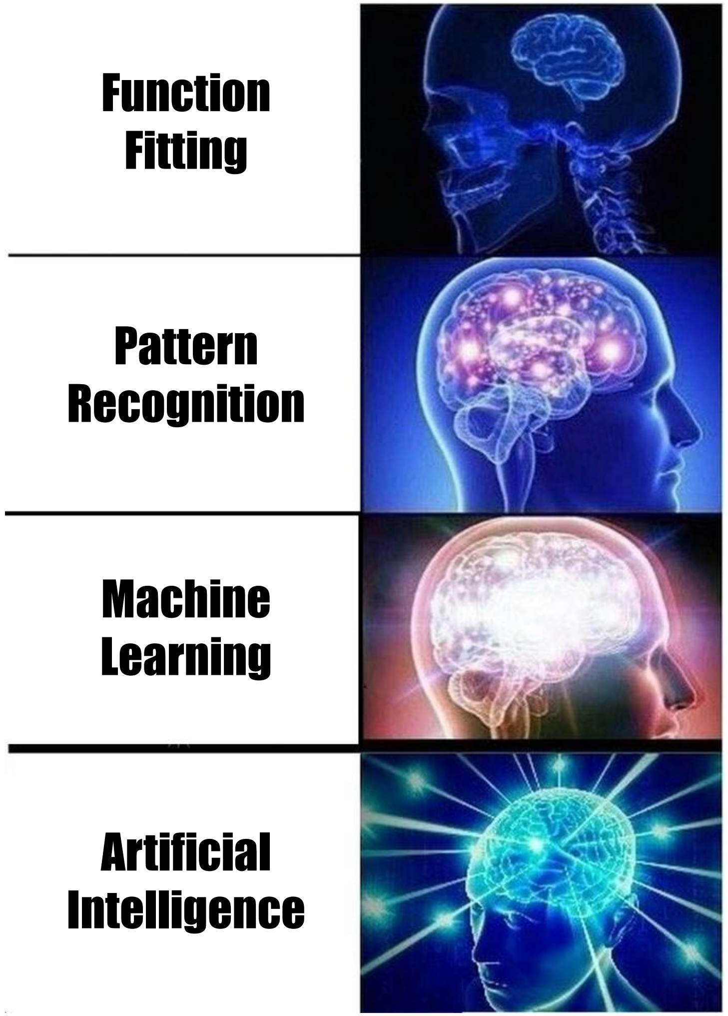
- Raw data: waveform per PMT
- Typical reconstructed info (for each scatter):
  - S1 (prompt scintillation) total area
  - S2 (ionization signal) total area
  - X, Y position (from S2 PMT hit pattern)
  - Z (from Δt between S1 and S2)
- Weighted sum of S1, S2 gives E
- S1/S2 ratio implies recoil type
  - NR is signal-like
  - ER is background-like



TRAINING YOUR MACHINE

This could be you!

WHAT IS MACHINE LEARNING?



Applied statistics
(with varying levels of PR)

WHAT IS MACHINE LEARNING?

Supervised

Inputs + labels → Model → New labels

New inputs ↑ Model

Classification

- Cat or not?

Regression

- Expected house price

Unsupervised

Inputs → Model

Dimensionality Reduction

- Personality traits (psych)

Clustering

- Netflix reccom.

Reinforcement

Model → Action → World Input → Model

- Teach drones to fly
- Master Go

WHAT IS ML GOOD FOR?

- When is it (most) useful?
  - Information-rich contexts (high dimensionality/many variables)
  - Complex or hard-to-model relationships between variables
  - Computationally-expensive problems
- Can use for more than just improved classification, *i.e.* cuts to remove backgrounds
  - Speed up computation (*e.g.* costly sims requirements)
  - Save manpower (*e.g.* avoid hand-tuning non-ML algorithms to find all edge cases)
  - Supplement traditional methods/extend simplified physical pictures by teaching you what information is valuable or not (NOT just a black box)

ML PARADIGMS

Shallow learning

- Around for decades
- Common algorithms:
  BDTs, NNs, SVMs
- Inputs: a few high-level “engineered” features (e.g. S1 and S2 areas, positions)
- Tunable parameters: Tens to hundreds
- Training: often manageable on a laptop in a few seconds to a few hrs

Deep learning

- Use has ballooned since ~2010
- Common algorithms: typically some flavor of NN (e.g. CNN, RNN)
- Inputs: many raw features (e.g. 2D image of PMT hit pattern; 1D time-series waveform)
- Tunable parameters: Thousands to millions
- Training: <1 hr to many days or longer on dedicated GPU nodes (e.g. Google research)
- Architecture must find clever ways to make training feasible given # of params
  (e.g. regularization, weight-sharing)
- Higher complexity -> potentially more sensitive to quirks in training dataset – if MC, have to trust more

SUPERVISED LEARNING FLOWCHART

- How to be a good teacher:
- **Most important:** provide maximum info (choose inputs wisely)
- Semi-important: choose algorithm intelligently
- Less important: "hyperparameters" (architecture)
- Training set: often most of the work!
- Validations and systematics: afterward – very problem-dependent

Define the problem
(labeled examples, objective function)

↓

Choose algorithms/hyperparameters

↓

Train!

↓

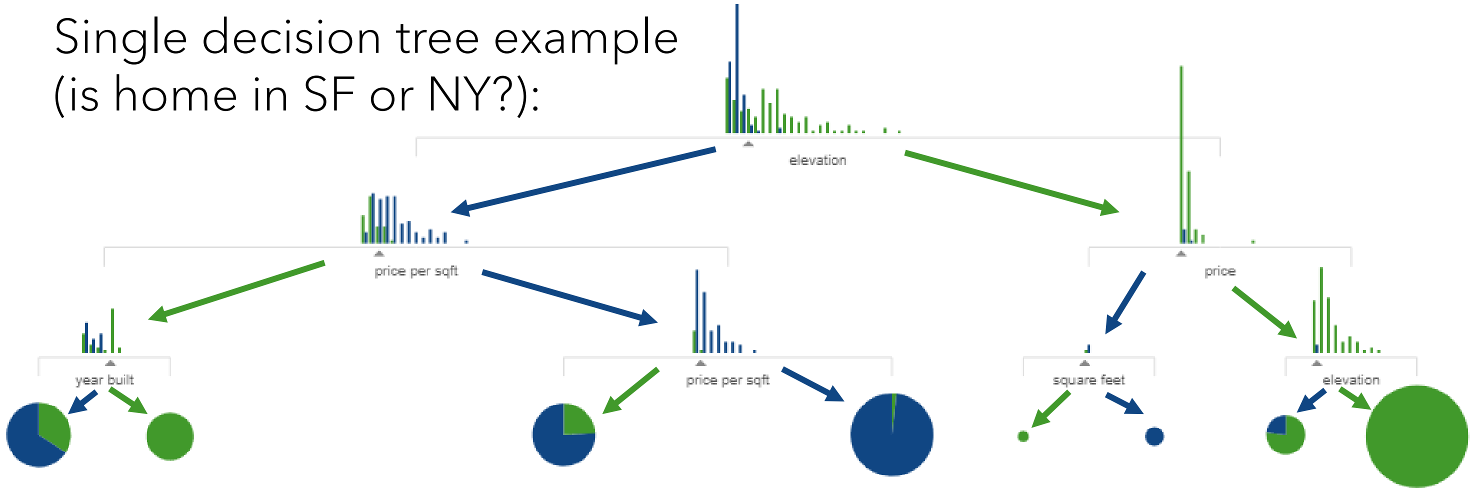
Measure performance, analyze

↓

Apply trained algorithm to data

OVERVIEW OF A FEW POPULAR ALGORITHMS: BDT

Single decision tree example
(is home in SF or NY?):

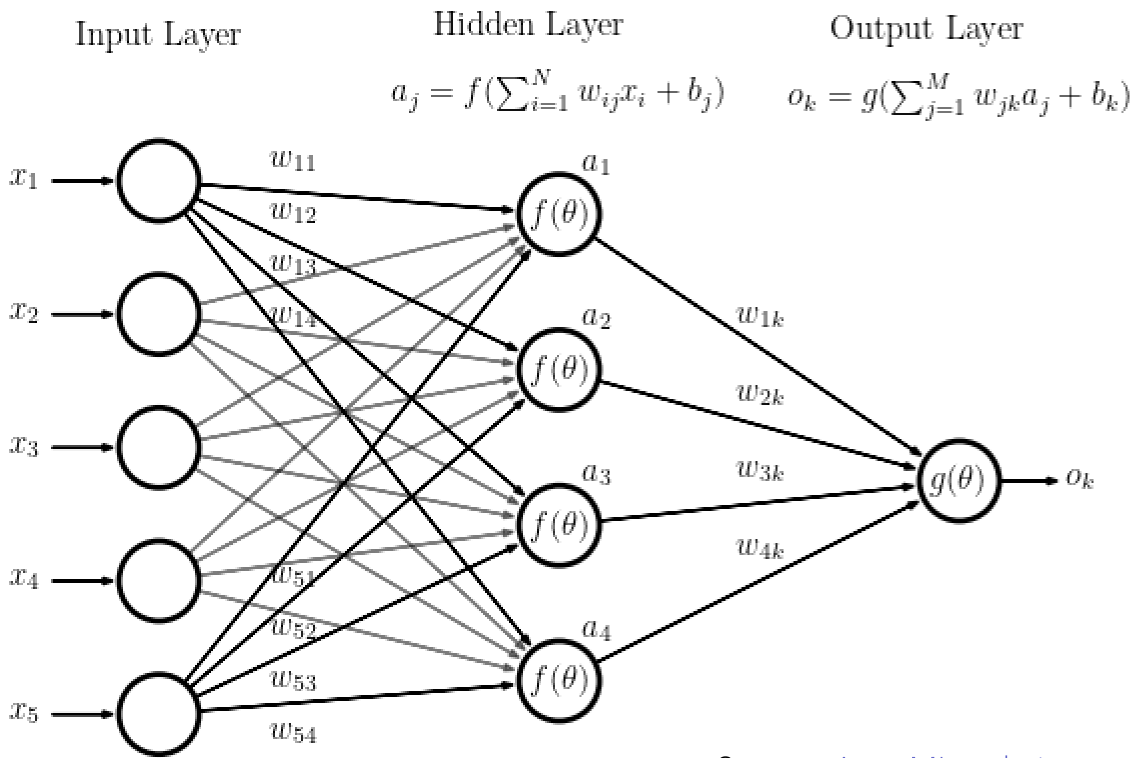


Source: [A Visual Introduction to Machine Learning]

- BDT (**boosted decision tree**) is a weighted sum of many decision trees
- Nice visualization of how a BDT cut looks [here]

OVERVIEW OF A FEW POPULAR ALGORITHMS: NN

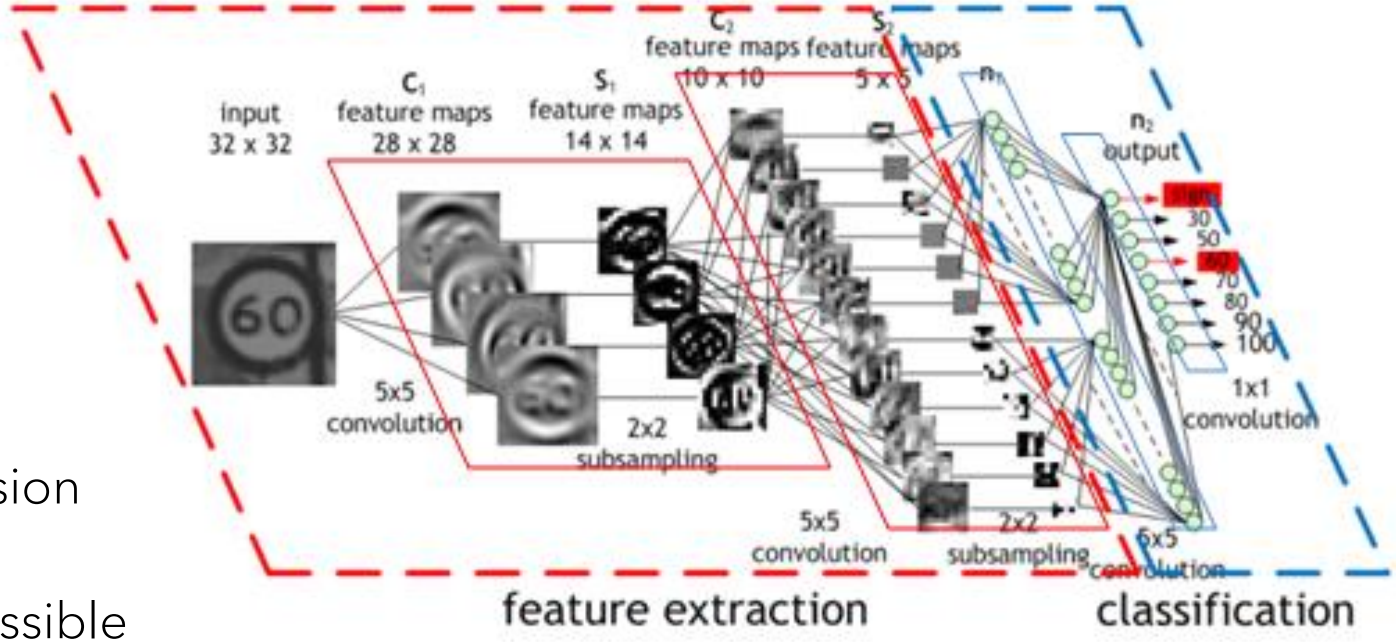
- Fully-connected **neural network** (NN)
  aka multi-layer perceptron (MLP)
  aka artificial neural network (ANN)
- Actually the most general case:
  - Other fancier NN algorithms find ways to simplify/speed up training by reducing the effective number of params (weights)



Source: [AstroML website]

OVERVIEW OF A FEW POPULAR ALGORITHMS: CNN

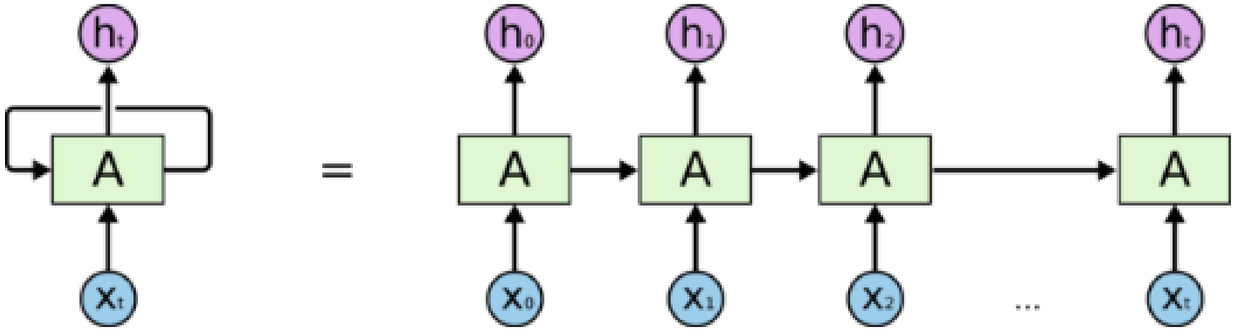
- Convolutional neural network
  - Assumes translational symmetry
  - Weights are learned for specific feature maps (rather than each node)
  - Good for processing **images** on a regular square grid
  - Typically 2D but can also do 1D version (e.g. for time series)
  - 3D can be tricky to optimize, but possible
- See also: **graph neural network** (GNN) – useful for data with irregular relationships (e.g. images not on a regular grid); very flexible, can be harder to train



Source: NVIDIA

OVERVIEW OF A FEW POPULAR ALGORITHMS: RNN

- Recurrent neural network
  - Network structure is repeated multiple times in sequence
  - Each prior network gives context from the previous element
  - Useful for sequences of unknown length
  - Commonly used for natural language processing (e.g. understanding text on the internet; completing words in a sequence)



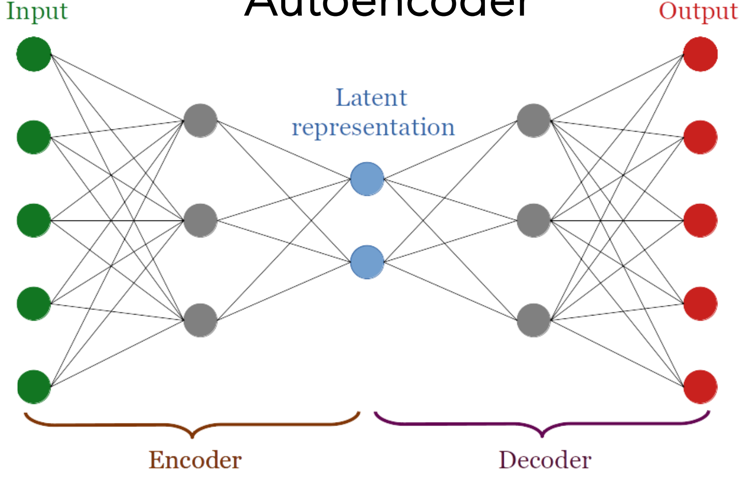
An unrolled recurrent neural network.

Source: [Christopher Olah's blog]

OVERVIEW OF A FEW POPULAR ALGORITHMS: UNSUPERVISED

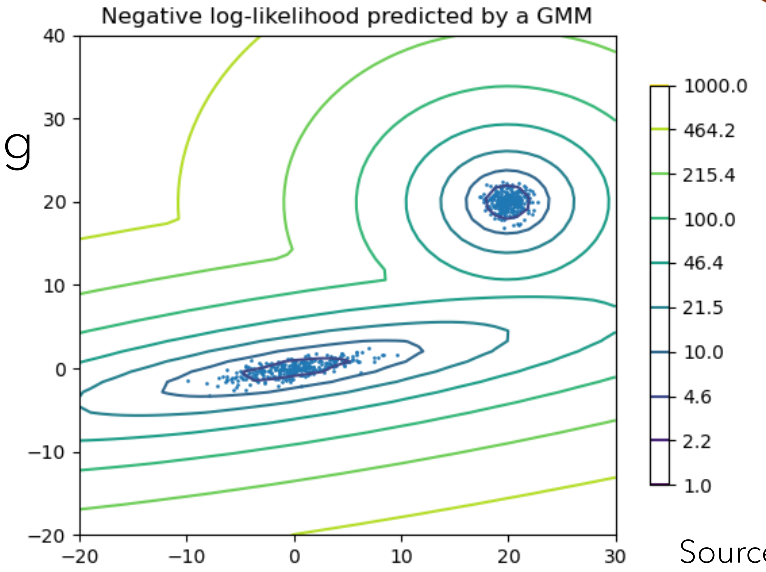
- Dimensionality reduction
  - t-Distributed Stochastic Neighbor Embedding (**t-SNE**)
  - Uniform manifold approximation and projection (**UMAP**)
  - Autoencoder (technically supervised)
- Clustering:
  - Gaussian mixture model (**GMM**)
  - Density-based spatial clustering of applications with noise (**DBSCAN**)
- Anomaly finding:
  - Autoencoder (again)
  - Isolation forest

Autoencoder



Source: Towards Data Science

GMM



Source: scikit-learn

HEURISTICS FOR NN TRAINING

- Generally, training is sped up by having more hidden layers rather than more nodes/layer
- Regularization is often necessary for good deep learning:
    - Adjust the objective function to penalize large weights
    - Add dropout layers
- ReLU or similar is a good default activation function
- Adam or similar is a good default optimizer (smart gradient descent)
- Don't sweat details of architecture too much
    - If it really matters, do hyperparameter optimization if possible

HOW CAN MACHINE LEARNING HELP US FIND DARK MATTER?

DARK MATTER + ML: A UNIQUE CHALLENGE

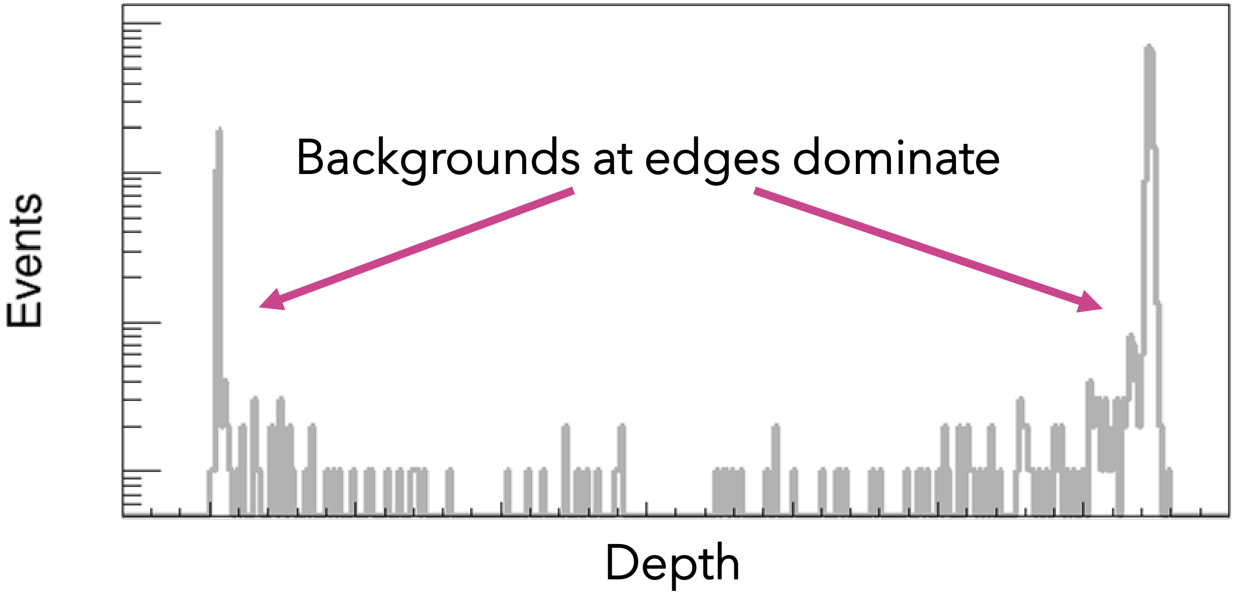
- DM analysis is esp. sensitive to mismodeling: at most a few candidate signals
- Collider physics, neutrino detection have pioneered use of ML in physics
- Value in collaboration across experiments (c.f. DANCE-ML 2020* workshop)
- ML growing in DM + Neutrinoless ββ decay†
- Examples of improvements from ML for DM:
    - Extending physics reach
    - Fast and flexible analysis
    - Better understanding of data

*https://indico.physics.lbl.gov/indico/event/DANCE_ML_2020
† *Machine Learning in the Search for New Fundamental Physics,*
G. Karagiorgi, G. Kasieczka, S.K., B. Nachman, D. Shih,
Invited review at *Nature Reviews Physics* [arXiv 2112.03769]

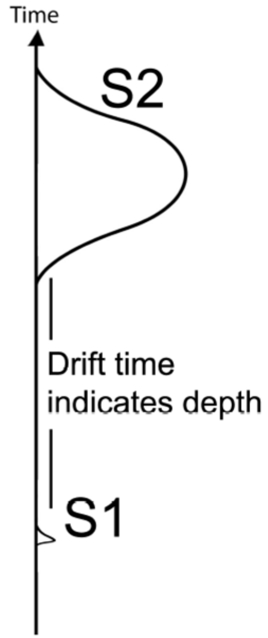
CHARGE-ONLY ANALYSIS

- Goal: extend range of DM models we can explore through lower energy threshold
- Challenge: low-energy events only have the 2nd (larger) flash of light ("S2") – depth unknown
- …but backgrounds from wires at the top and bottom are significant! Can't remove w/o depth

Joint work w/
K. C. Oliver-Mallory



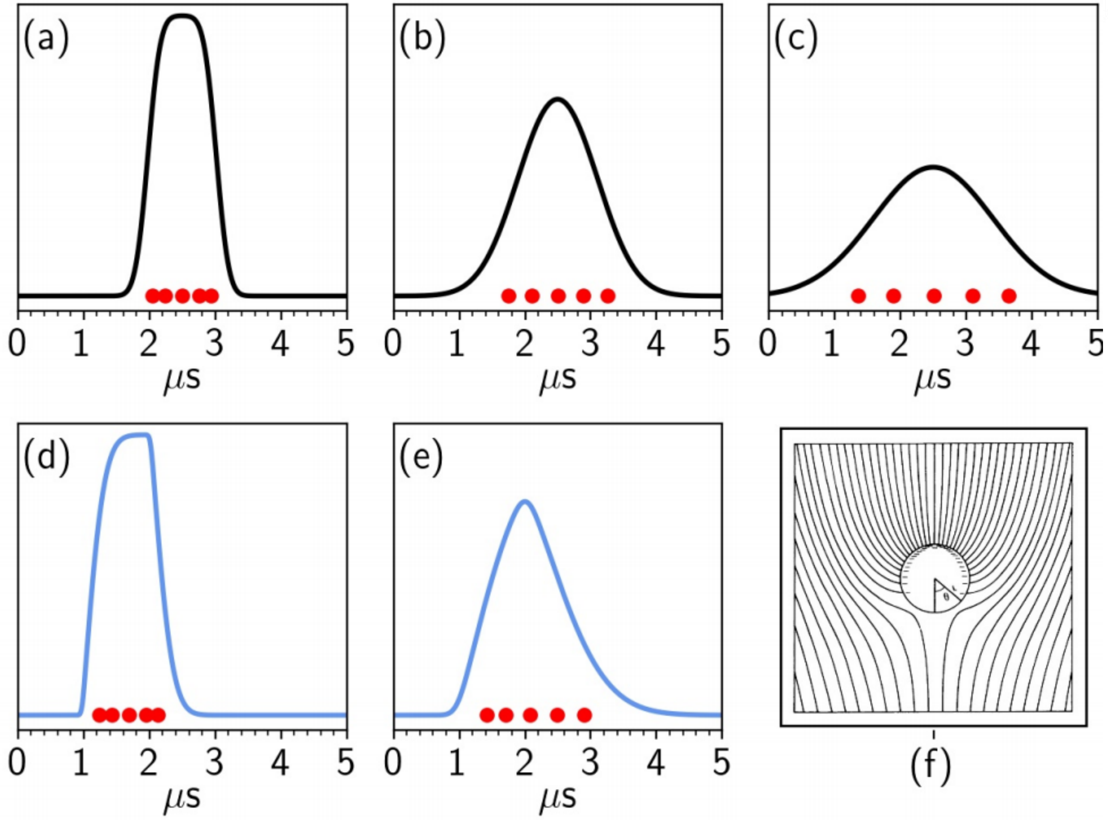
(Plot from events with S1)



Phys. Rev. D **104** 012011
[arXiv 2011.09602]

DATA-DRIVEN TRAINING

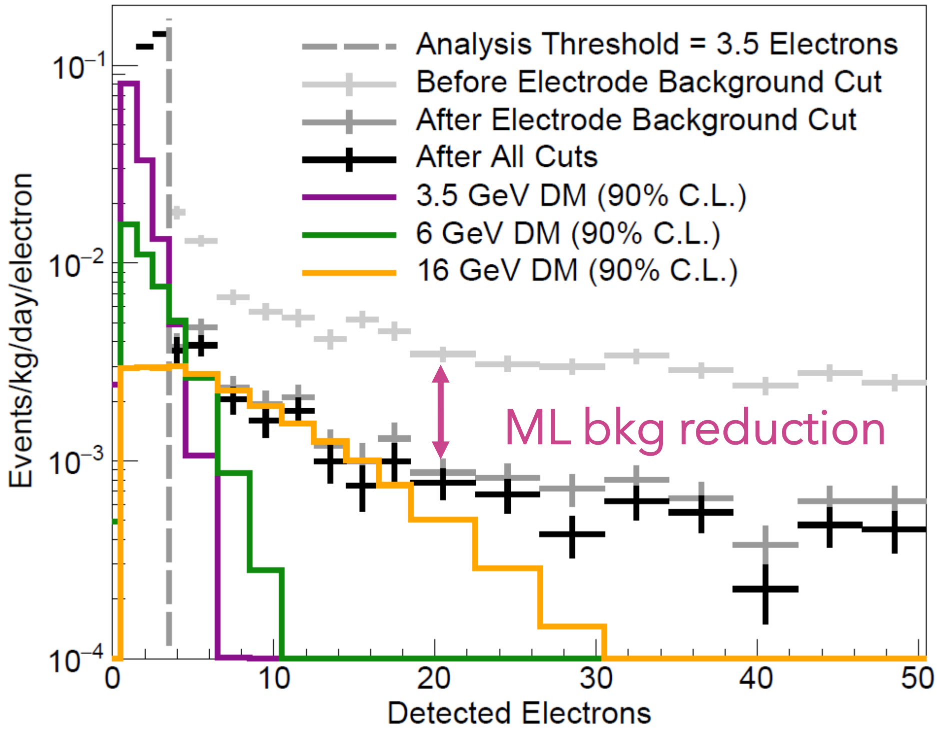
- Can we use the shape of S2 pulses to tag events from the grid wires?
- **Cannot trust simulations** of S2 pulse shapes near grid wires! Too tricky to accurately model all processes (strongly-varying field, electron diffusion, etc.)
- Solution: **use real data** from events with S1 and S2 **as a training set**; S1 gives location tag, but ML model uses only S2 shape info

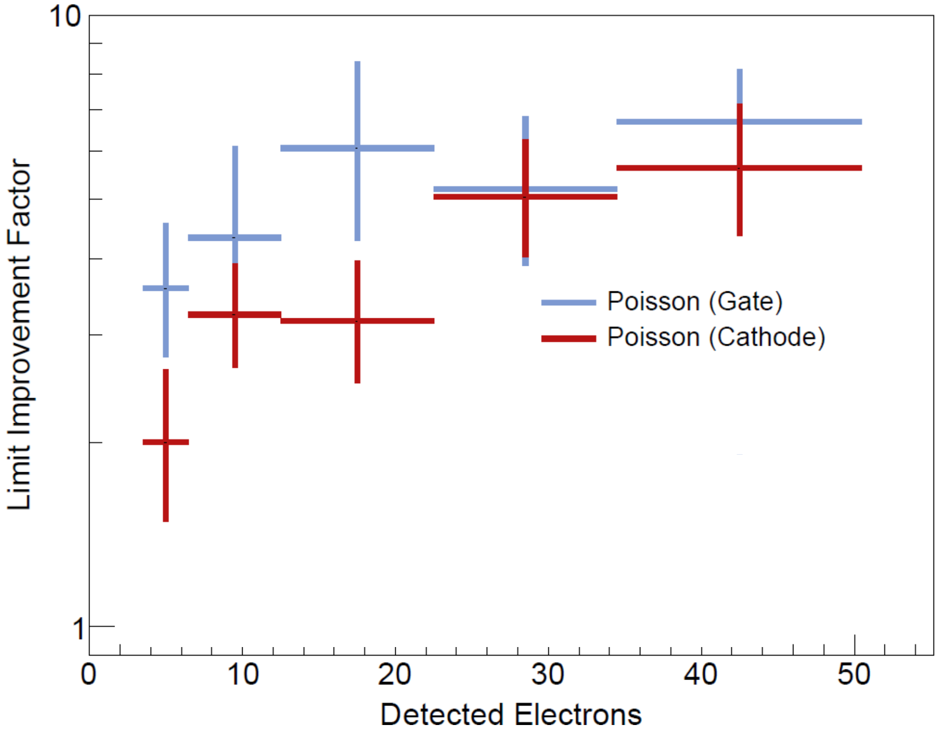


S2s from the center (a-c, black) are symmetric; S2s from the grids (d-e, blue) are stretched out due to uneven electric field close to wires.

RESULTS

- Train a boosted decision tree to distinguish grid events from bulk
- BDT cut reduces the observed event rate by ~4x while retaining ~60% signal efficiency
- No DM observed → set a limit; sensitivity of search improved by 2-7x over a simple Poisson counting analysis, depending on true ratio of backgrounds (unknown)



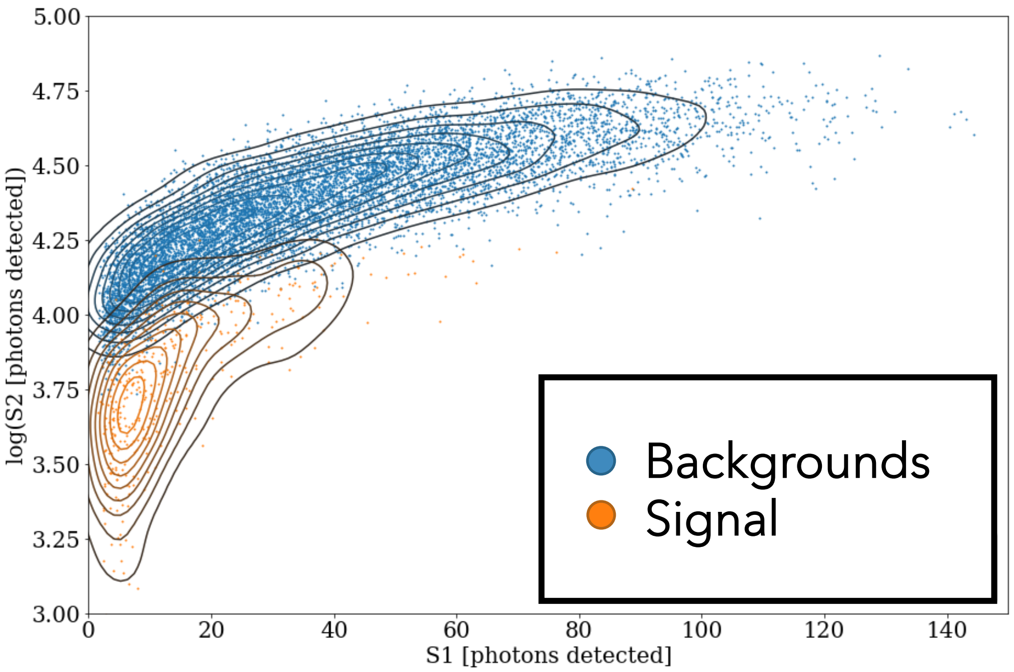


IMPROVED FITTING W/ ML

- Traditional approach:
    - Create models of backgrounds, DM signal (PDFs)
    - Use PDFs to create a likelihood function, fit model to data
- Generating PDFs, calculating limits intractable >3-4D
- Must assume independence of variables, e.g. $\{r, z, \phi\} \otimes \{S1, S2\}$
- Instead, use NN to compress all info into 1D:
    - Improved speed
    - Important correlations preserved
    - Allows additional inputs

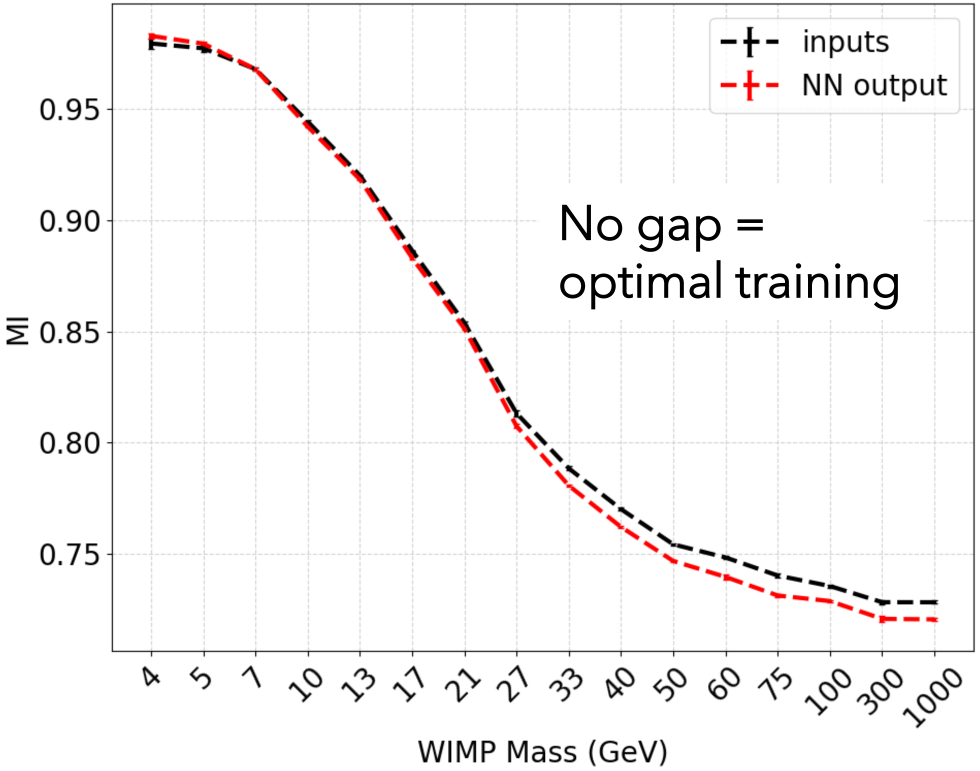
New!
arXiv 2201.05734,
Submitted to Phys. Rev. D

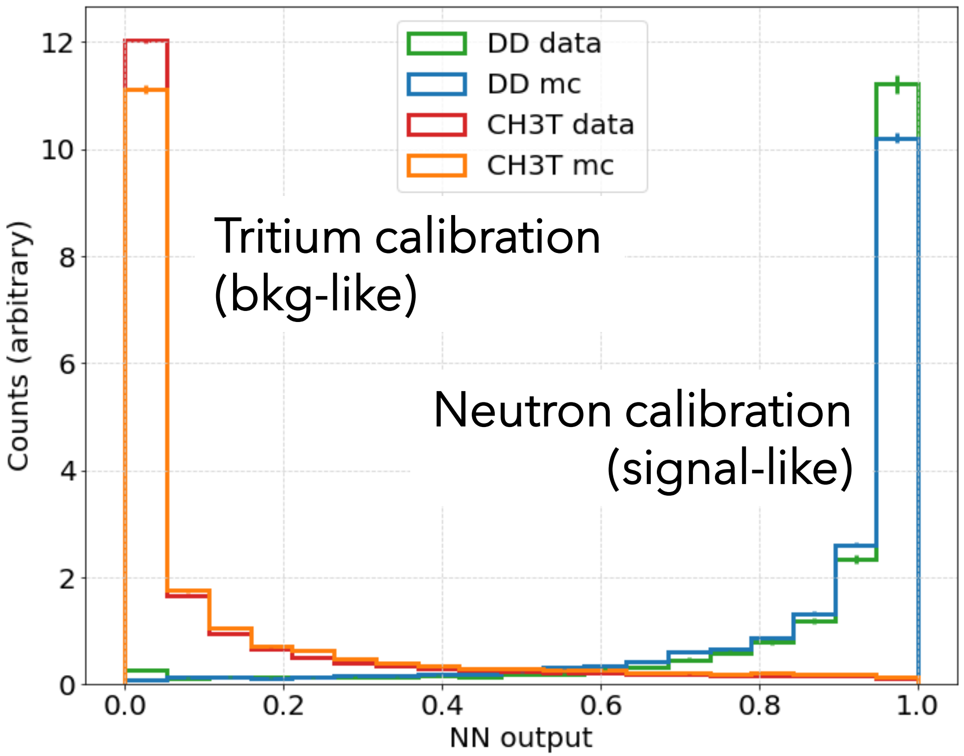
Joint work w/
N. Carrara



TRAINING PERFORMANCE

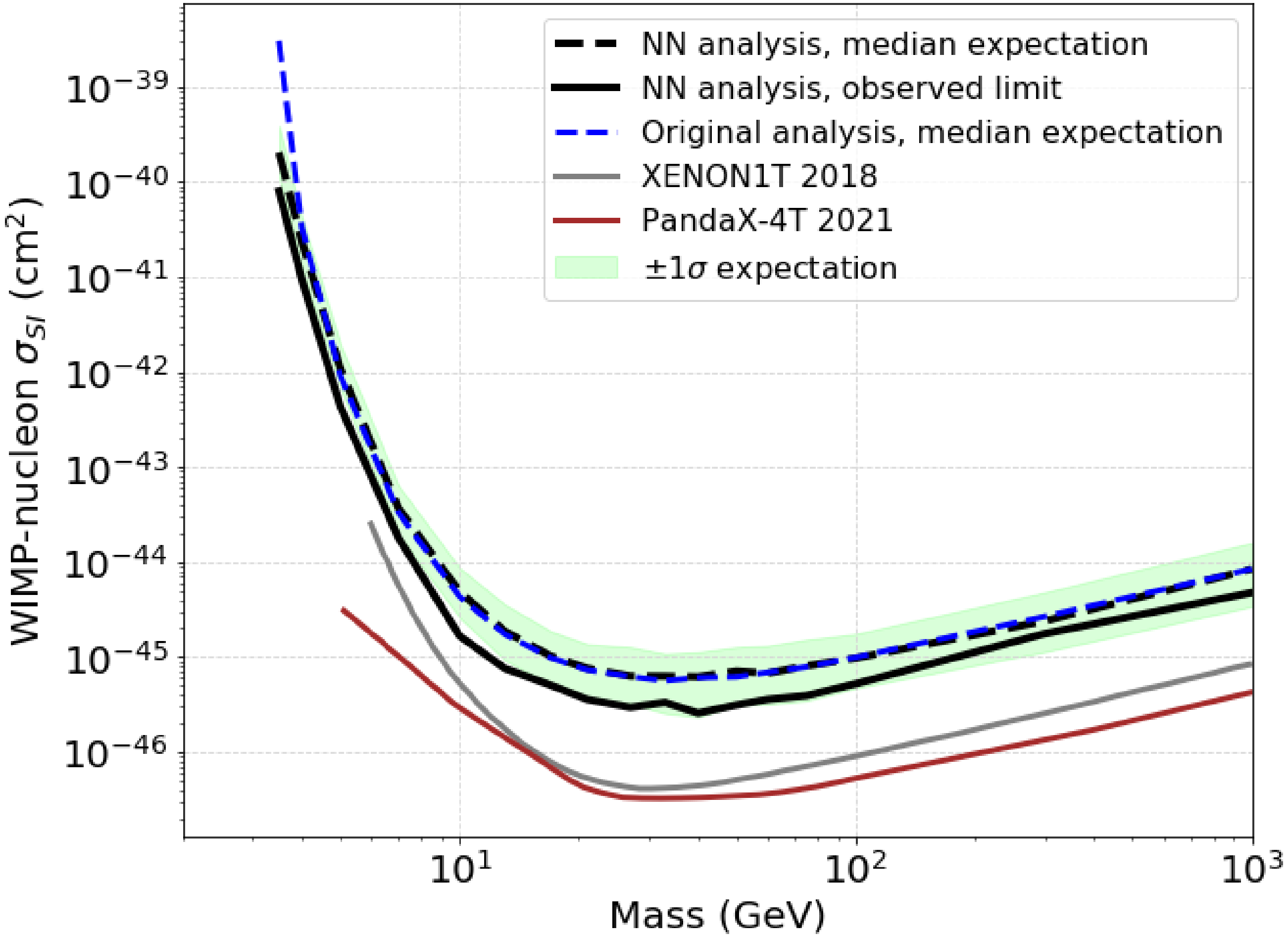
- Ensure information is preserved using mutual information (MI) on full space vs 1D output
- Confirm that Monte Carlo (MC) sims faithfully represent real data using calibration sources





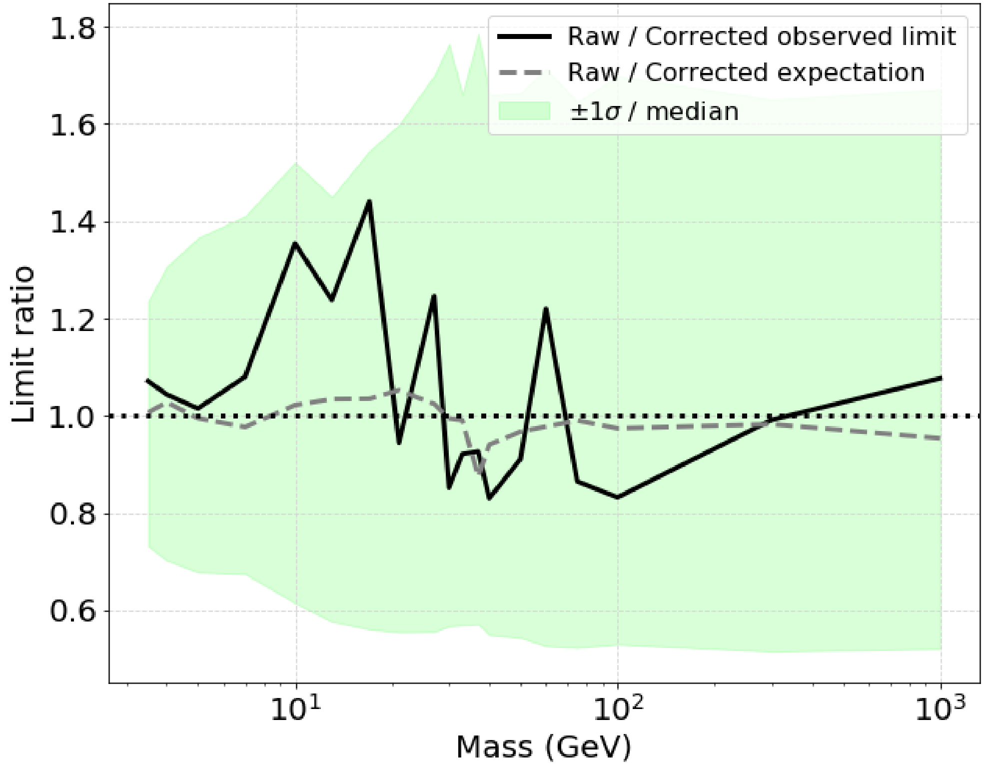
LIMIT RESULTS

- Compare to published result using same inputs {r, z, S1, S2}
- Reproduces limit almost exactly
- Limit generation runs much faster
- Dedicated test indicates **35x improvement in speed**
  - Important for exploring a **broader range of models**
  - e.g. EFT searches w/ 15 operators x 24 different masses = 360 hypothesis tests
  - **Enables more complex analysis**



FLEXIBILITY IN ANALYSIS

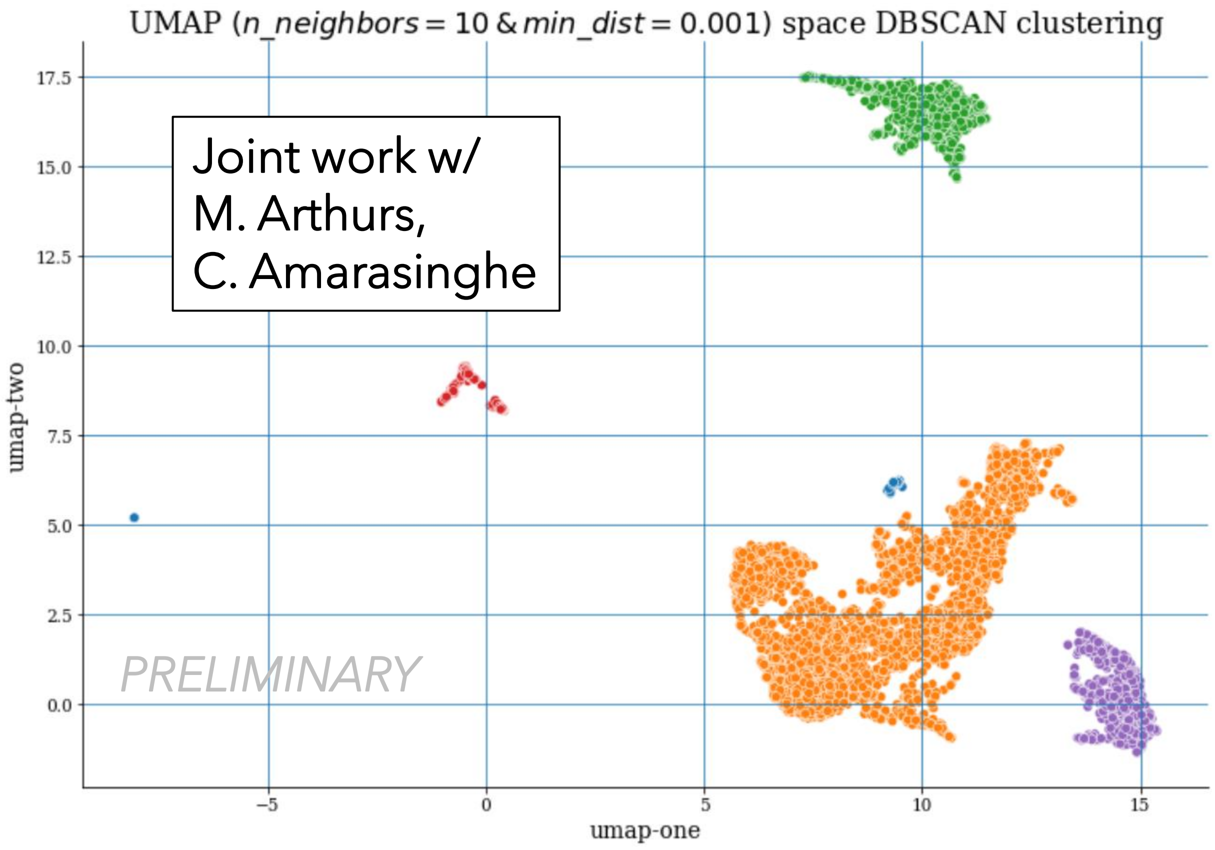
- Relevant correlations are captured and utilized:
  - Equal limits established when using {r, z, $S1_{raw}$, $S2_{raw}$}
- Scales well with more inputs:
  - S1 pulse shape variable easily added
  - No significant penalty in analysis (CPU) or coding (human) time



Analysis	Workspace creation (hr)	MC generation (hr)	Hypothesis testing (hr)	NN training (hr)	**Total (hr)**
Original EFT search	2.7	39.0	8.8	–	**50.5**
NN case	2.4e-3	1.0e-2	0.81	0.64	**1.46**
NN speedup	1100×	3900×	11×	–	**35×**

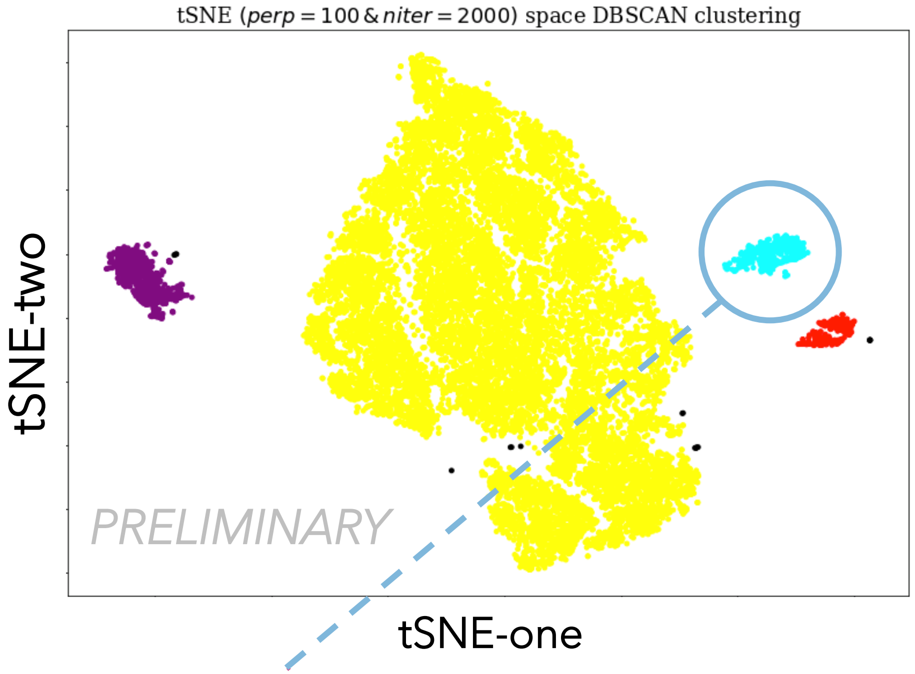
ANOMALY FINDING IN LZ

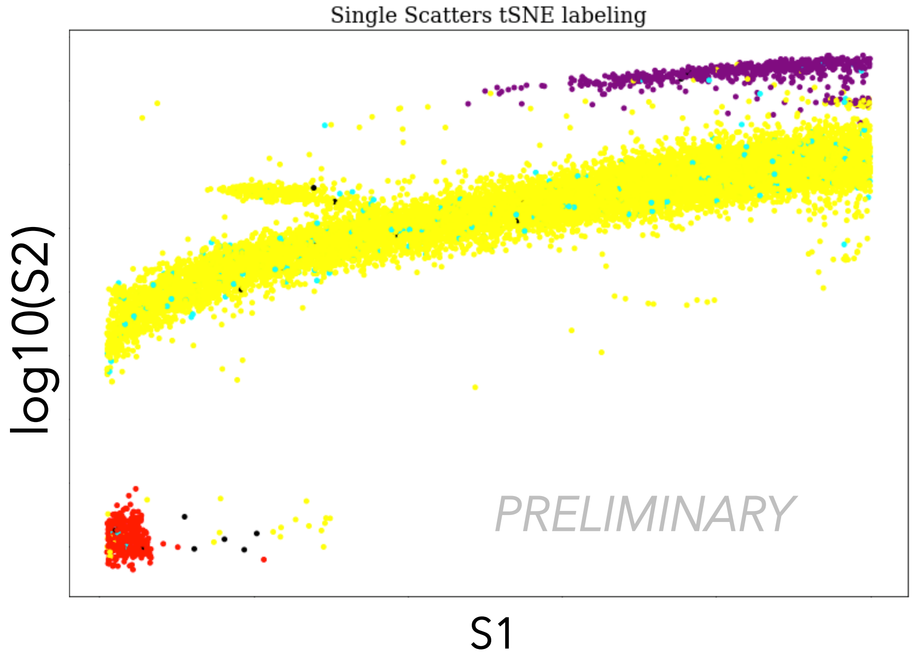
- Early stages of LZ focused on understanding detector
- Before sims are calibrated to match data → unsupervised learning ideal
- Use dimensionality reduction (UMAP) + clustering (DBSCAN):
  - Quickly ID problematic populations
  - Assist in understanding physical origins and removal techniques

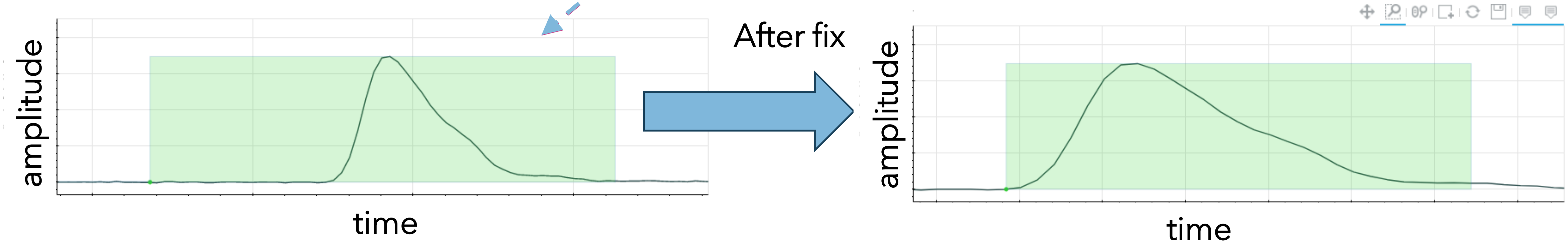


ANOMALIES: RECONSTRUCTION

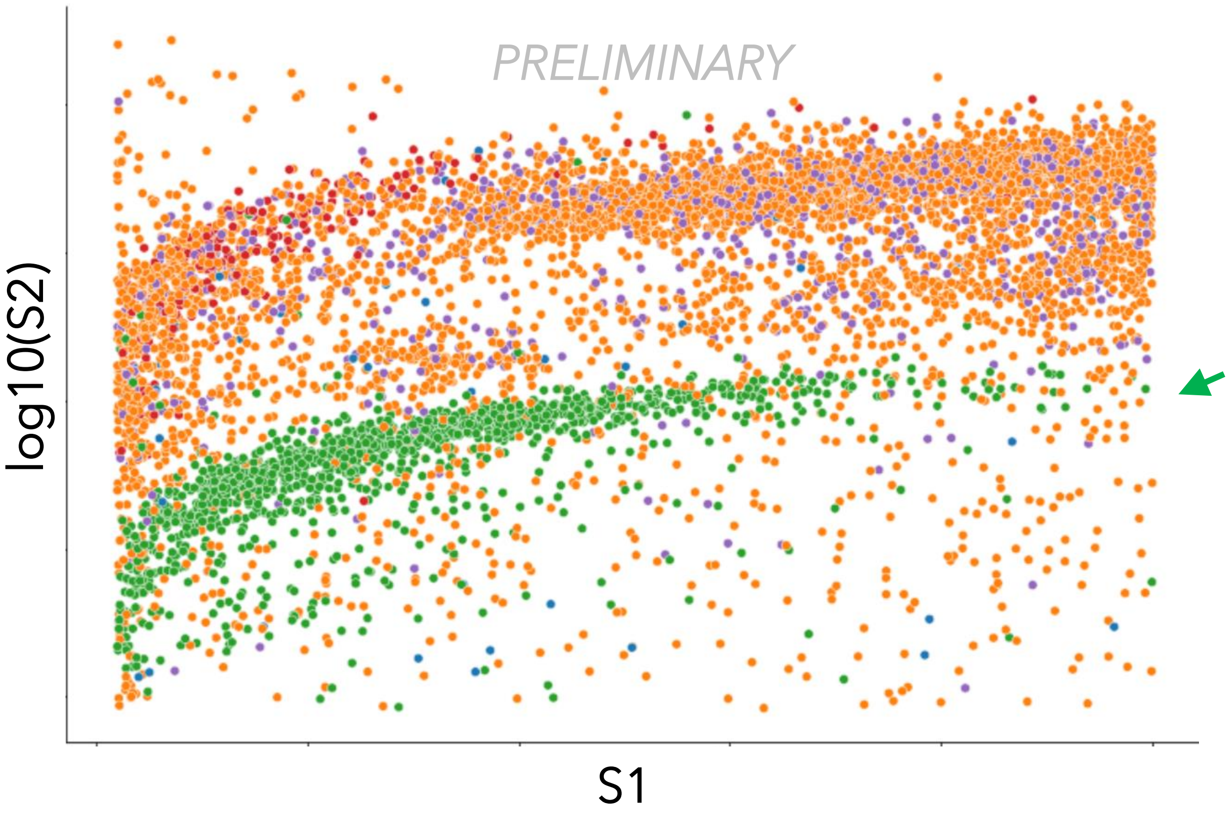
- Simulated data
- Fix long rise time in reconstruction

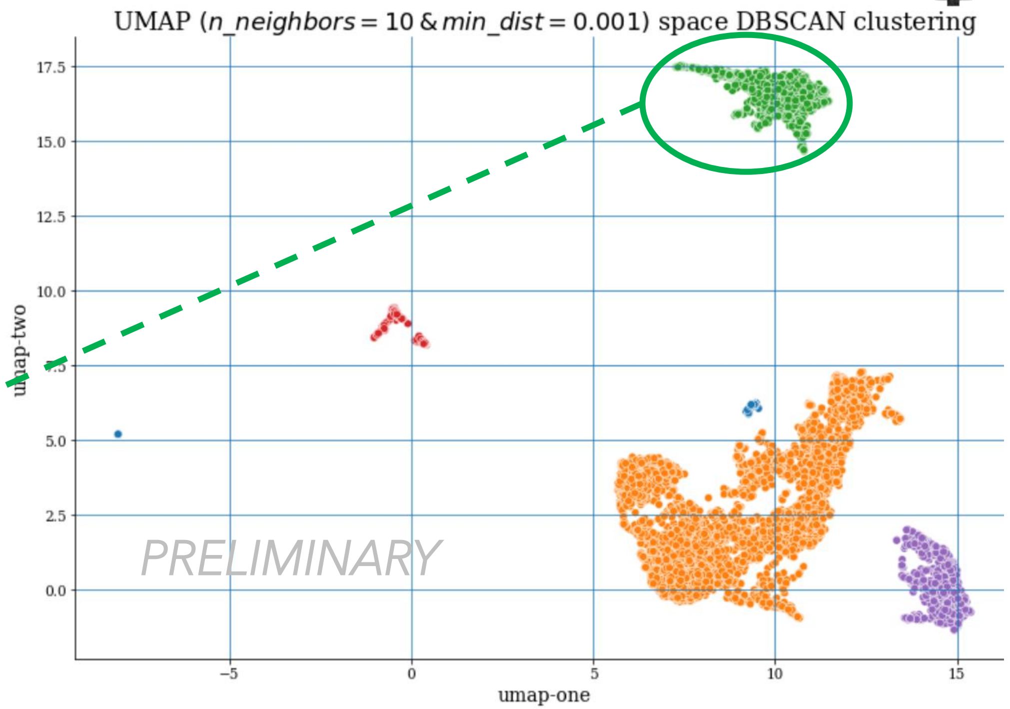






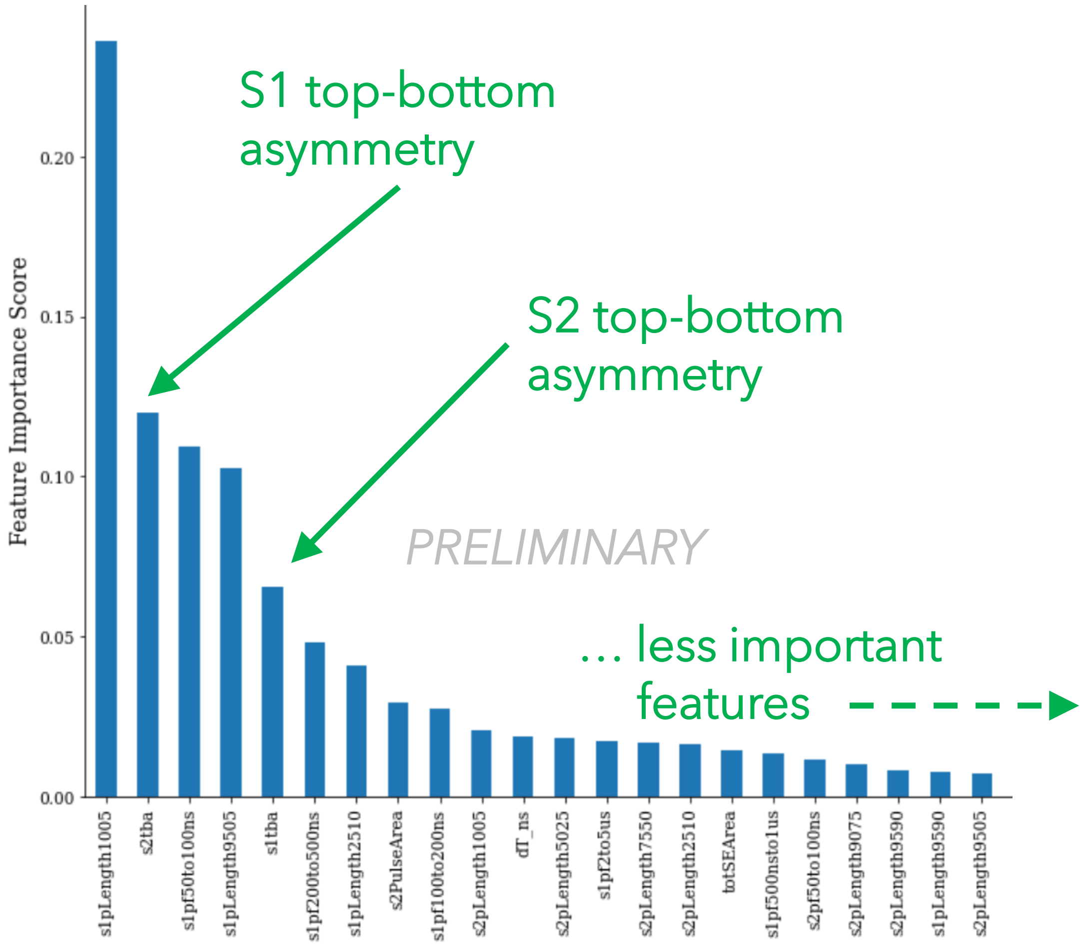
ANOMALIES: DETECTOR EFFECTS

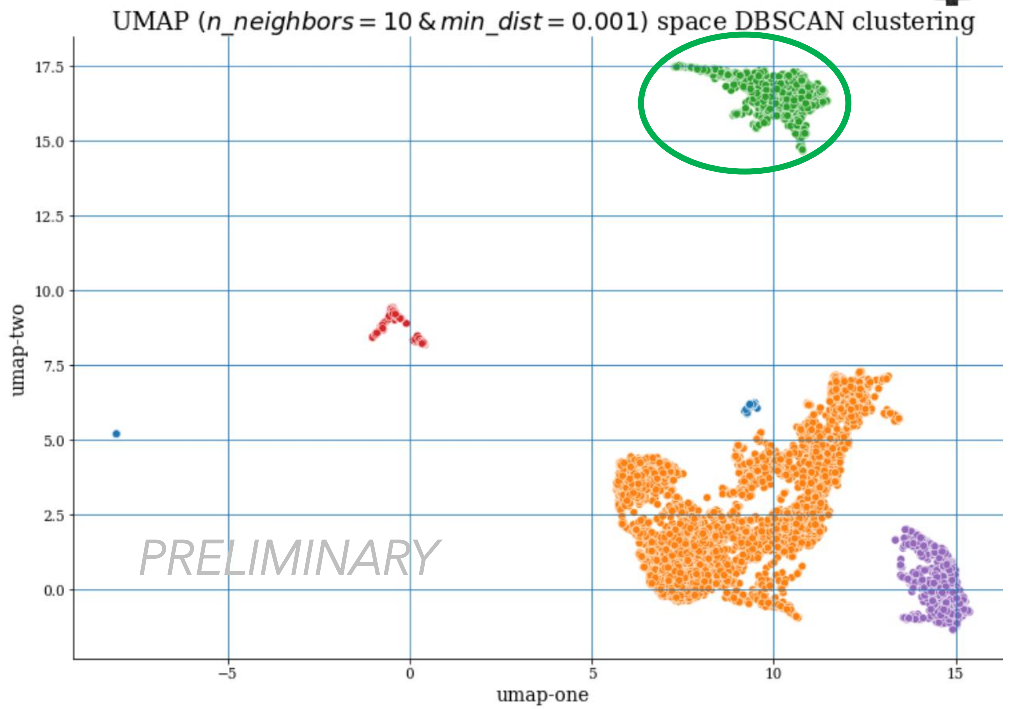




- Real commissioning data
- Abnormally-small S2s
- Normal-looking pulses

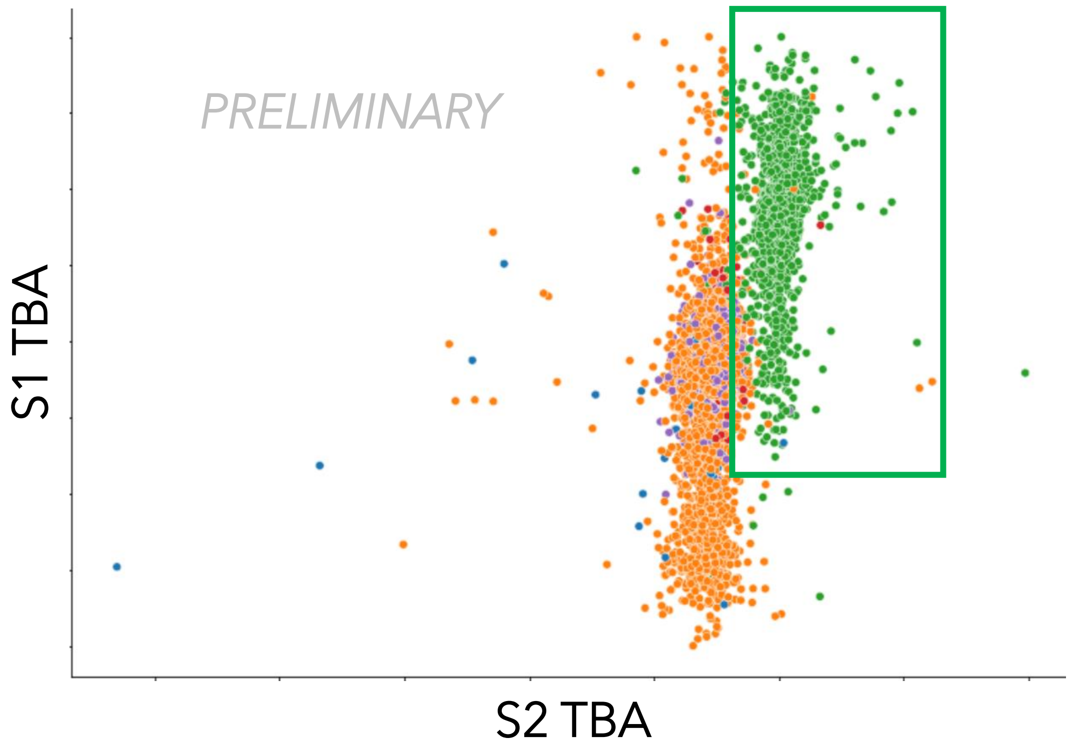
ANOMALIES: DETECTOR EFFECTS

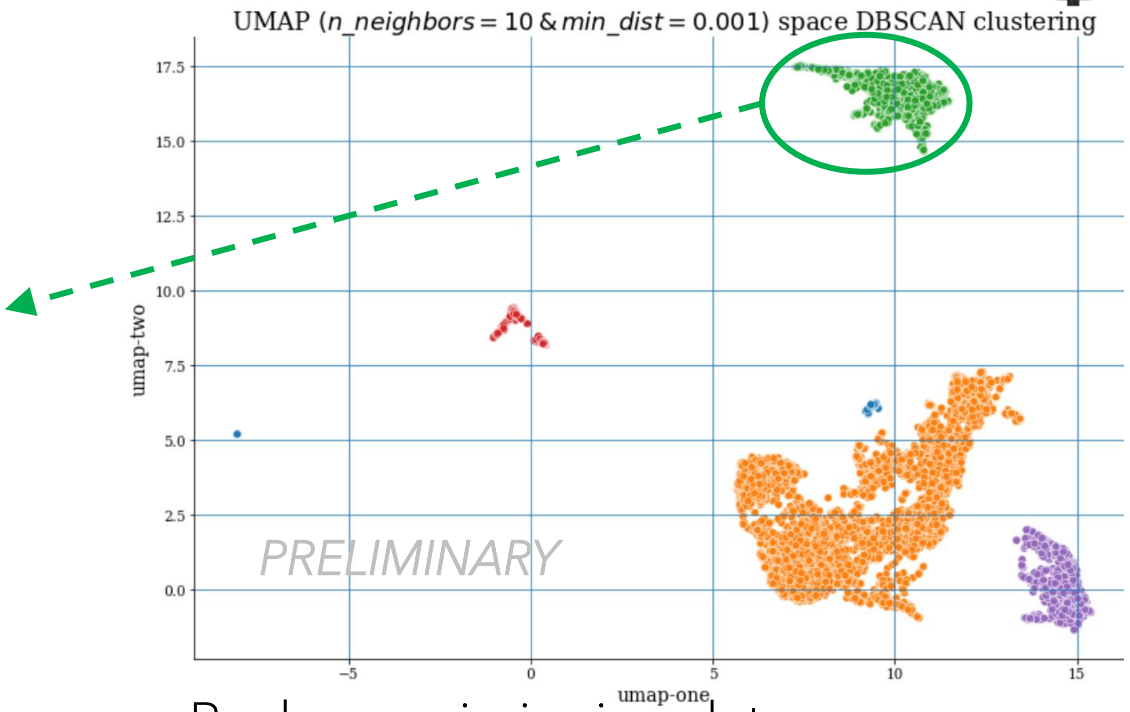




- Real commissioning data
- Ratio of top-to-bottom PMT arrays is flagged as important for this cluster

ANOMALIES: DETECTOR EFFECTS

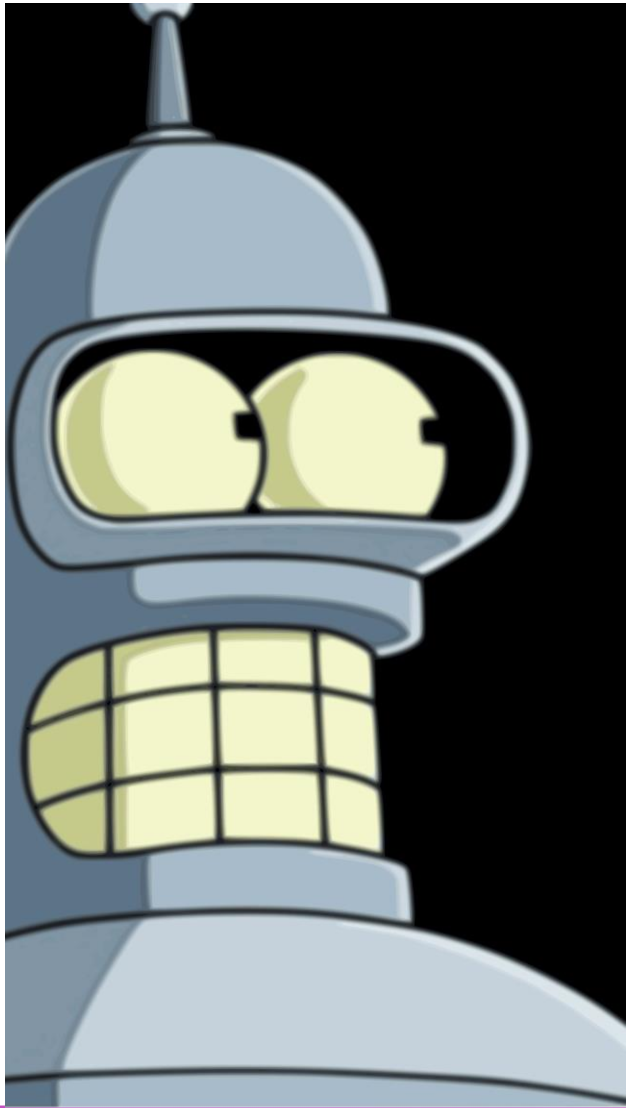




- Real commissioning data
- Cut in S1 TBA vs S2 TBA efficiently removes
- Helped identify physical origin: above-anode gas events

THE FUTURE OF ML FOR PARTICLE PHYSICS

- Physics-inspired architectures:
  - Enforcing known symmetries in the network, e.g. energy conservation (energy flow networks)
  - Interpretability: reverse-engineering deep learning insights
- Build your own network (LEGO for ML):
  - Combine layers (e.g. combine CNN w/ high-level inputs)
  - Add multiple learners (e.g. pivoting)
  - Custom objective functions (e.g. to account for systematics)
- Techniques for moving away from simulation dependence:
  - Pivoting (sims only; reduce reliance on uncertain quantities)
  - Domain adversarial training (part sims, part data)
  - Training using impure or unlabeled data from calibrations (fully data-driven)



ML RESOURCES

- Hitchhiker's Guide to ML for physicists (list of annotated links: tutorials, blogs, courses)
- Status of ML research in particle physics
  - Living review of particle physics (lots of links to papers, attempts to stay updated)
  - Recent review of ML in particle physics: *Machine Learning in the Search for New Fundamental Physics* [arXiv link] – see section on rare event searches
- DANCE-ML 2020 workshop on ML in DM and neutrino physics
  - Includes tutorials and presentations
  - My general-purpose tutorial (Jupyter notebook, covers most major steps of a ML analysis)
  - Indico page here
- Local group for ML work in particle physics at LBL (must be affiliated w/ LBL)
  - Slack link, mailing list, webpage