Using Deep Learning to Explore Daya Bay Data

Sam Kohn
Physics 290E Seminar
19 October 2016

Neutrino oscillations

Result of mismatch between mass and flavor eigenstates

$$|\nu_\alpha\rangle = \sum_i U^*_{\alpha i} |\nu_i\rangle$$

Mixing angles determine amplitude of oscillation

Δm^2 determines oscillation period in L/E space

matter effect & δCP

$$U = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_{23} & s_{23} \\ 0 & -s_{23} & c_{23} \end{bmatrix} \begin{bmatrix} c_{13} & 0 & s_{13}e^{-i\delta} \\ 0 & 1 & 0 \\ -s_{13}e^{i\delta} & 0 & c_{13} \end{bmatrix} \begin{bmatrix} c_{12} & s_{12} & 0 \\ -s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

PMNS matrix structure ($s_{ij} = \sin\theta_{ij}$, etc.) [1]

$$P_{\alpha\to\beta} = |\langle \nu_\beta(t)|\nu_\alpha\rangle|^2 = \left|\sum_i U^*_{\alpha i} U_{\beta i} e^{-im_i^2 L/2E}\right|^2$$

Calculation of oscillation/survival probability

$$\begin{aligned} P_{\text{sur}} =& 1 - \cos^4\theta_{13}\sin^2 2\theta_{12}\sin^2\Delta_{21} \\ & - \sin^2 2\theta_{13}(\cos^2\theta_{12}\sin^2\Delta_{31} + \sin^2\theta_{12}\sin^2\Delta_{32}) \end{aligned}$$

electron (anti)neutrino survival probability [2]

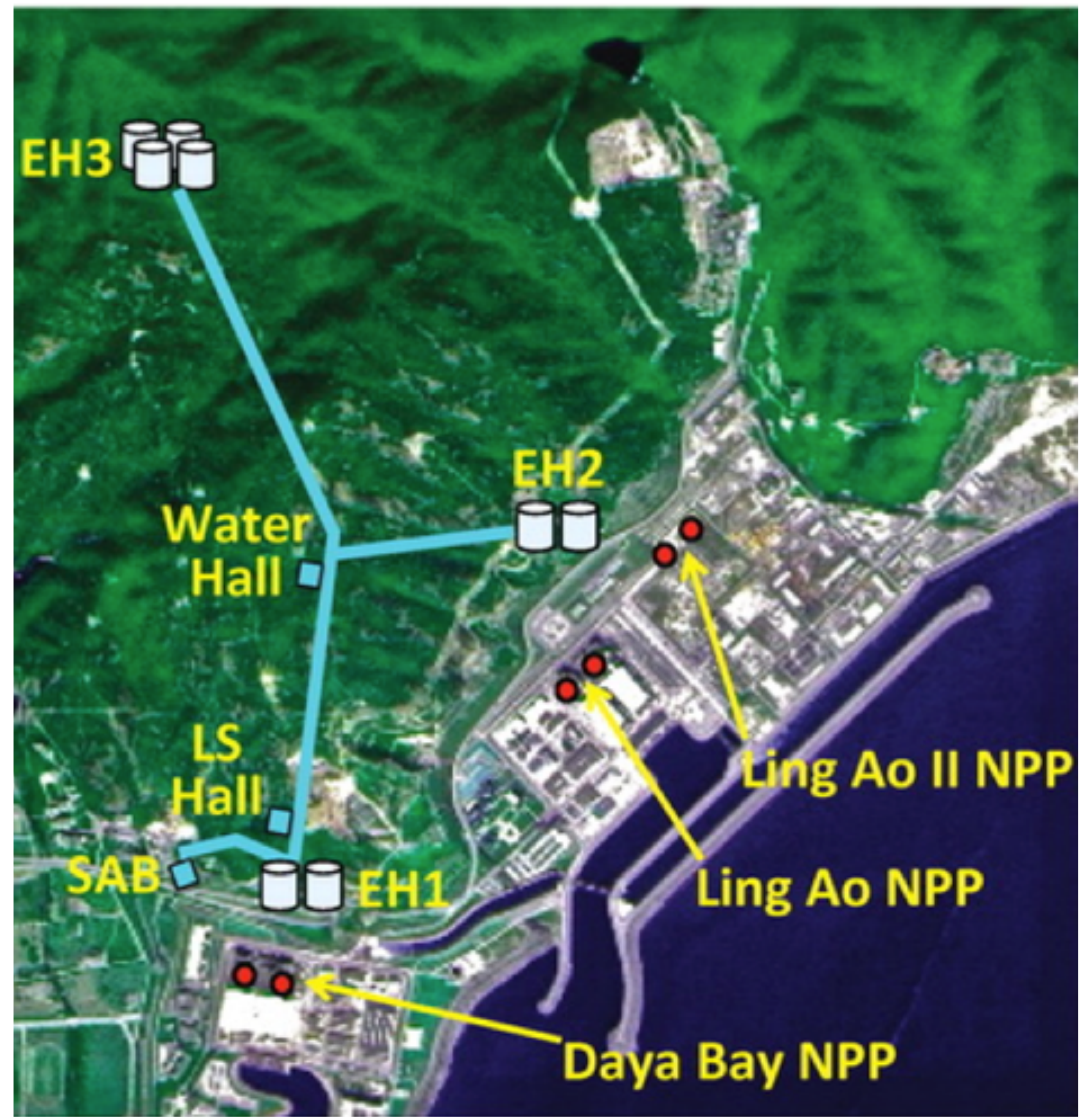
Daya Bay Experiment

Discovery and precision measurement of nonzero θ_{13}

Reactor antineutrinos

- Large, isotropic flux
- Well-understood spectrum
- "Free"

Note: Daya Bay deals only with electron antineutrinos, but I will still just use "ν" for simplicity



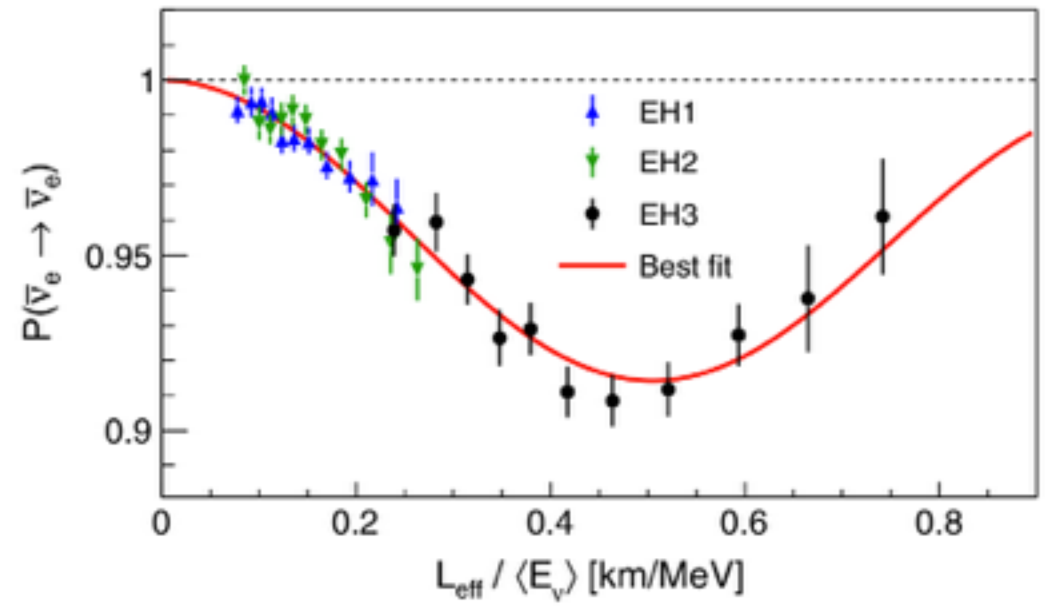
Results (spoiler!)

First nonzero measurement of θ_{13} in 2012, now at $\sin^2 2\theta_{13} = 0.084 \pm 0.005$ [2]

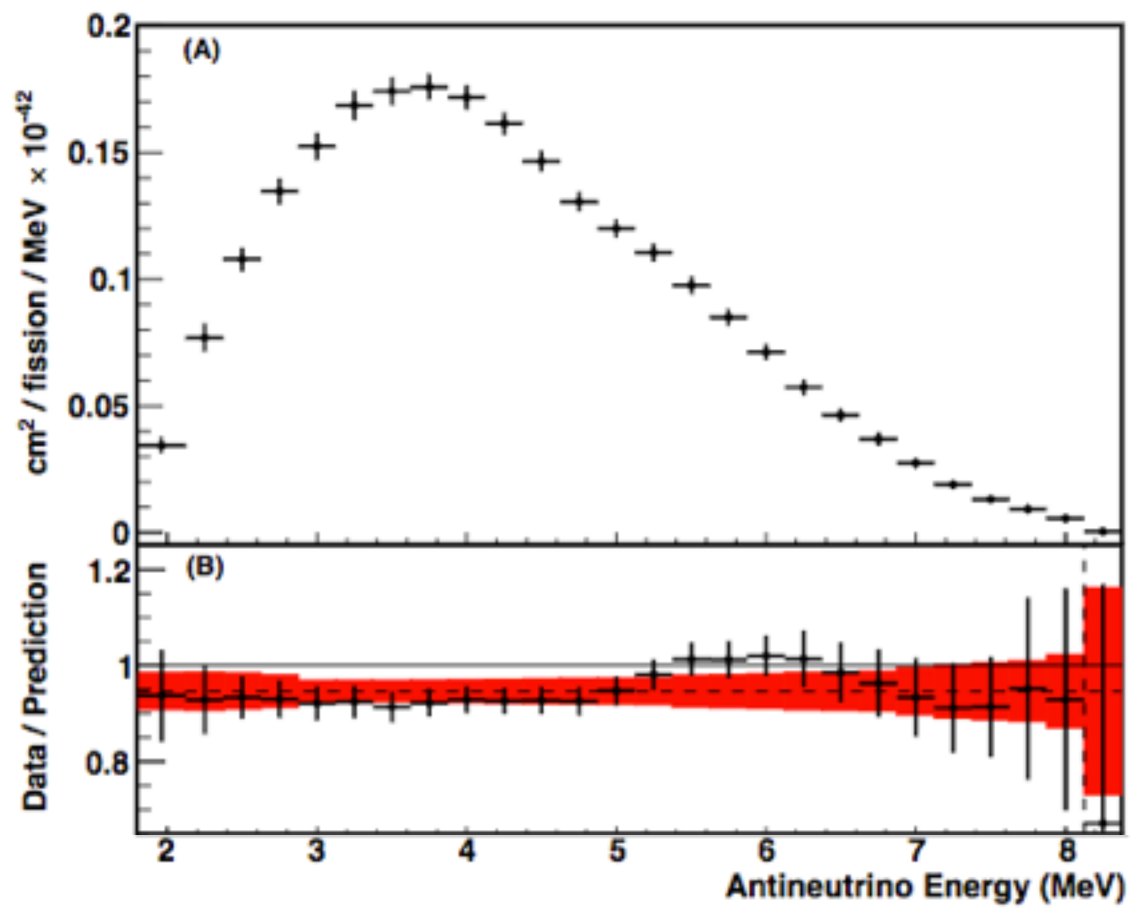
Measurement of $\Delta m^2_{ee/13/23}$

Measure reactor ν spectrum

Sterile neutrino search



L/E oscillation curve for 2015 measurement [2]



Reactor antineutrino absolute spectrum
Note deviations between model and data [3]

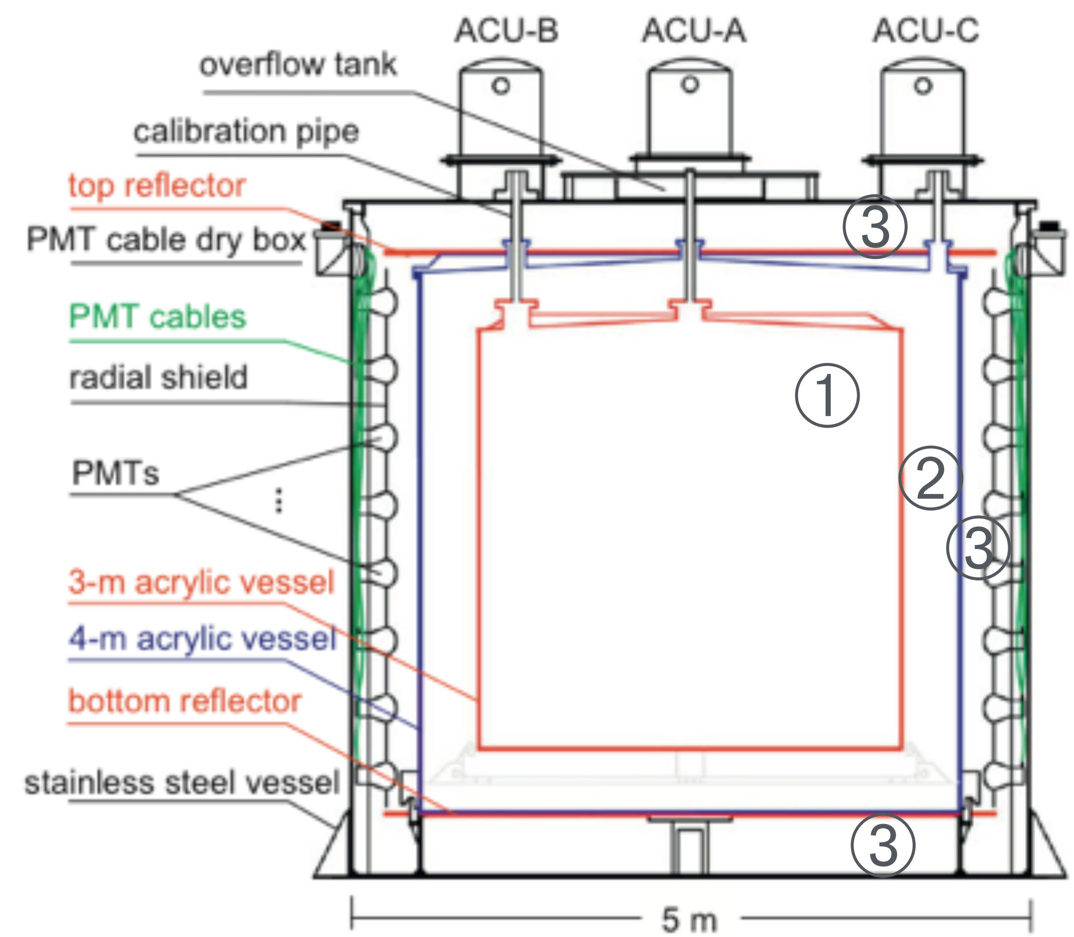
Detectors

8 identically-designed antineutrino detectors (ADs)

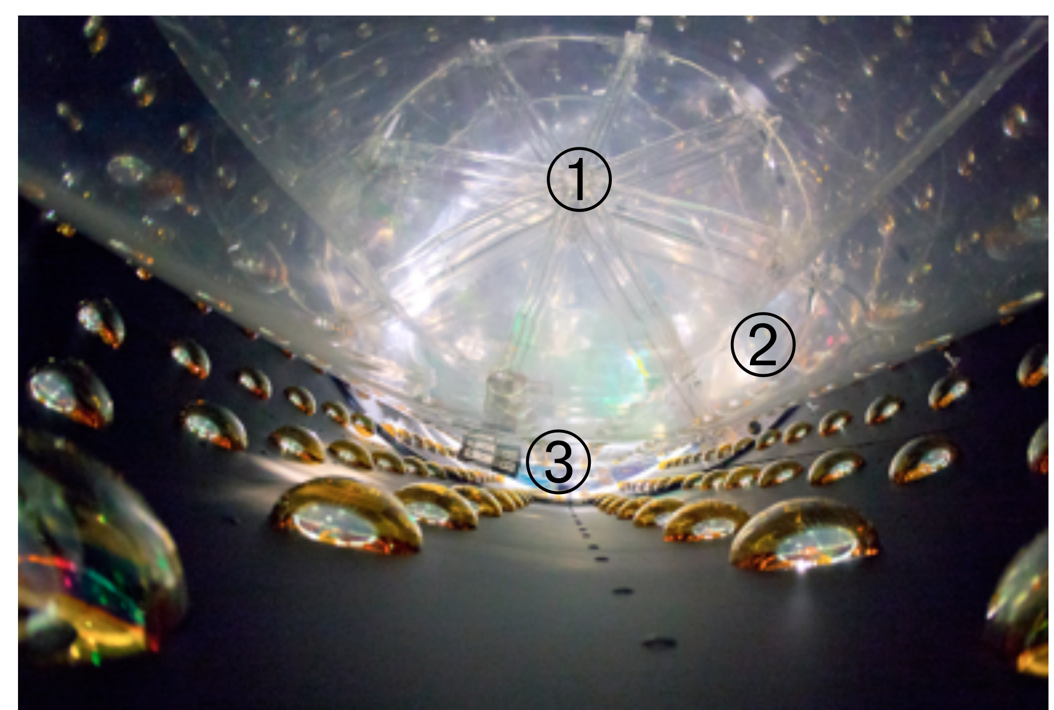
① Gd-doped LS target (LAB + bis-MSB + PPO)

② LS and ③ mineral oil in concentric layers

Water pools for shielding and muon veto (not shown in figures here)



Daya Bay AD schematic [4] and photograph [1]



Event types

① *inverse β decay (IBD)*

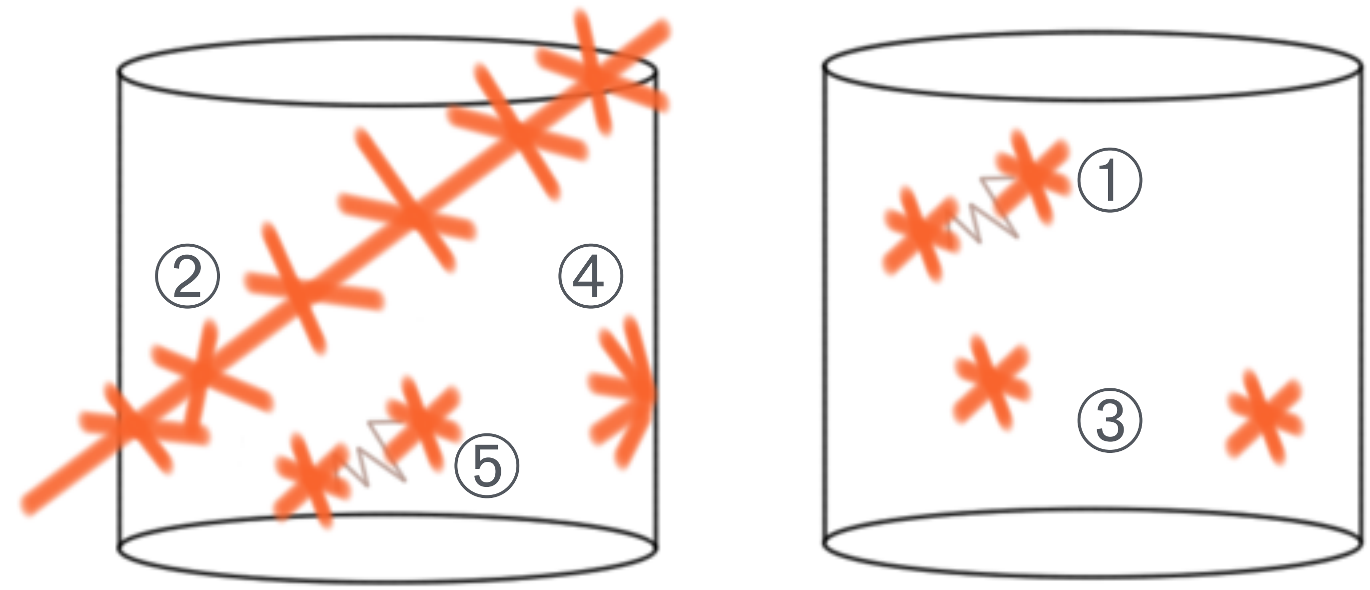
② muon

③ *uncorrelated/accidental*

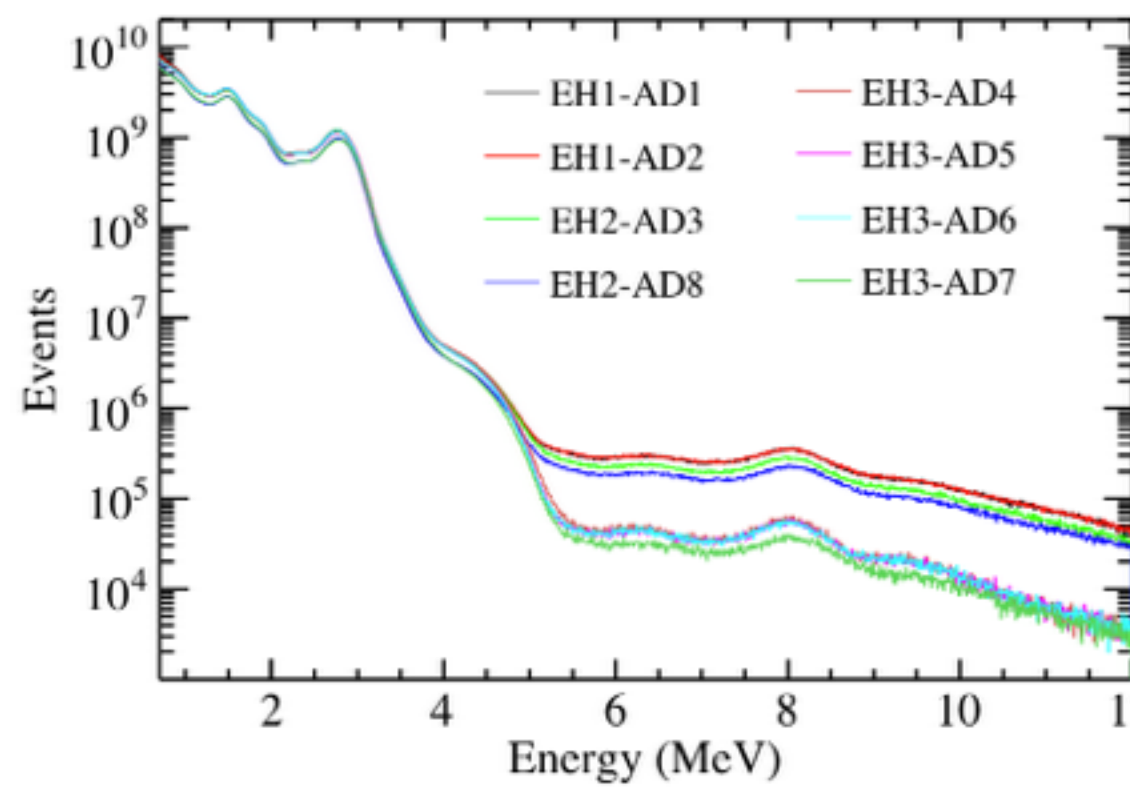
④ flashers

⑤ *^{9}Li β-n decay*

Events in *italics* are hard to distinguish from each other



Artist's (my) depiction of AD events



Measured spectrum of single AD flashes, a.k.a. half an accidental event [5]

ν selection

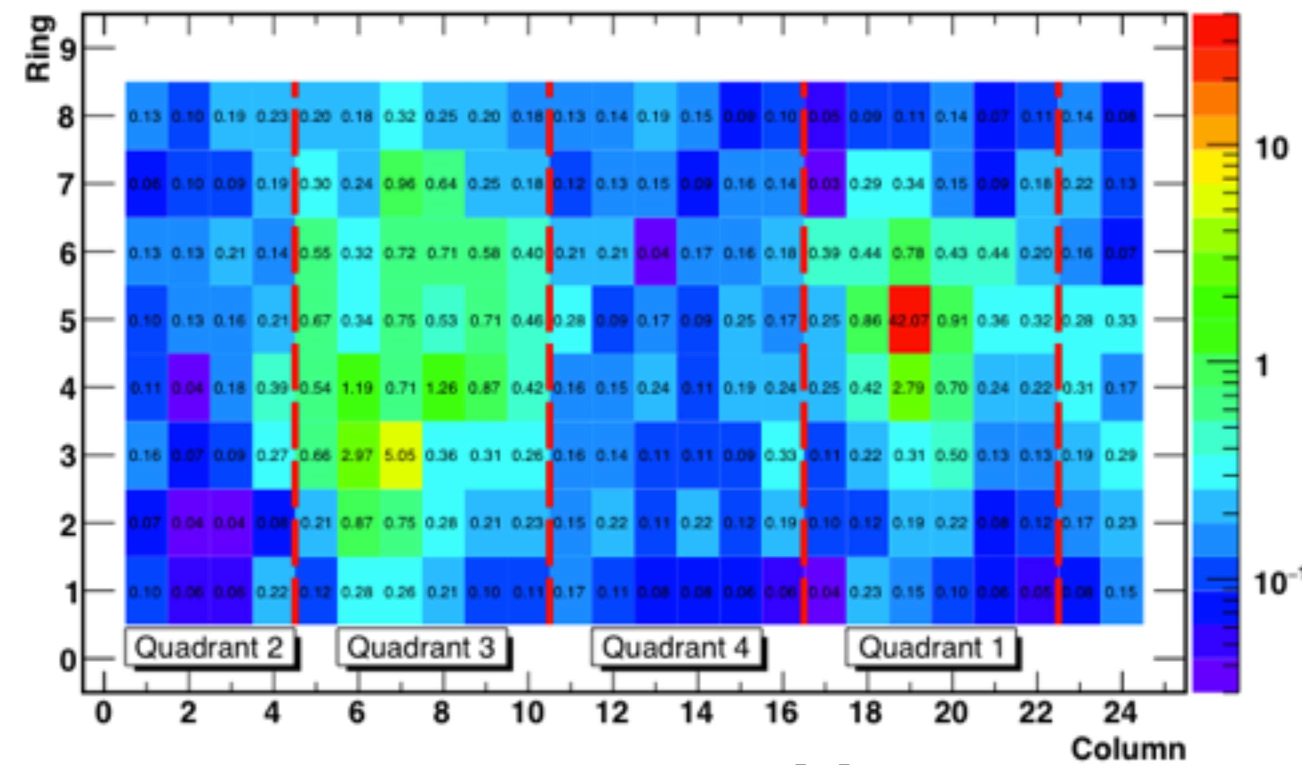
flasher cut

muon vetoes (rejects muons and ^{9}Li)

Δt for pair, $\tau_{neutron}$ ~ 30 μs (rejects accidentals)

prompt and delayed energy (rejects accidentals)

purity: ~98% IBDs



Anatomy of a flasher event [5]

Prompt energy	(0.7, 12.0) MeV
Delayed energy	(6, 12.0) MeV
Prompt-delayed Δt	(1, 200) μs
Multiplicity veto *(pre)*	Only one signal (0.7, 12) MeV 400 μs before delayed
Multiplicity veto *(post)*	No signal (6, 12) MeV 200 μs after delayed
Water Shield muon veto	Veto (-2, 600) μs after NHIT $>$ 12 in OWS or IWS
AD muon veto	Veto (0, 1.4) ms after $>$3,000 p.e. ($\sim$18 MeV) signal
AD shower veto	Veto (0, 0.4) s after $>3\times10^5$ p.e. ($\sim$1.8 GeV) signal

Proof from a Daya Bay paper that the selection is quite straightforward [5]

Spectral analysis

Predict far detector flux for each energy bin using near detector flux + an oscillation model

Subtleties

- Near detectors see some oscillation—over 2 different baselines
- Livetime/efficiency varies by detector due to muon and multiplicity vetoes

Define χ^2 to include the standard statistical errors plus nuisance parameters to account for systematic uncertainties

Systematic uncertainties

Number of protons/target mass

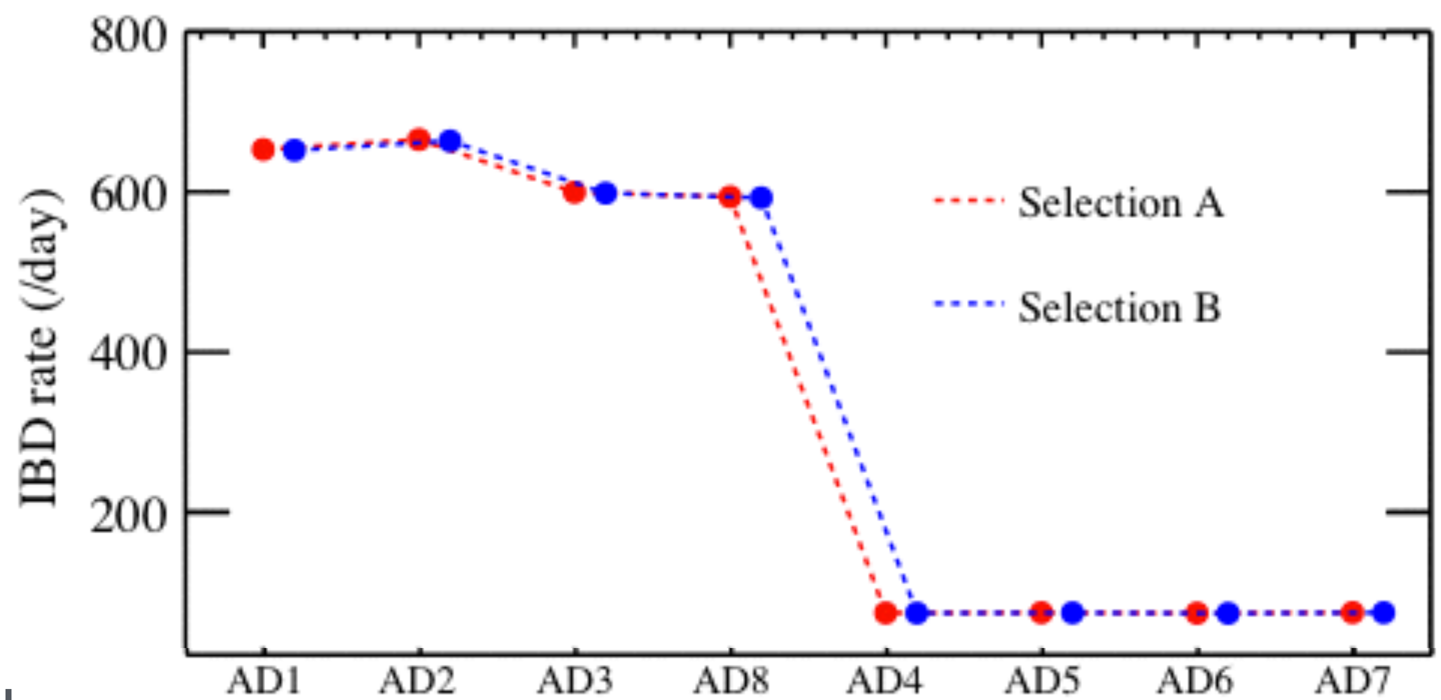
Relative energy scale

Reactor flux (essentially cancels in near/far ratio)

^{9}Li

- Byproduct of cosmic μ's
- Mimic IBD events ⟹ hard to measure rate
- Different rates for each detector hall (near/far)

Largest & purest ν data set



IBD rate for each detector [5]
Selection A/B are 2 different analyses

2,000,000 IBD events

~10^5 times more additional "singles" events (nuclear decays)

There has to be more physics in this data set than mixing parameters, reactor spectrum and a sterile ν search!

Things to look for

High-level

- ν_e disappearance ✔
- sterile ν search ✔
- Other unknown physics (surprises)

Low-level

- Better understanding of backgrounds
- Other backgrounds not yet considered

Explore the data

Use machine learning

Find patterns without knowing exactly what to look for

Group/sort data based on qualities humans may miss

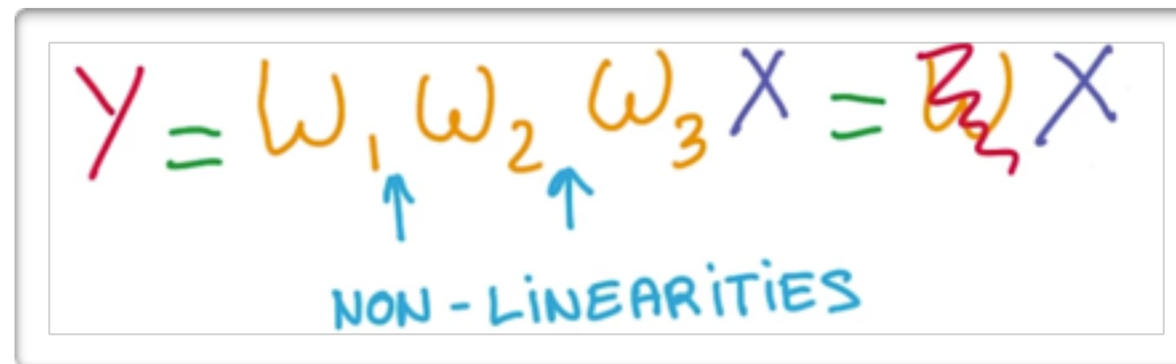
Learn from 10^3-10^6 examples (many more than humans can deal with)

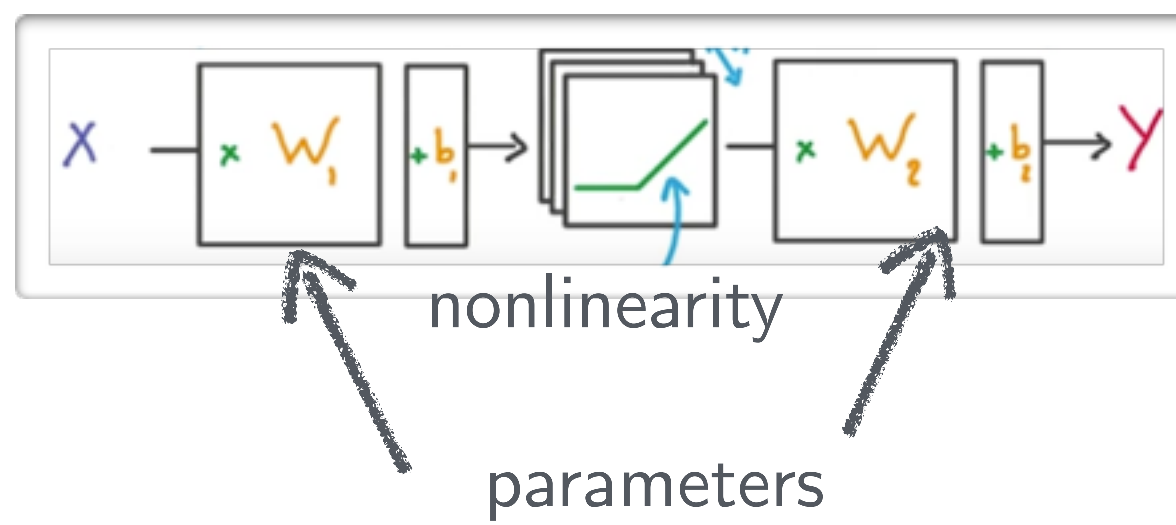
Neural networks

Series of matrix multiplies to make a prediction based on input vector

Nonlinear function between matrices allows for more complex models

Training is adjusting entries in matrix to give the desired “predictions” for given inputs





These pictures are from [6]

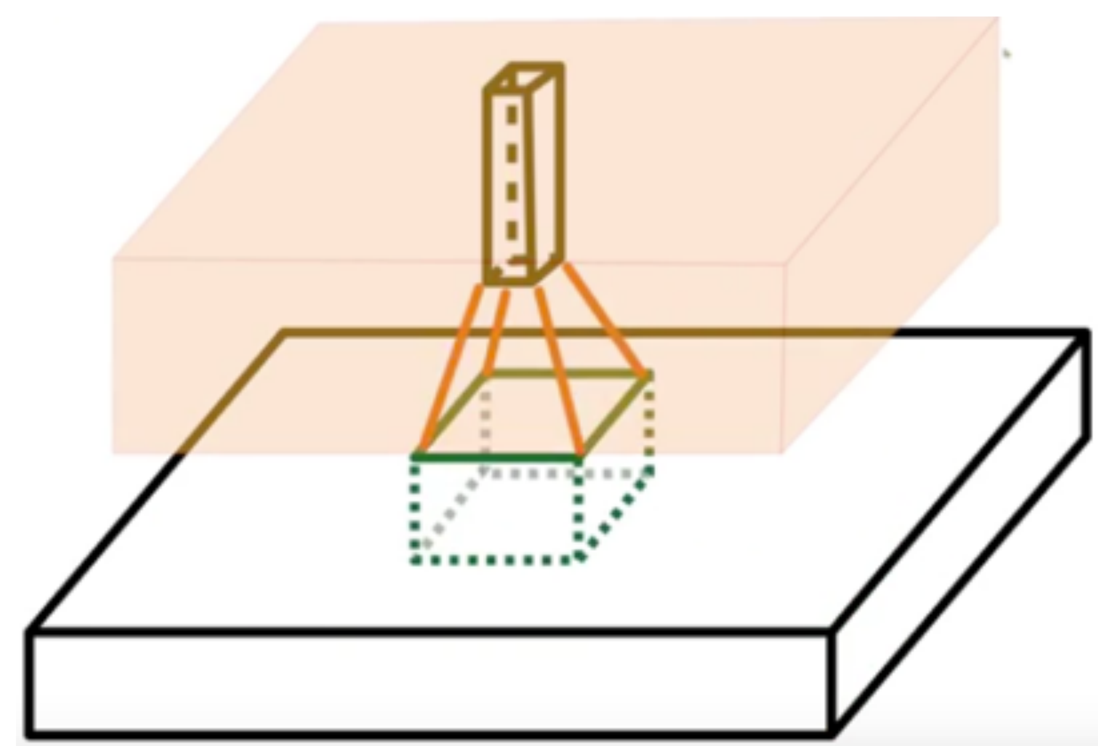
Convolutional NNs

Convolution: for images

Want to recognize features no matter where they are

Instead of one big matrix for the whole image, go one small patch at a time

Layer's output is a "feature map" showing locations of recognized features



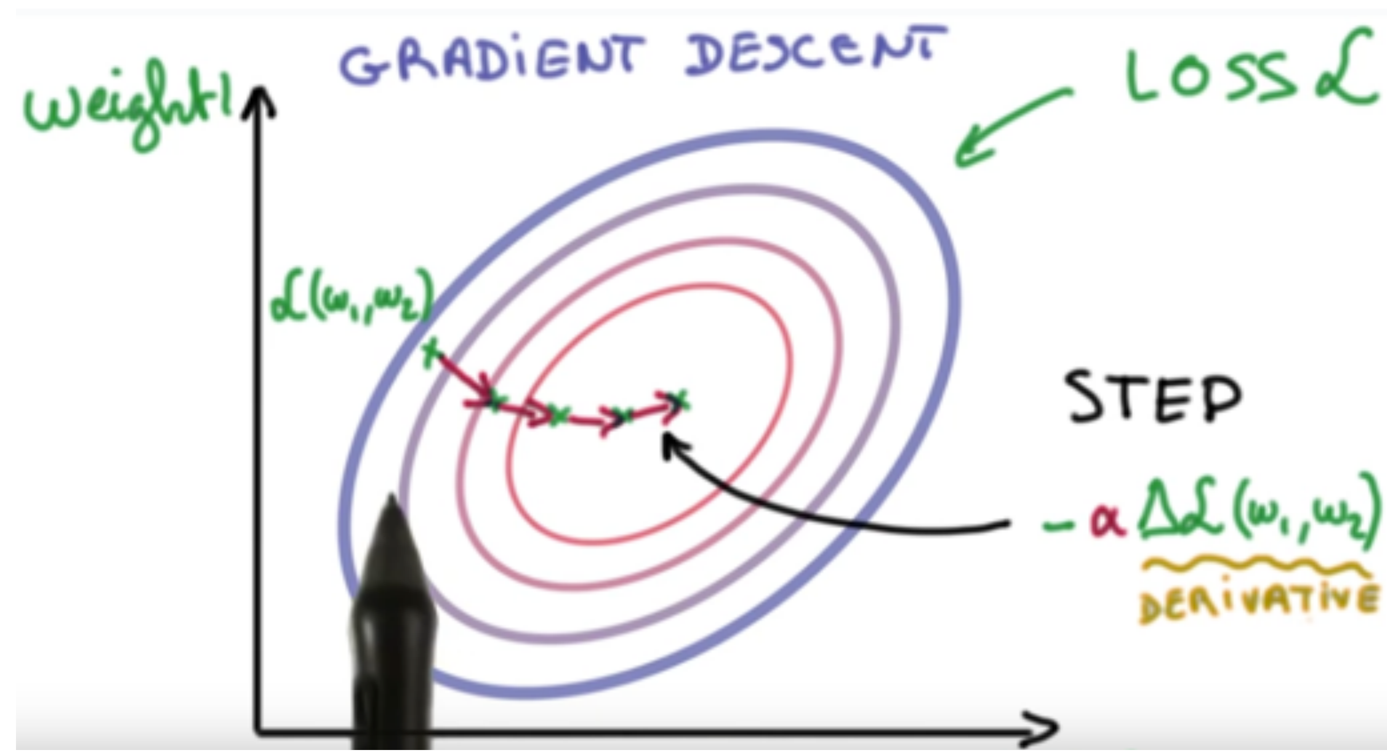
Training a NN

Gradient/steepest descent

Define loss/cost to evaluate one NN input

Repeat for many inputs to find total loss for model

Take derivative w.r.t. each NN paramter and adjust in the opposite direction



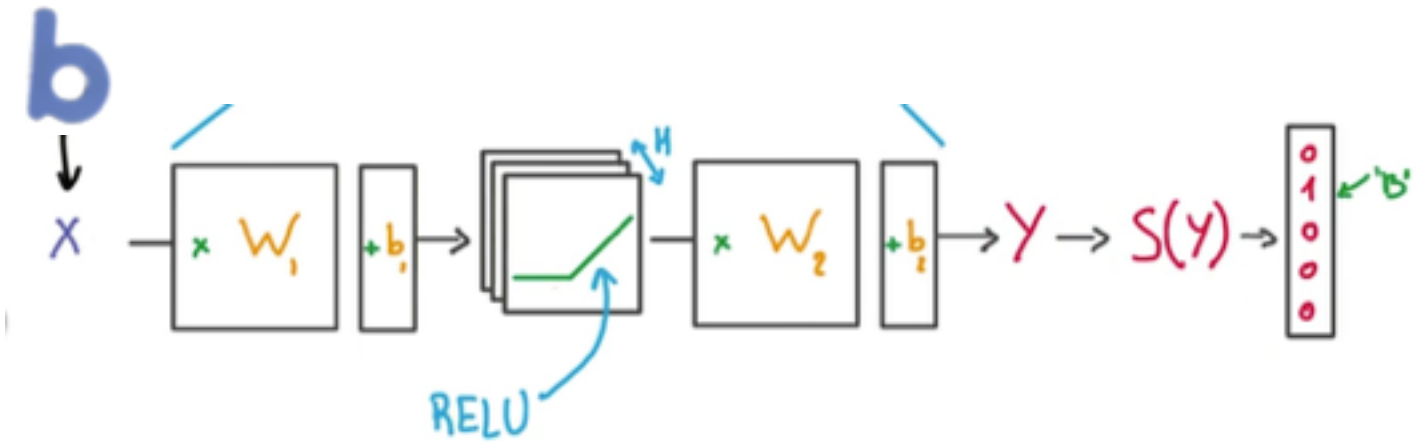
Interpretation of NN in/output

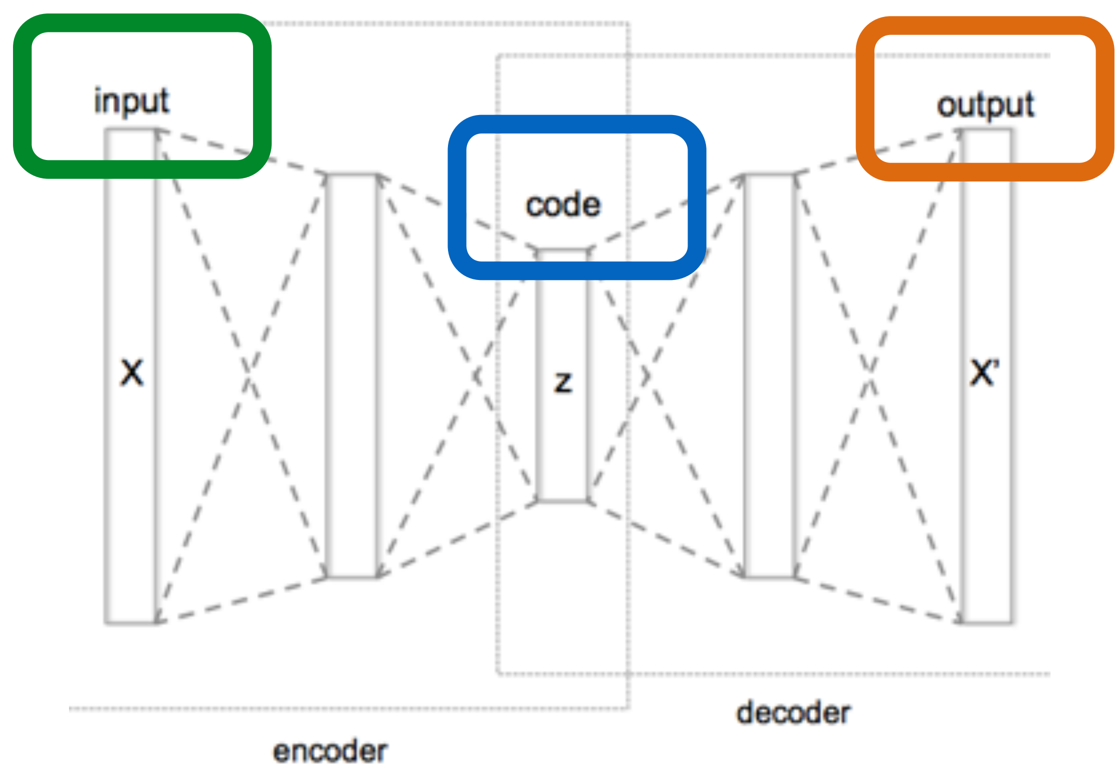
Input vector is some data

- An image (reshaped into a column vector)
- List of E, p, n_{jet}, etc.

Output interpretation varies

- Supervised learning: i-th component as prediction that input is of type i
- Unsupervised: output is attempted reconstruction of input





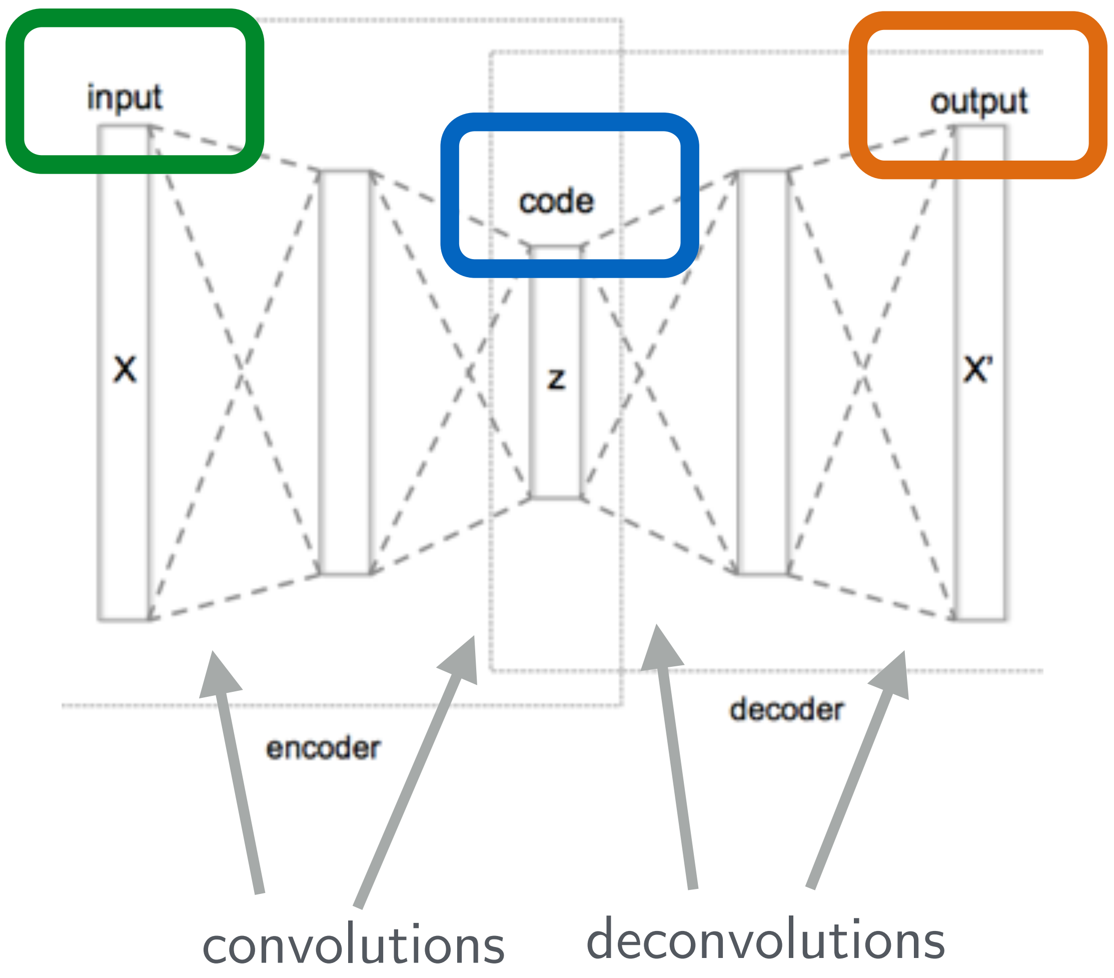
Source: [1]

Unsupervised learning

Easy to train NN to predict classes if you know the answer for some inputs

What if you don't?

- Cannot train NN on class prediction
- Train NN to recover ("reconstruct") input
- Interpret middle layer as encoding of input in "semantic space"



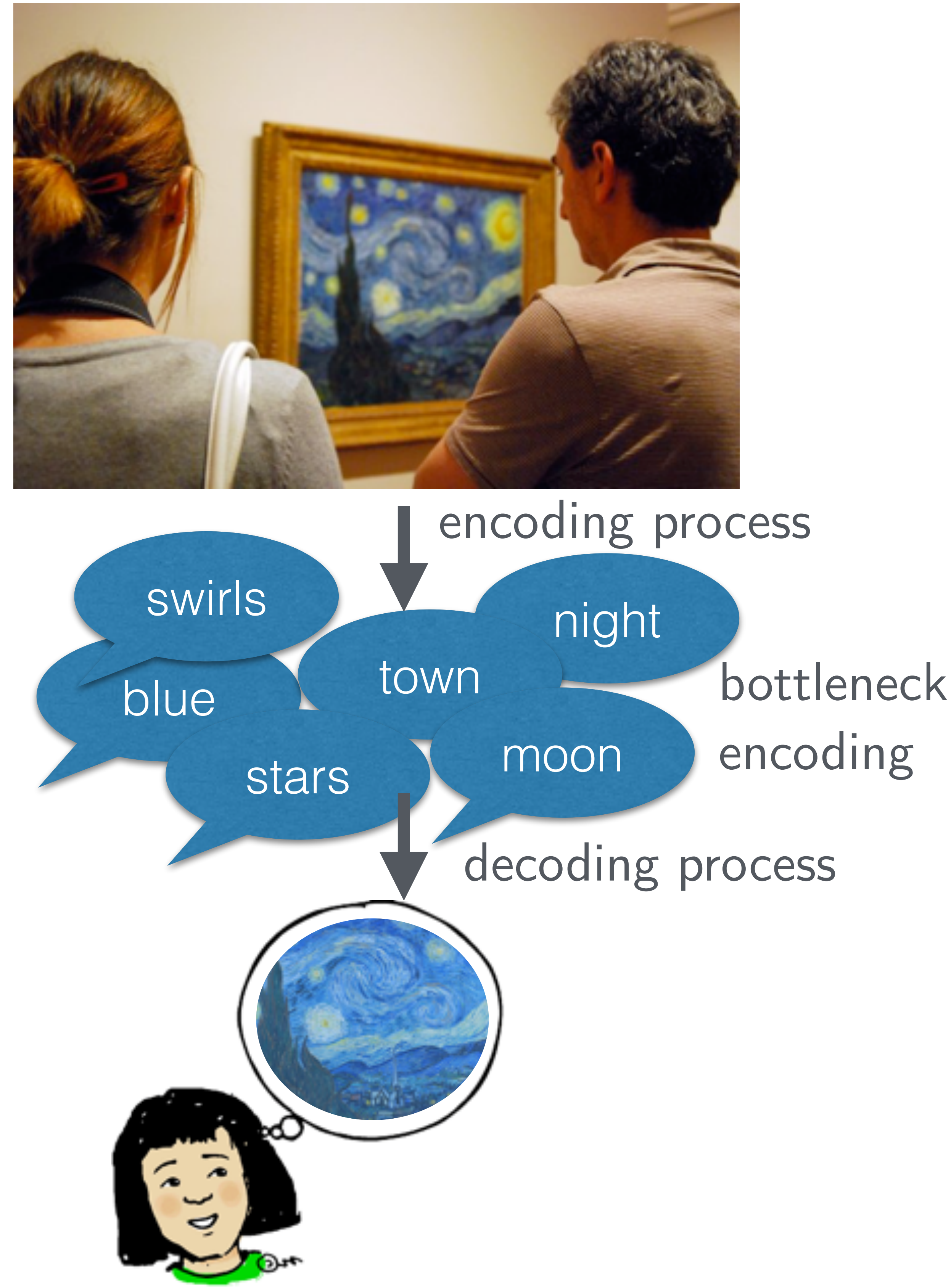
The bottleneck

Special layer whose output has small number of components

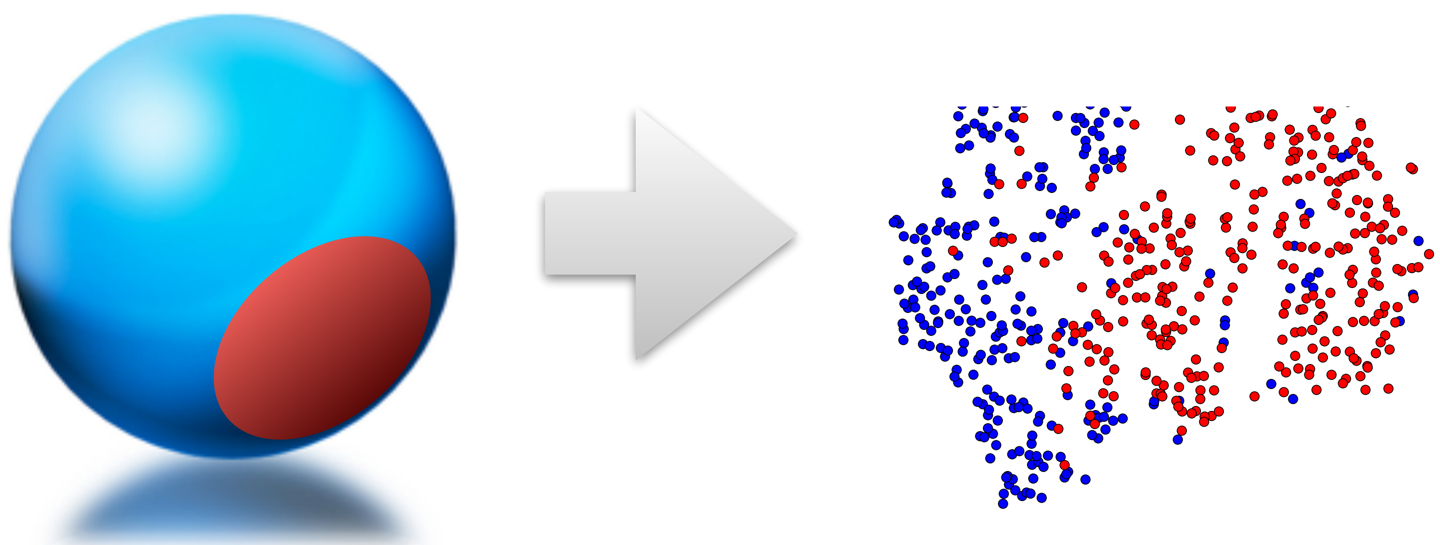
Interpret as "encoding" of input as understood by first half of network

Second half of network must start with encoding and recover original input

Expect similar inputs to have similar encodings



t-SNE evaluation



Examine encodings to look for patterns

Expect similar style events to have similar encodings

Use t-SNE algorithm to map N-dimensional encodings onto 2D plot [7]

Nearby points in N dimensions become nearby points in 2D plot

Progress on my project

Computing resources

Cori and Edison supercomputers at NERSC

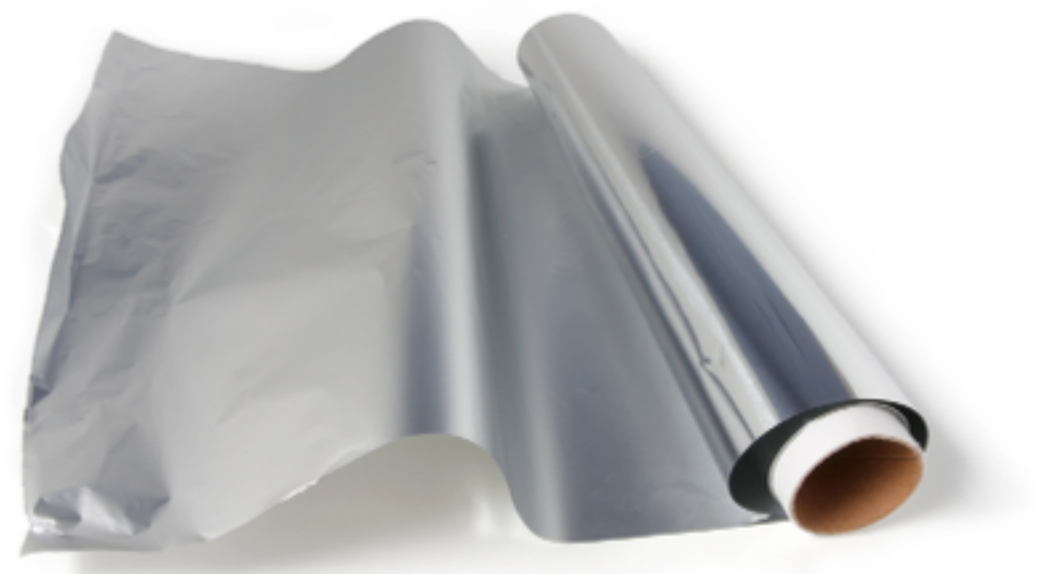
Software frameworks: all in Python!

- Theano + Lasagne for NN
- Scikit-learn for t-SNE
- HDF5 + numpy for data storage and manipulation

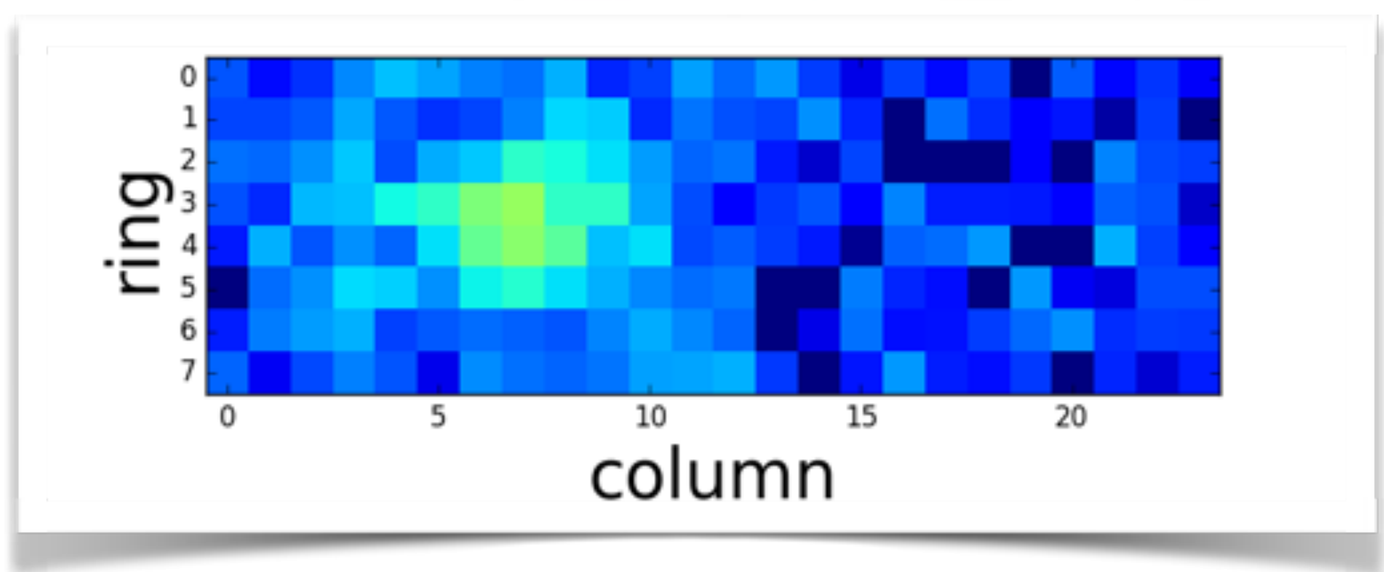
Collaborators: MANTISSA-HEP machine learning group @ LBNL

- Offering machine learning expertise to high energy physicists
- Performed a related analysis on Daya Bay data [8]

Interpret PMTs as pixels



Unroll cylindrical detector into 8×24 pixel map of PMT charges for each detector trigger



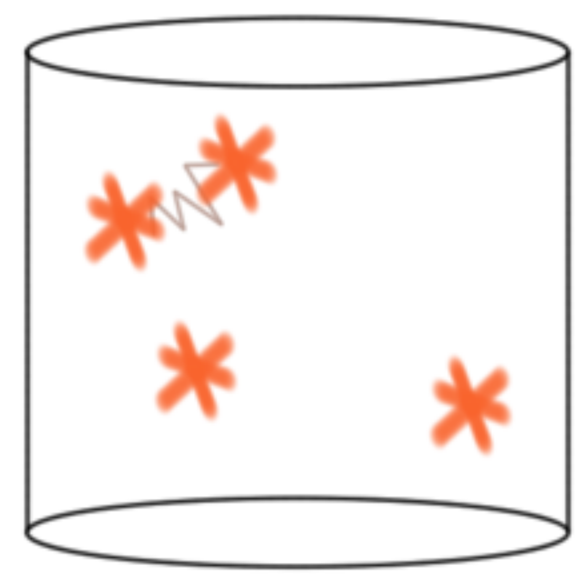
Feed into NN to look for ways to distinguish IBDs from various backgrounds

Write traditional analysis using insights from NN

Study: IBD vs. accidentals

Accidentals are two uncorrelated signals that mimic an IBD event

- Background in Daya Bay: 1% of IBD sample is accidental
- Well-understood background allows for evaluation of NN methods



Use autoencoder to analyze differences between IBDs and accidentals

Input data

- pair up prompt and delayed images to make a 2-channel image similar to RGB in a photo
- 9,000 IBD events, 9,000 accidental events

Architecture

Use a basic architecture for first study

Many opportunities for improvement

Input 2 channels representing prompt and delayed

8×24 pixels per channel

Bottleneck width of 16 “pixels”

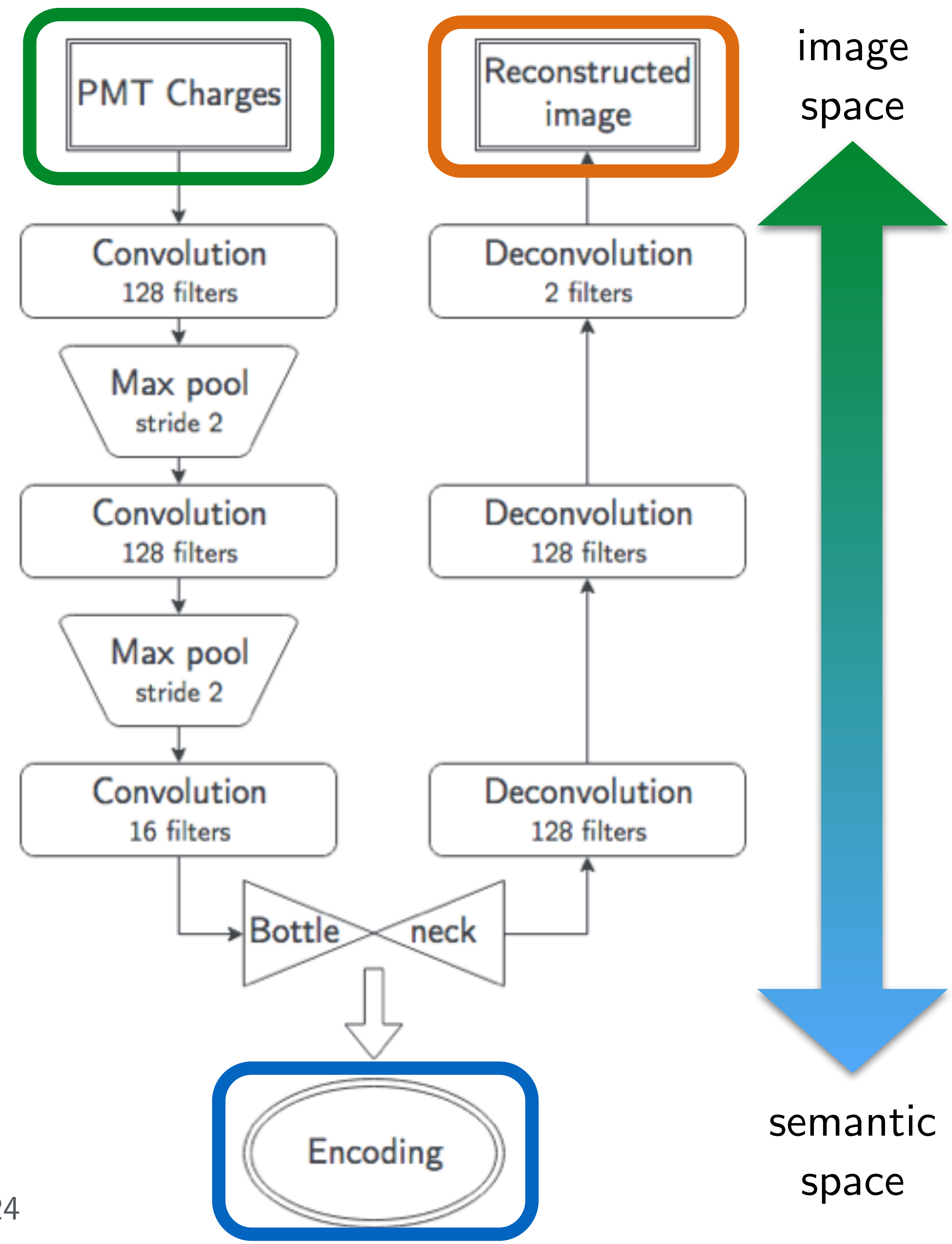


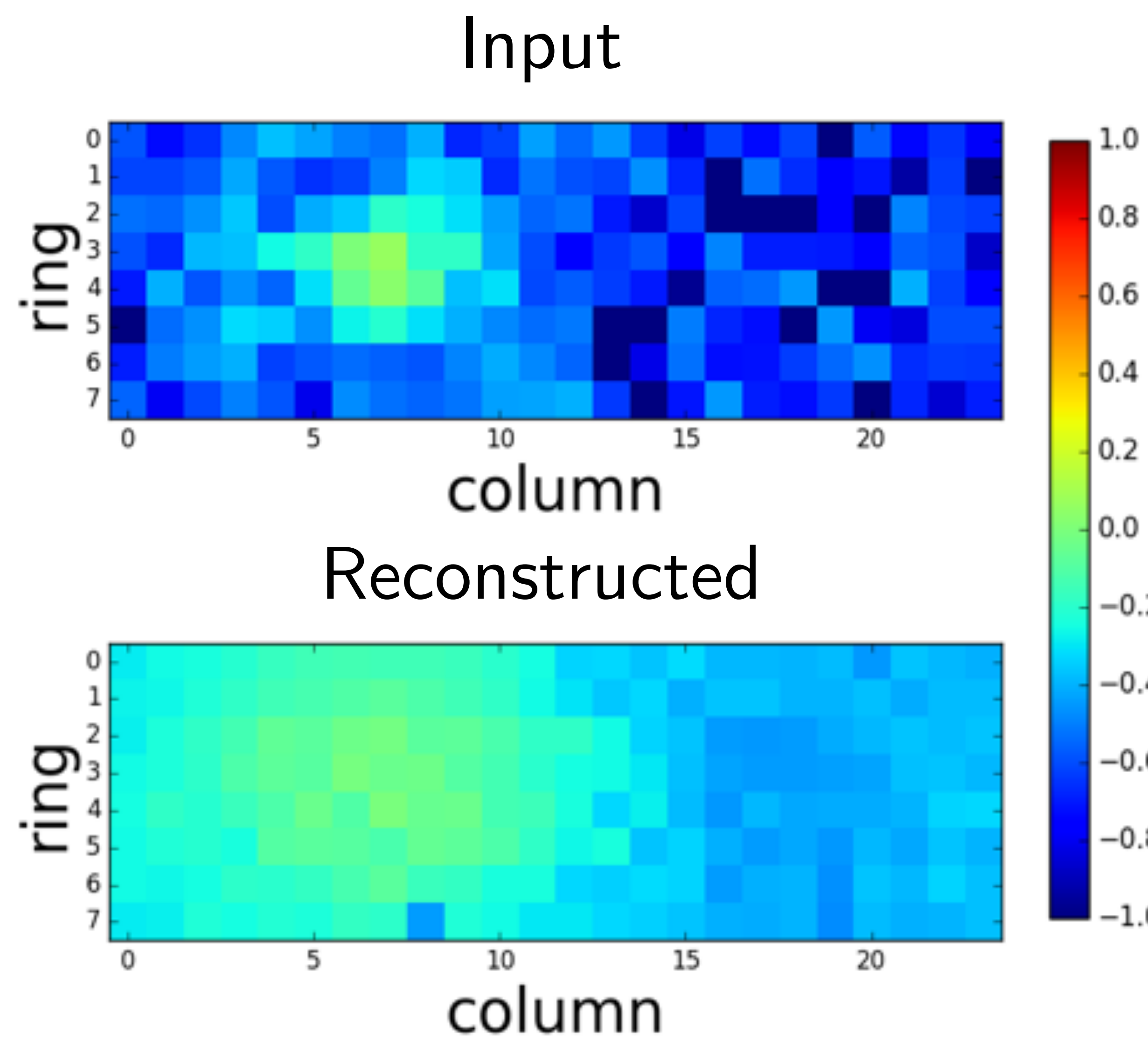
Image reconstructions

Zeroth-order evaluation of training

Qualitatively good reconstructions indicate the NN is learning how to encode the images

Does not accurately reconstruct fluctuations in PMT charge

Does reconstruct position and intensity of charge pattern



t-SNE plot

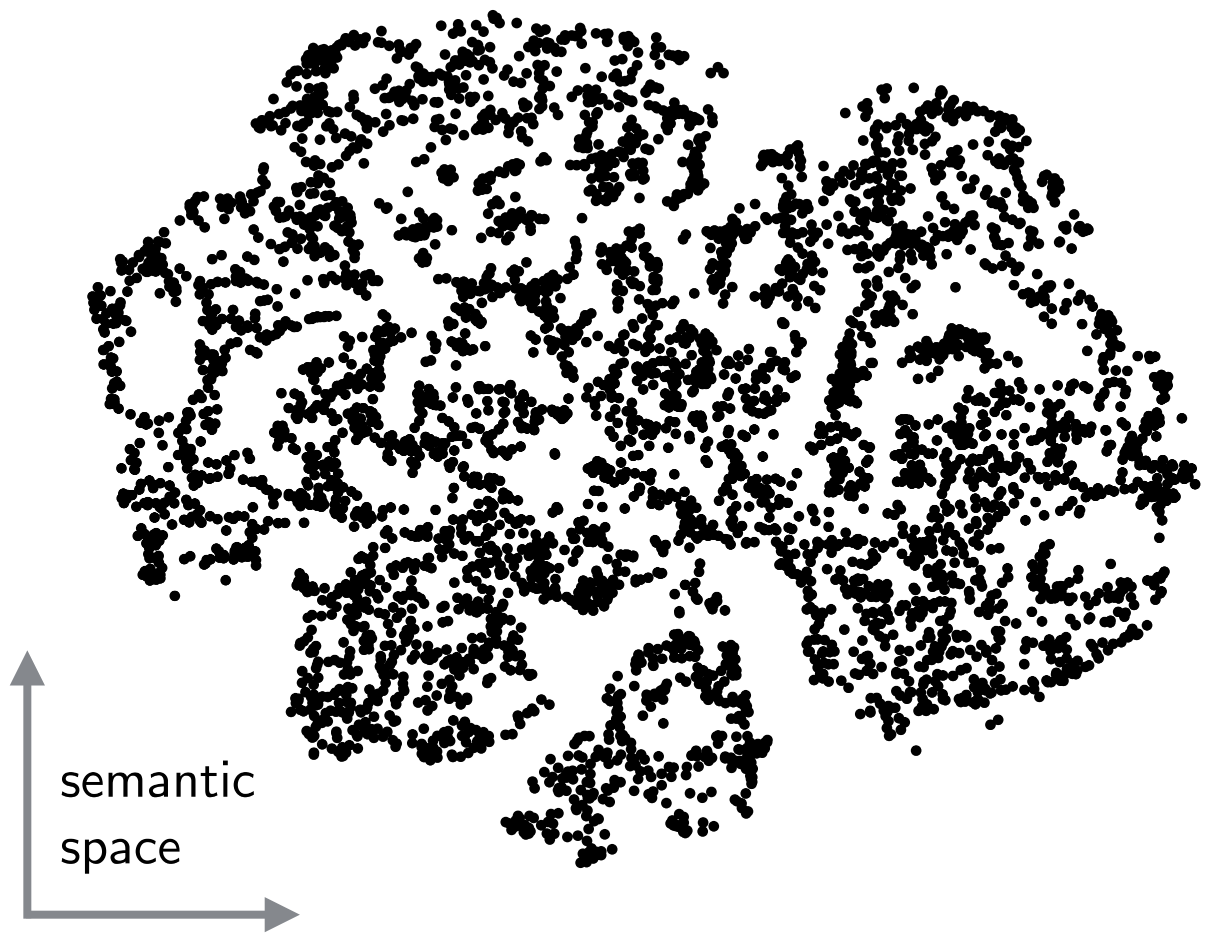
5120 data points

Each point represents the bottleneck encoding of one IBD or accidental event

Nearby points on this plot have similar encodings

Axes do not represent physical quantities

Information is in the distance between data points



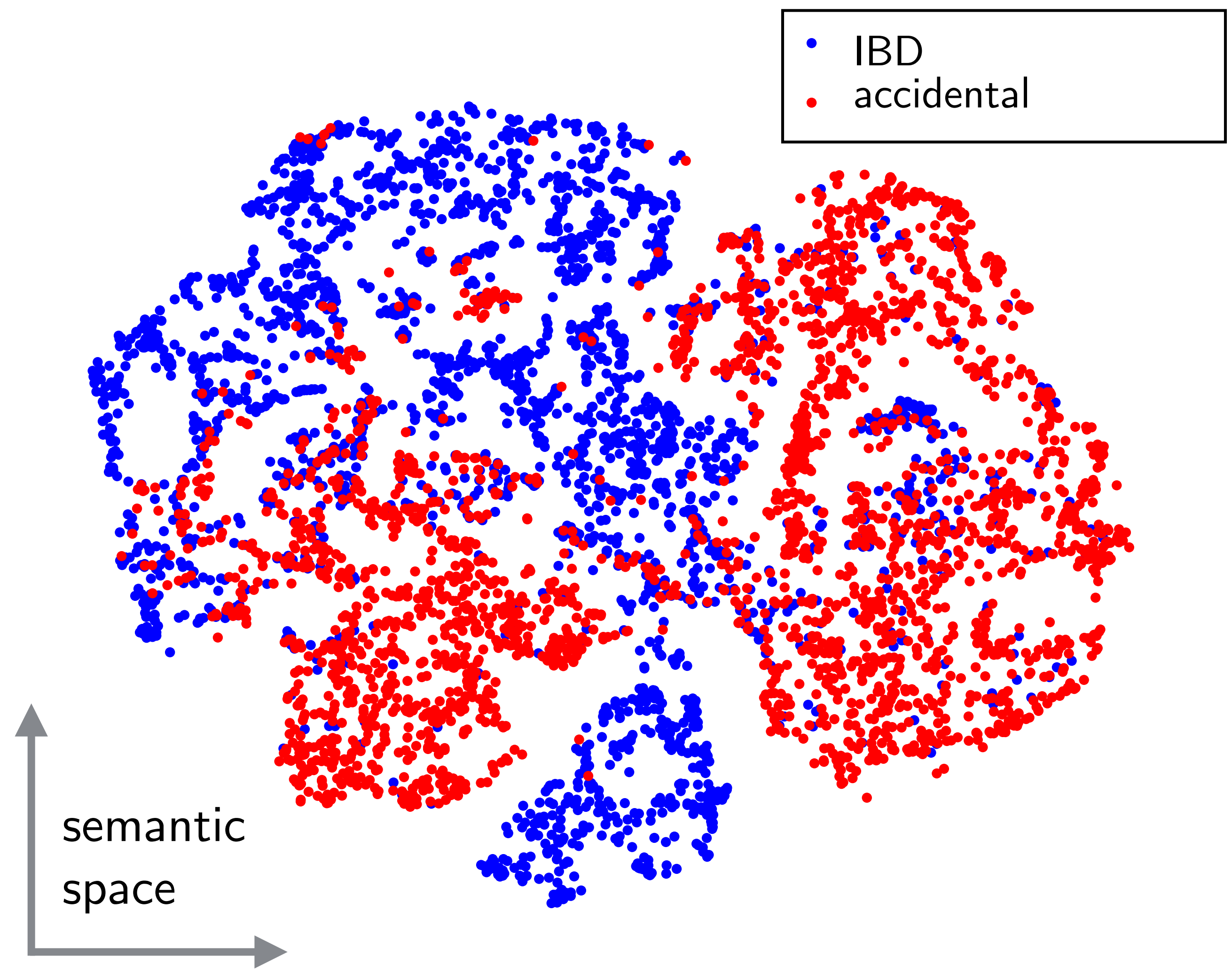
t-SNE plot color-coded

Same 5120 data points

Color represents which data set the point belongs to (IBD or accidental)

NN *was not* given this information!

Separation of red and blue indicates *NN discovered different features* for IBD and accidentals events



What's in store for the future

Continue analysis on current NN and t-SNE plot to uncover what NN learned & validate result

Code up new, more sophisticated NNs for better chances of success with ^{9}Li

Determine signature of ^{9}Li using NN (if such a signature exists)

Write analysis taking advantage of this new knowledge

Thank you

References

[1] Google Image Search and Wikipedia

[2] F.P. An et al. *Phys. Rev. Lett.* **115**, 111802 (2015).

[3] F.P. An et al. arXiv: 1607.05378.

[4] F.P. An et al. NIMA **685**, 78 (2012).

[5] F.P. An et al. arXiv: 1610.04802.

[6] Udacity. https://www.udacity.com/course/deep-learning--ud730.

[7] Journal of Machine Learning Research 9, 2579 (2008)

[8] E. Racah, et al. "Revealing Fundamental Physics." arXiv:1601.07621