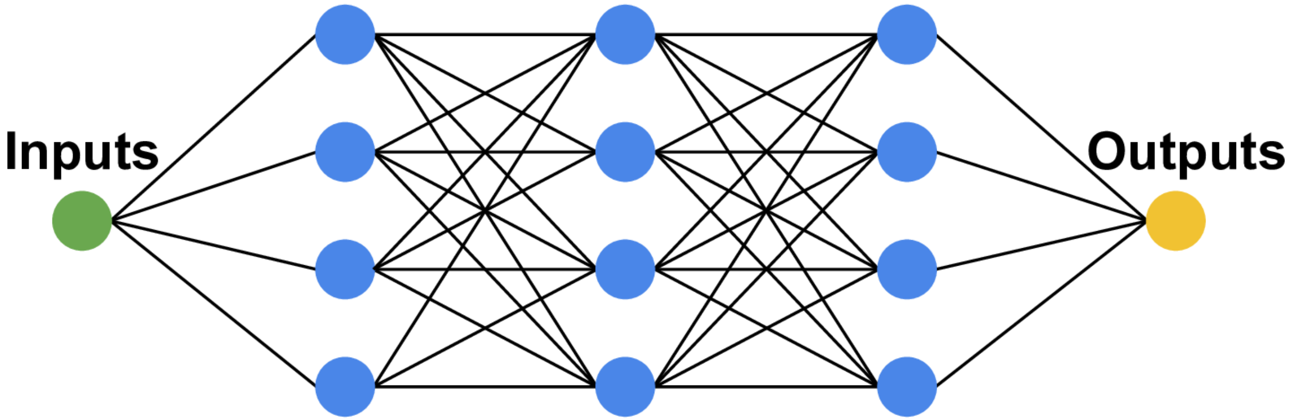
PHYSICALLY INTERPRETABLE MACHINE LEARNING FOR NUCLEAR MASSES



Los Alamos NATIONAL LABORATORY EST. 1943

LA-UR-22-22701

MATTHEW MUMPOWER

INT ML Nuclear Physics Workshop

Thursday March 31st 2022

Theoretical Division

NUCLEAR PHYSICS IS UBIQUITOUS IN MODERN APPLICATIONS

Example: fission yields are needed for a variety of applications and cutting-edge science

Industrial applications: simulation of reactors, fuel cycles, waste management

Experiments: backgrounds, isotope production with radioactive ion beams (fragmentation)

Science applications: nucleosynthesis, light curve observations

Other Applications: national security, nonproliferation, nuclear forensics

Figure: Chi-Nu detector (left), Reactor cooling towers (middle), Neutron star merger (right)

NUCLEAR ASTROPHYSICS REQUIRES NUCLEAR THEORY

Even in the era of **FRIB**

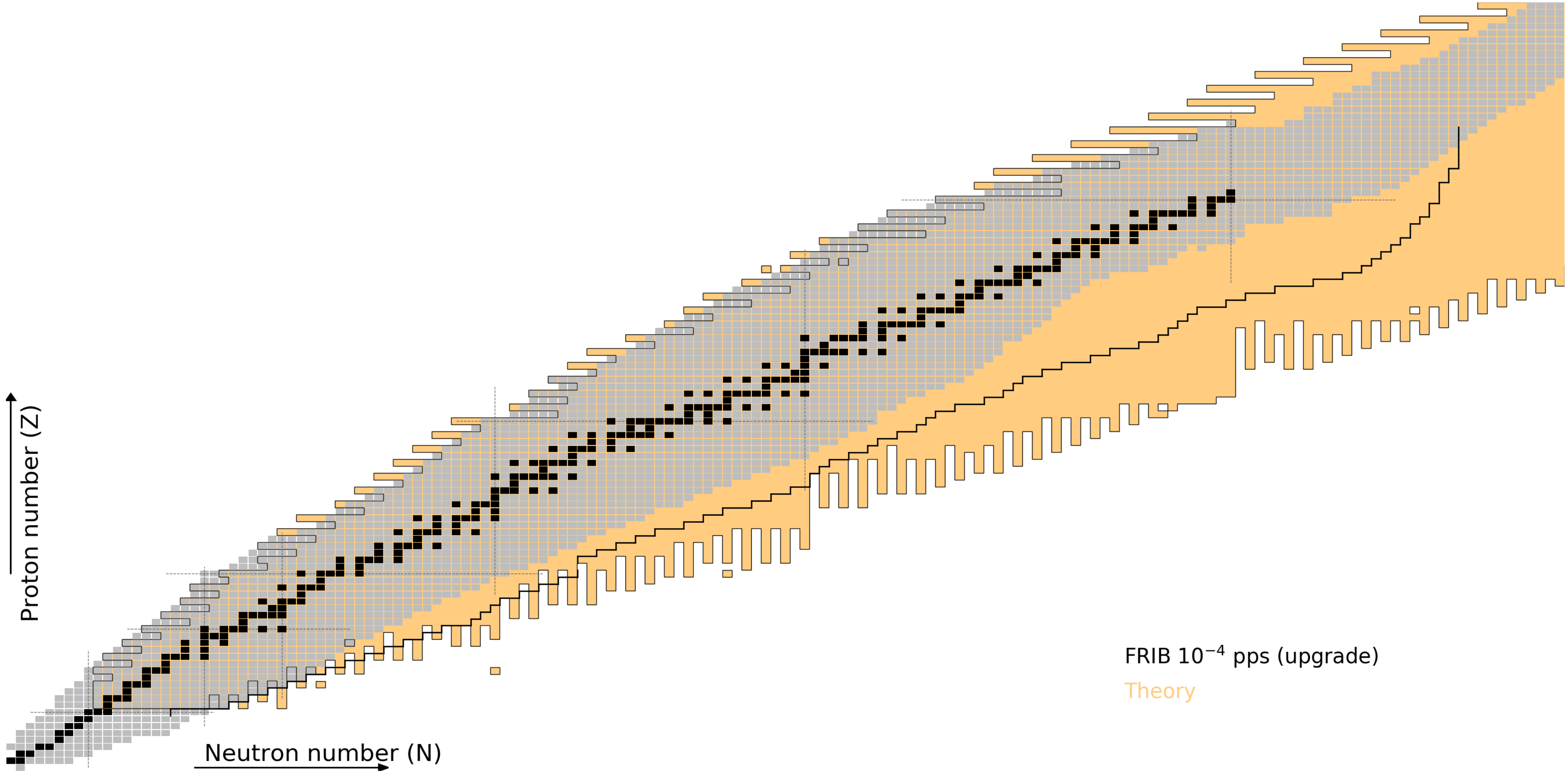
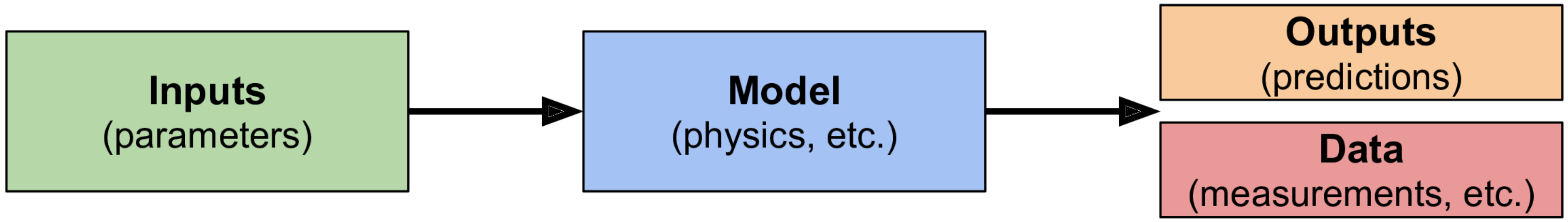


Figure by **Mumpower**

NUCLEAR PHYSICS FROM A FORWARD PROBLEM PERSPECTIVE



When we model nuclear properties we think of it as a 'forward' problem

We start with our model and parameters and try to match measurements / observations

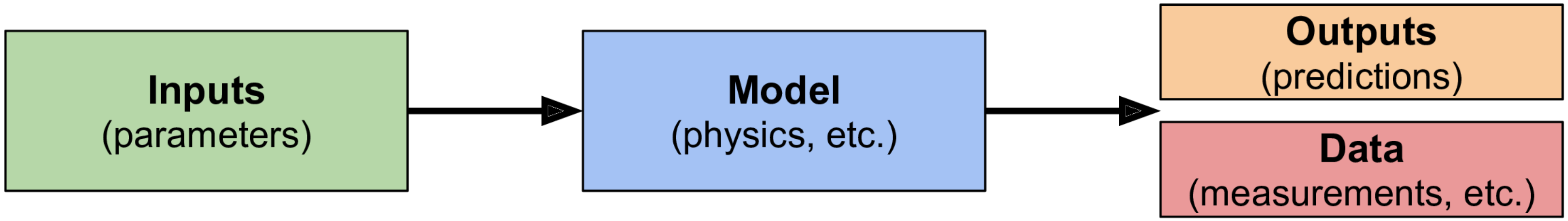
This is an extremely successful approach and is generally how we use theory models

Mathematically: $f(\vec{\boldsymbol{x}}) = \vec{\boldsymbol{y}}$

Where f is our model, $\vec{\boldsymbol{x}}$ are the parameters for the model and $\vec{\boldsymbol{y}}$ are the predictions

Figure by **Mumpower**

FORWARD PROBLEM... PROBLEMS...



There can be many challenges with this approach

For instance: model computationally expensive or how do we update our model if we don't match data?

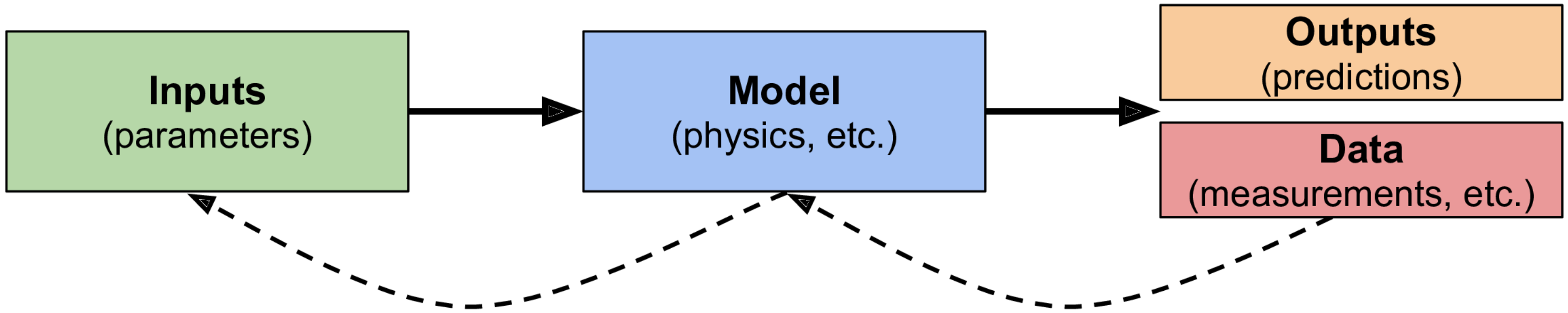
This can be challenging

We don't always know what modifications are required or the physics could be very hard if not impossible to model given current computational limitations (e.g. many-body problem)

Treating this as an 'inverse problem' provides an alternative approach...

Figure by **Mumpower**

NUCLEAR PHYSICS FROM AN INVERSE PROBLEM PERSPECTIVE



Suppose we start with the nuclear data (and its associated uncertainties)

We can then try to figure out what models may match relevant observations

This approach allows us to *vary* and optimize *the model*!

Mathematically, still: $f(\vec{x}) = \vec{y}$; but we want to now find f; fixing $\vec{x}$

Machine learning and artificial intelligence algorithms are ideal for this class of problems...

Figure by **Mumpower**

AN EXAMPLE OF HOW THIS WORKS (FROM MY LIFE)

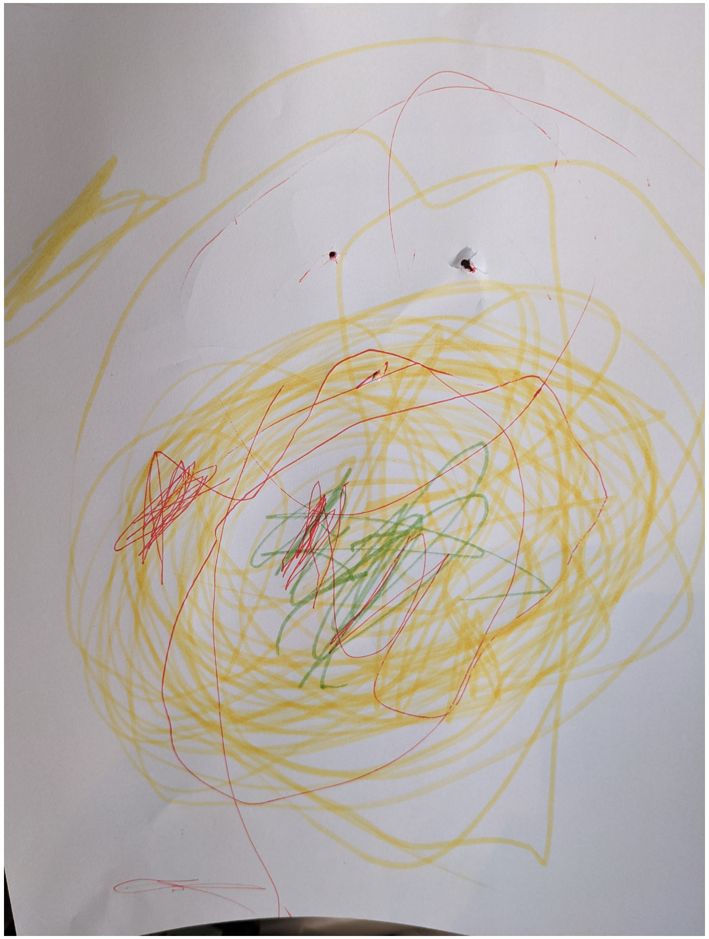
My son Zachary has been very keen on learning his letters from an early age

Letters are well known (this represents the data we want to match to)

But the model (being able to draw - eventually letters) must be developed, practiced and finally optimized

This is a complex, iterative process - it takes times to find 'f'!

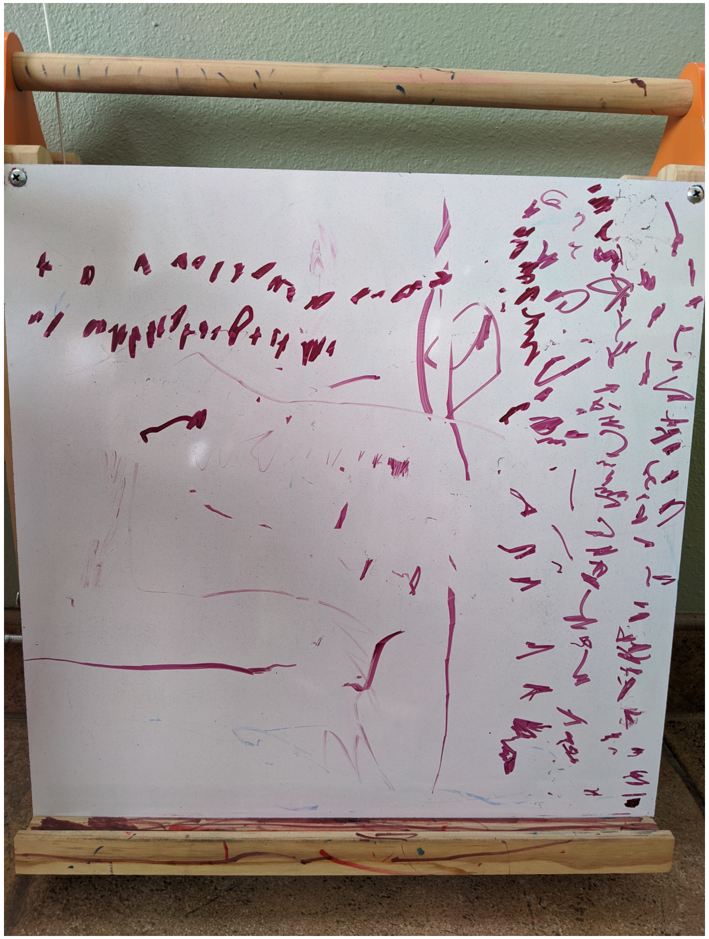
AN EXAMPLE OF HOW THIS WORKS (FROM MY LIFE)



Example from 1 years old; random drawing episode; time taken: 10 minutes

Drawing by Z. R. Mumpower

AN EXAMPLE OF HOW THIS WORKS (FROM MY LIFE)



Example from 2.5 years old; first ever attempt to draw letters / alphabet; time taken: 2 hours!

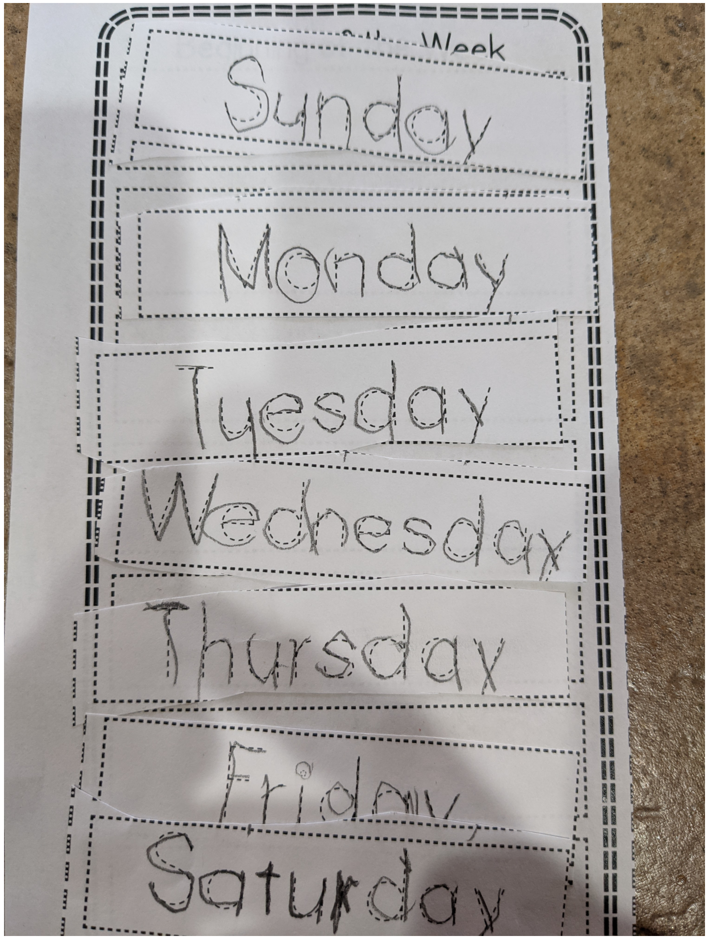
Drawing by Z. R. Mumpower

AN EXAMPLE OF HOW THIS WORKS (FROM MY LIFE)

Example from 3.5 years old; drawing scenery; time taken: 10 minutes

Drawing by Z. R. Mumpower

AN EXAMPLE OF HOW THIS WORKS (FROM MY LIFE)



Example from 4 years old; tracing the days of the week; time taken: 30 minutes

Drawing by Z. R. Mumpower

LET'S APPLY THIS IDEA TO A BASIC PROPERTY...

A relatively easy choice is a scalar property; how about atomic binding energies (masses)?

Recall: $f(\vec{\boldsymbol{x}}) = \vec{\boldsymbol{y}}$

f will be a neural network (a model that can change)

$\overrightarrow{\boldsymbol{y}_{obs}}$ are the observations (e.g. Atomic Mass Evaluation) we want to match $\vec{\boldsymbol{y}}$ with

But what about $\vec{\boldsymbol{x}}$?

Why not a *physically motivated* feature space!?

proton number (Z), neutron number, (N), nucleon number (A), etc.

Lovell, Mohan, Sprouse & **Mumpower** arXiv:2201.00676 (2022)

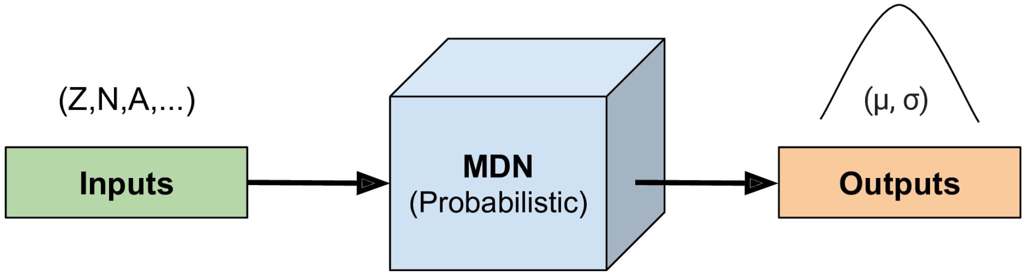
NETWORK FEATURE SPACES ($\vec{x}$)

Model Name	Feature Space	Output
M2	N, Z	δM
M6	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$	δM
M8	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$, Z_{EO}, N_{EO}	δM
M10	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$, Z_{EO}, N_{EO} ΔN, ΔZ	δM
M12	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$, Z_{EO}, N_{EO} ΔN, ΔZ, N_{shell}, Z_{shell}	δM
MS2	N, Z	δM, S_n
MS6	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$	δM, S_n
MS8	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$, Z_{EO}, N_{EO}	δM, S_n
MS10	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$, Z_{EO}, N_{EO} ΔN, ΔZ	δM, S_n
MS12	N, Z, A, $A^{2/3}$, $Z(Z-1)/A^{1/3}$, $(N-Z)^2/A$, Z_{EO}, N_{EO} ΔN, ΔZ, N_{shell}, Z_{shell}	δM, S_n

We use a variety of possible physically motivated features as inputs into the model

Lovell, Mohan, Sprouse & **Mumpower** arXiv:2201.00676 (2022)

MIXTURE DENSITY NETWORK



We take a Bayesian approach: our network inputs are sampled based on nuclear data uncertainties

Our network outputs are therefore **statistical**

We can represent outputs by a collection of Gaussians (for masses we only need 1)

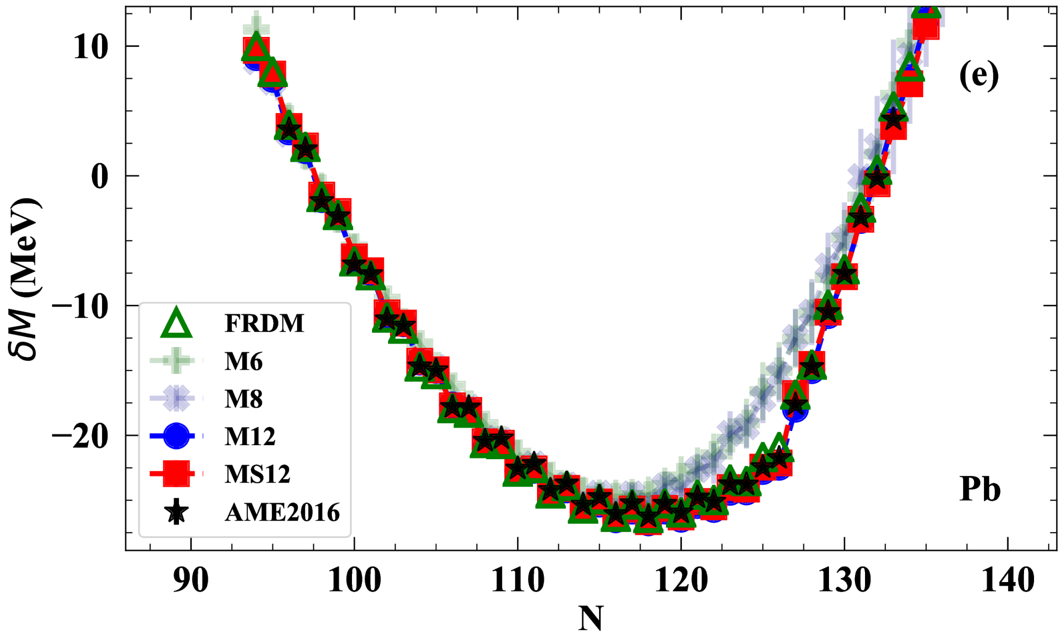
Our network also provides a well-quanitfied estimate of uncertainty (fully propagated through the model)

Network topology: fully connected $\sim$4-8 layers; 8 nodes; regularization via weight-decay

Last layer converts output to Gaussian ad-mixture

Bishop (1994 & 2006) • Lovell, Mohan, Sprouse & **Mumpower** arXiv:2201.00676 (2022)

RESULTS: MODELING ATOMIC MASSES



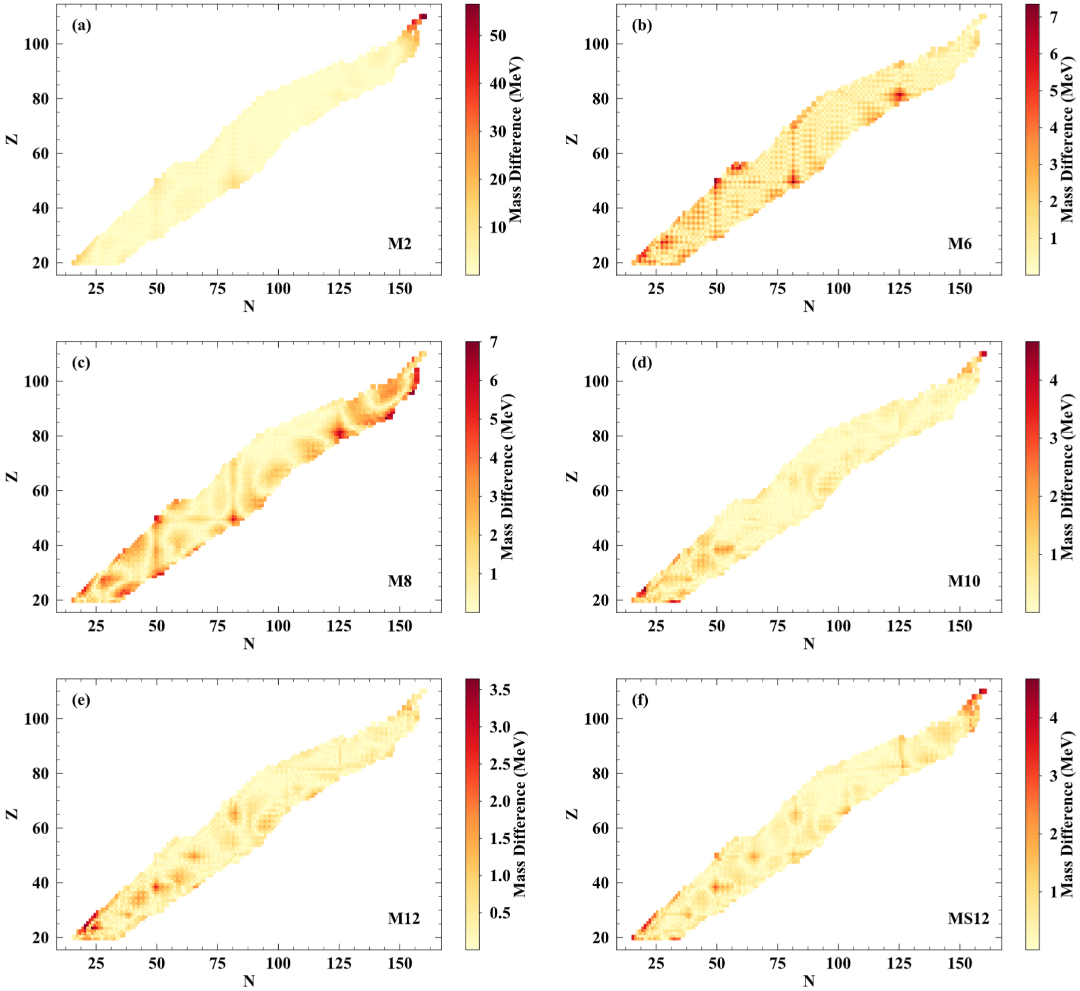
Here we show results along the lead (Pb) $[Z = 82]$ isotopic chain

Data from the Atomic Mass Evaluation (AME2016) in black stars

Increasing sophistication of the model feature space indicated by M##

Lovell, Mohan, Sprouse & **Mumpower** arXiv:2201.00676 (2022)

RESULTS: MODELING ATOMIC MASSES



Results across the chart of nuclides with increasing feature space complexity

Note the decrease in the maximum value of the scale

Lovell, Mohan, Sprouse & **Mumpower** arXiv:2201.00676 (2022)

RESULTS: MODELING ATOMIC MASSES

A summary of results

Model	δM σ_{RMS} **(MeV)**	S_n σ_{RMS} **(MeV)**
M2	3.90	—
MS2	2.43	1.25
M6	1.57	—
MS6	2.07	1.21
M8	1.66	—
MS8	2.21	0.57
M10	0.58	—
MS10	0.76	0.57
M12	0.56	—
MS12	0.64	0.47

Some lessons learned:

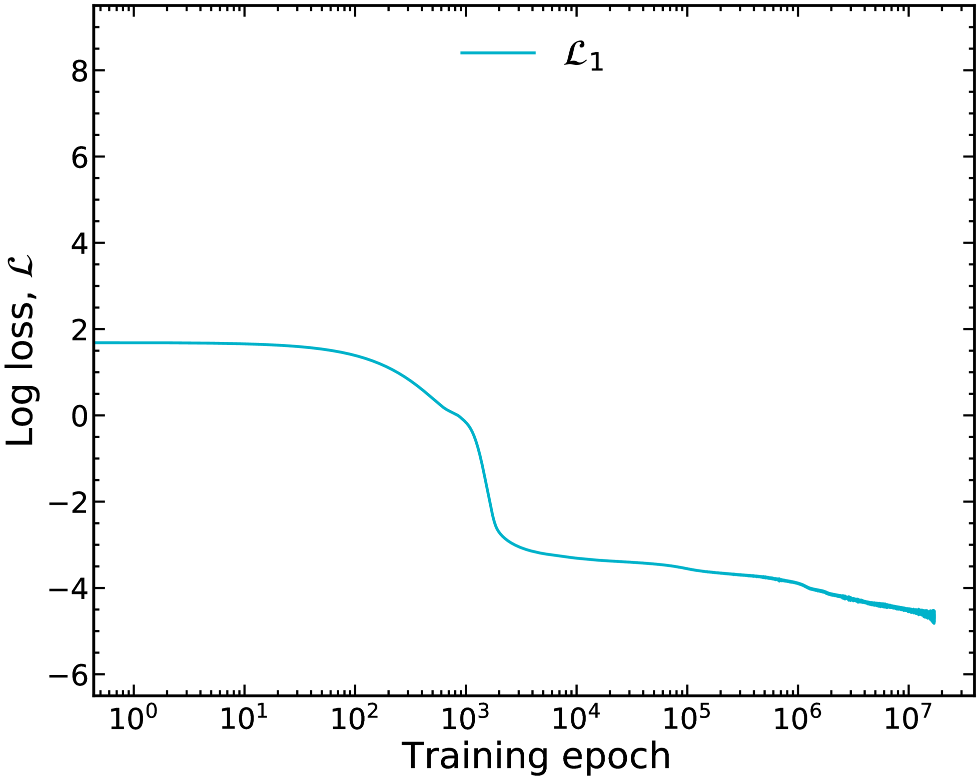
More physics added to the feature space → the better our description of atomic masses

Attempting to fit and predict other mass related quantities (e.g. separation energies) does impact our results

Although marginal compared to choice of feature space

Lovell, Mohan, Sprouse & **Mumpower** arXiv:2201.00676 (2022)

TRAINING ON MASS DATA

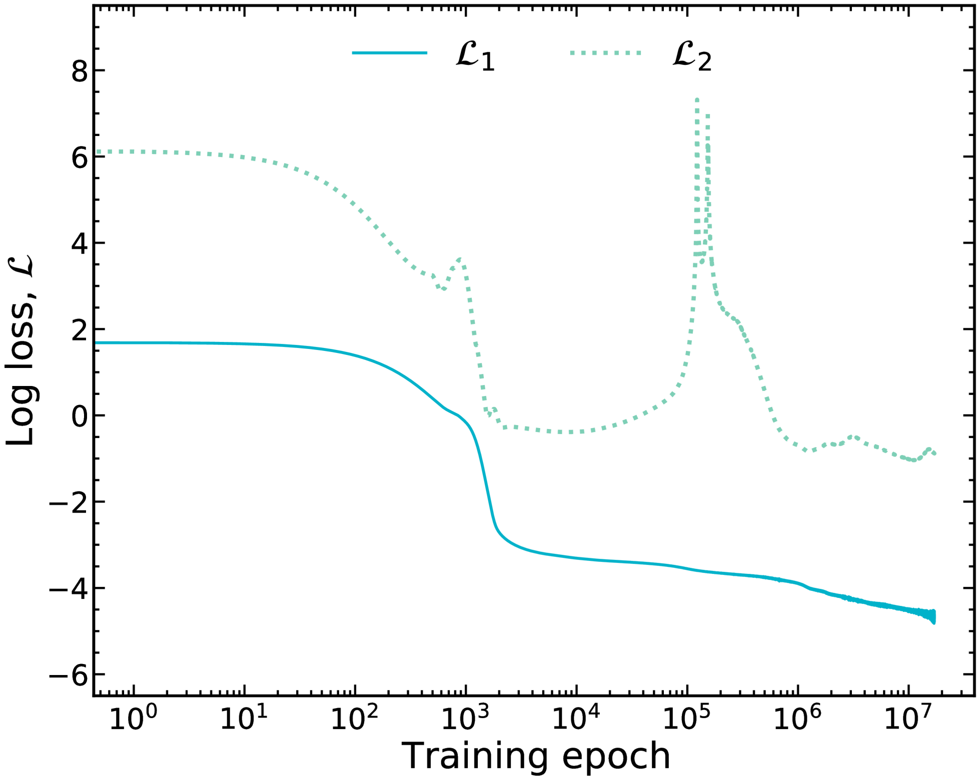


We train on mass data using a log-loss, $\mathcal{L}_1$

Training takes a considerably long time

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

INTRODUCE PHYSICAL CONSTRAINT VIA LOSS FUNCTION

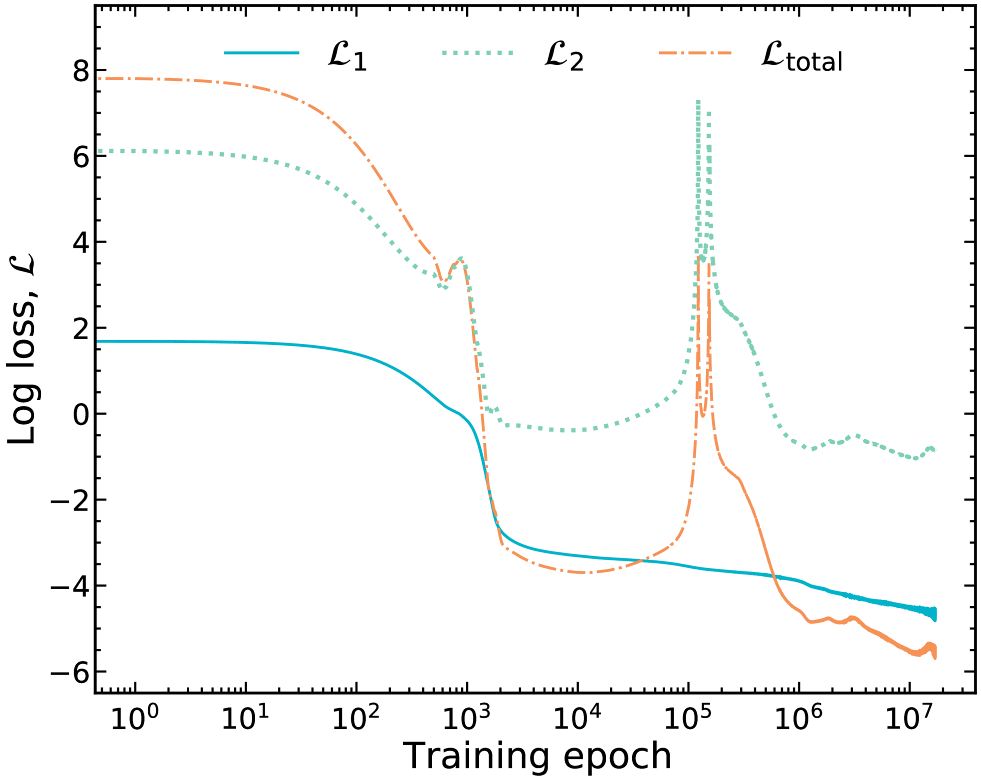


Now we optimize data, $\mathcal{L}_1$, **and** introduce a second (physical) constraint, $\mathcal{L}_2$

We have tried many physics-based losses, this particular one (Garvey-Kelson relations) performs well

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

TOTAL LOSS: SUM THE TWO LOSSES

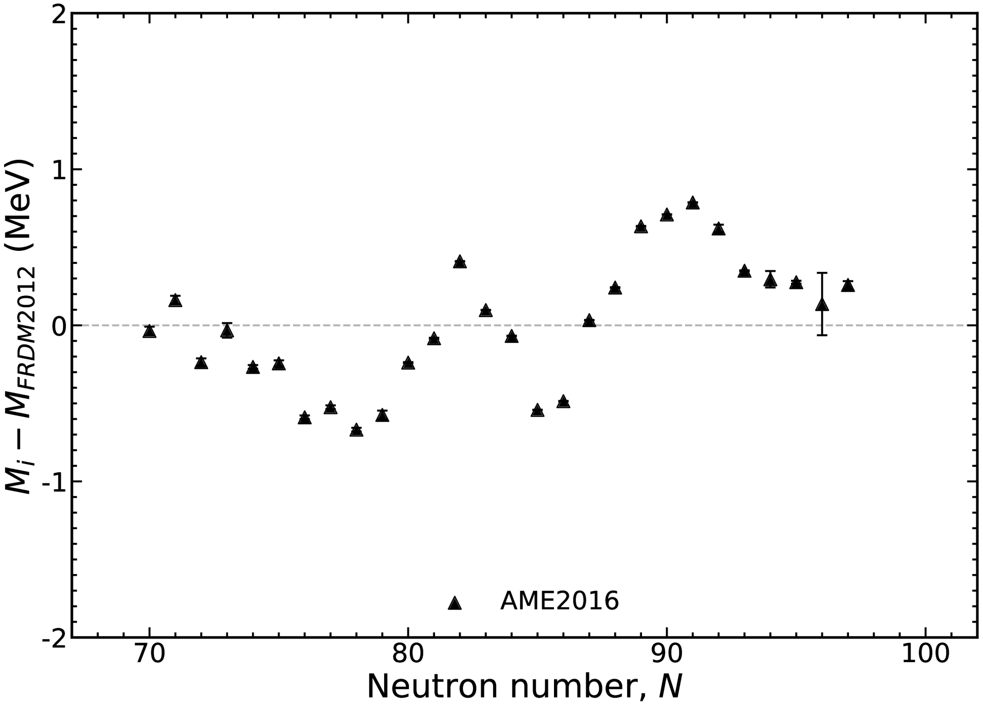


$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_1 + \lambda_{\mathrm{phys}}\mathcal{L}_2\,;\lambda_{\mathrm{phys}} = 1$ in this case

Good match to data does not imply good physics (although by happenstance it is in the end in this case)!

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

MASS TRENDS ALONG ISOTOPIC CHAIN (ND, $Z = 60$)

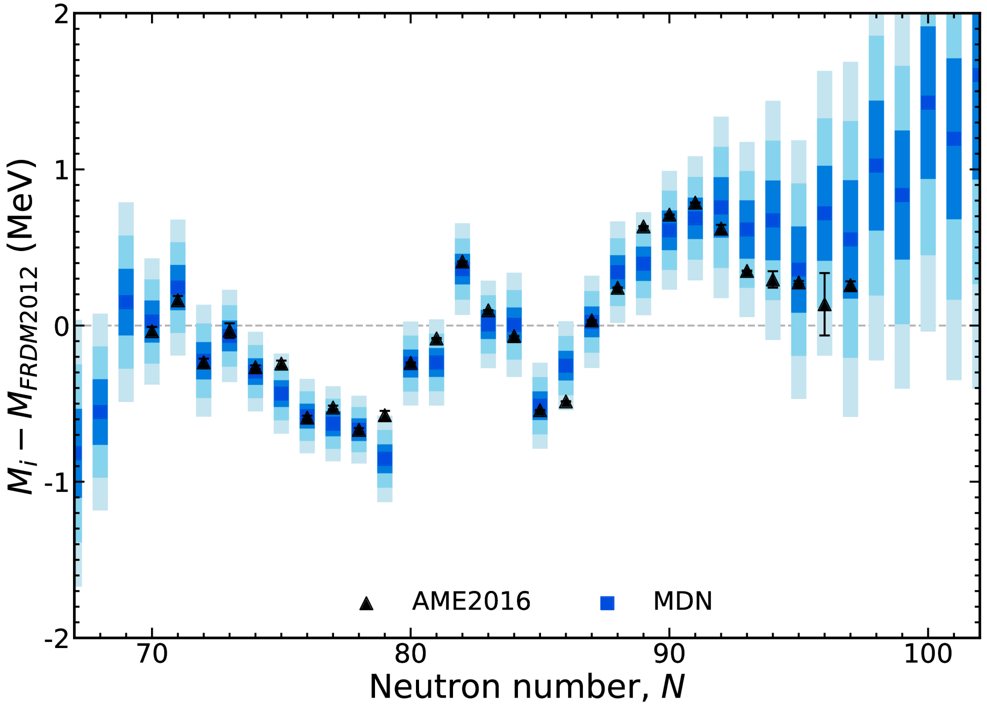


Results plotted relative to the Los Alamos FRDM model

Modern mass models draw a 'straight line' through data

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

RESULTS OF NEW MODEL (ND, $Z = 60$)

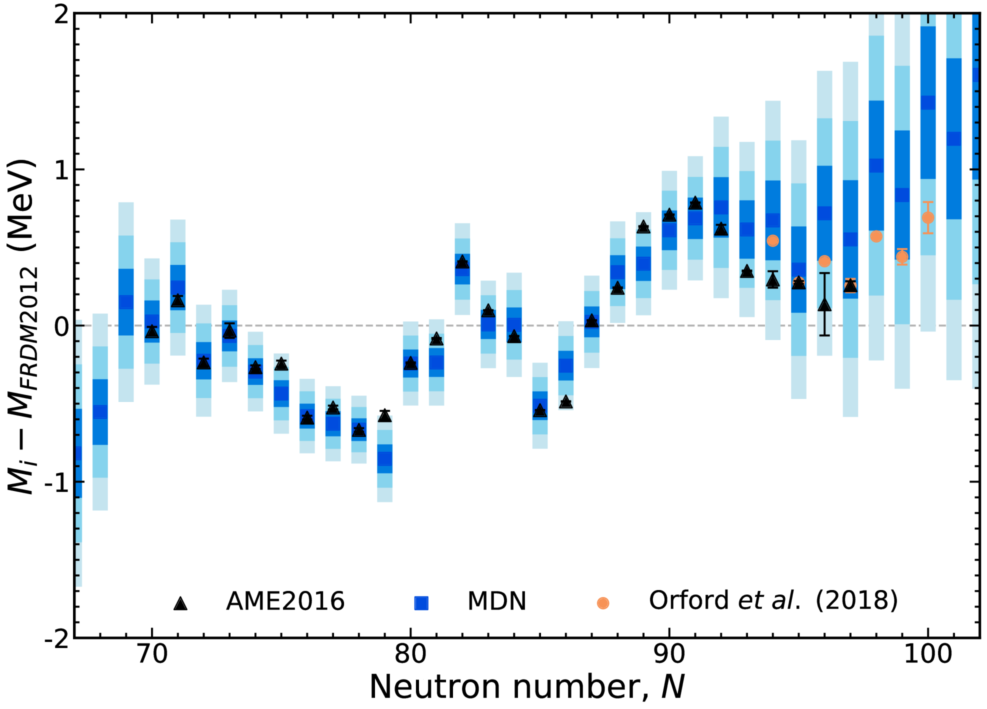


Probabilistic network → well-defined uncertainties predicted for every nucleus (1σ, 2σ 3σ)

Trends in data are captured far better than any mass model

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

RESULTS OF NEW MODEL (ND, $Z = 60$)



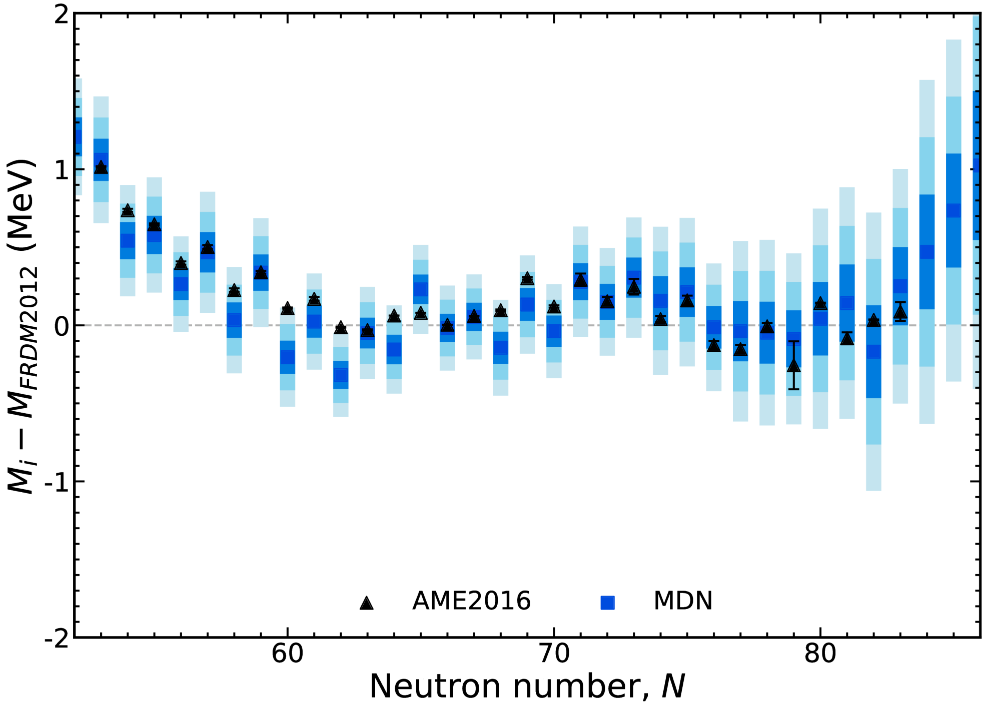
New data (unknown to the model) are in agreement with the MDN model predictions

Model is now trained on only 20% of the AME and predicting 80%!

The training set is <u>random</u>; different datasets perform differently

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

RESULTS OF NEW MODEL (IN, $Z = 49$)



Our results are not limited to any mass region ($Z \geq 20$) - all of them are described equally well

Results are better than any existing mass model; uncertainties inherently increase with extrapolation

We can predict new measurements and capture trends (important for astrophysics)

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

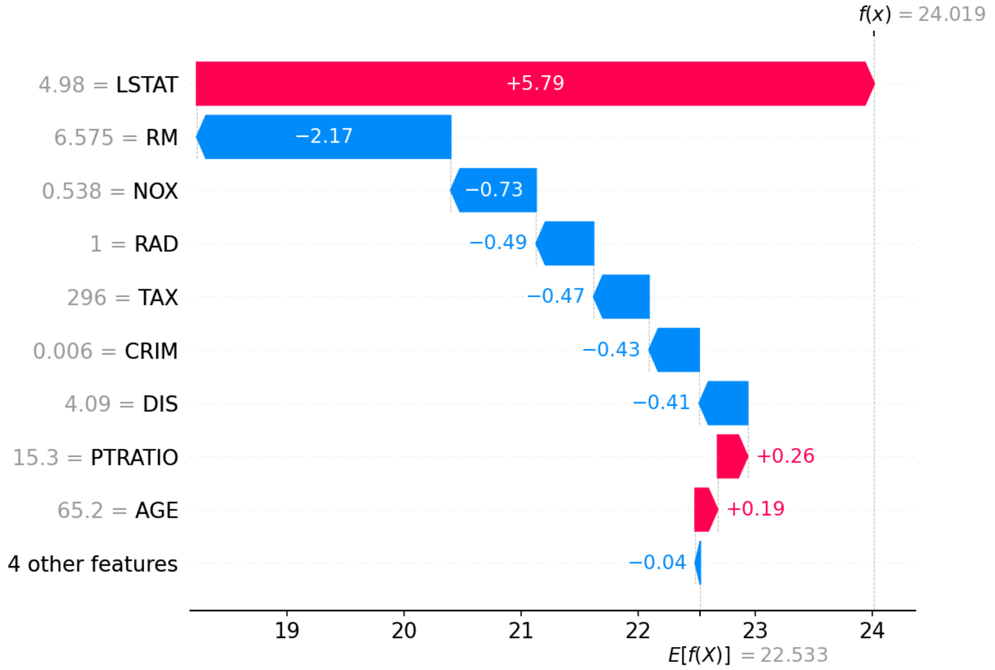
HOW CAN WE INTERPRET THESE RESULTS?

One method is to calculate **Shapley values** (Lundberg & Lee 2017)

Idea arises in Game Theory when a coalition of players (ML - features) optimize gains relative to costs

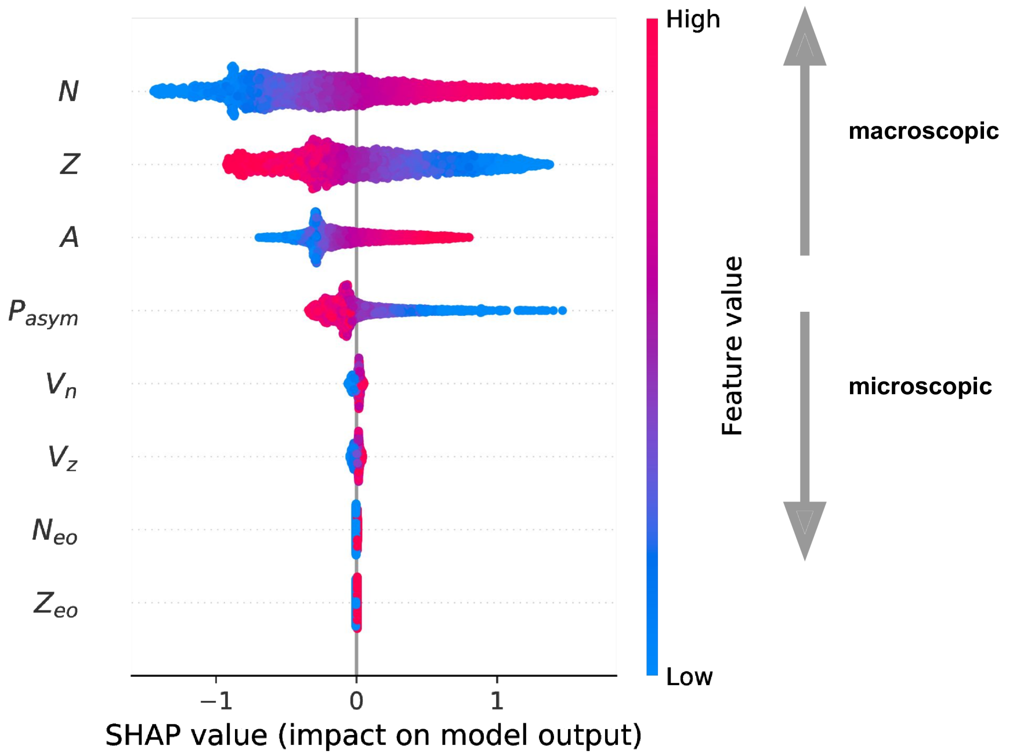
The players may contribute unequally, but the payoff is mutually beneficial

Applicable to many applications, ML in particular to understand feature importance



Lipovetsky, Stan, and Michael Conklin, Applied Stochastic Models in Business and Industry (2001) • Strumbelj, Erik, and Kononenko, Knowledge and information systems (2014) • Figure from SHAP Python package (https://shap.readthedocs.io/)

RANKING FEATURE IMPORTANCE



SHAP calculated for all of the AME (2020) [predictions!]

The model behaves like a theoretical mass model

Macroscopic variables most important followed by quantum corrections

Mumpower, Sprouse, Lovell, & Mohan arXiv:2203.10594 (2022)

My collaborators

A. Lovell, A. Mohan, & T. Sprouse

■ Postdoc

SUMMARY

We have developed an interpretable neural network capable of predicting masses

Advances:

We can predict masses accurately and suss out trends in data

Our network is probabilistic, each prediction has an associated uncertainty

Our model can extrapolate and retains the physics

The quality of the training data is important (not shown)

We can classify the information content of existing and future measurements with this type of modeling

This modeling is general, and can be applied to any nuclear property with sufficient data

Results / Data / Papers @ MatthewMumpower.com