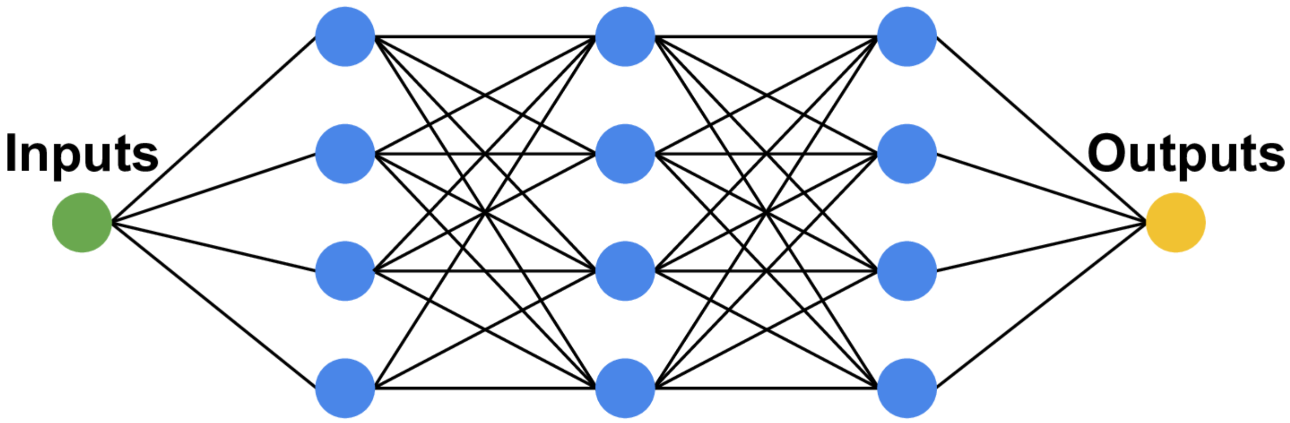
MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE FOR COMPUTATIONAL NUCLEAR DATA



Los Alamos NATIONAL LABORATORY EST. 1943

MATTHEW MUMPOWER

NDT
Nuclear Data Team
Theoretical Division

LA-UR-21-30576

IAEA AI4Atoms Workshop

Monday October 25th 2021

NUCLEAR DATA 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

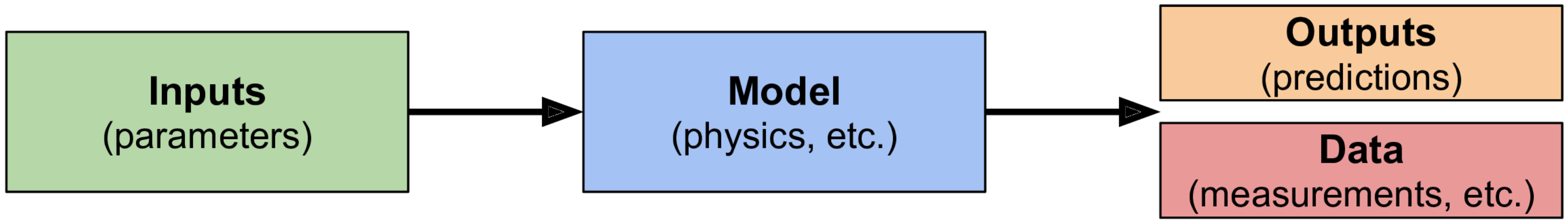
Experiments: backgrounds, isotope production with radioactive ion beams (fragmentation)

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 DATA FROM A FORWARD PROBLEM PERSPECTIVE



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

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

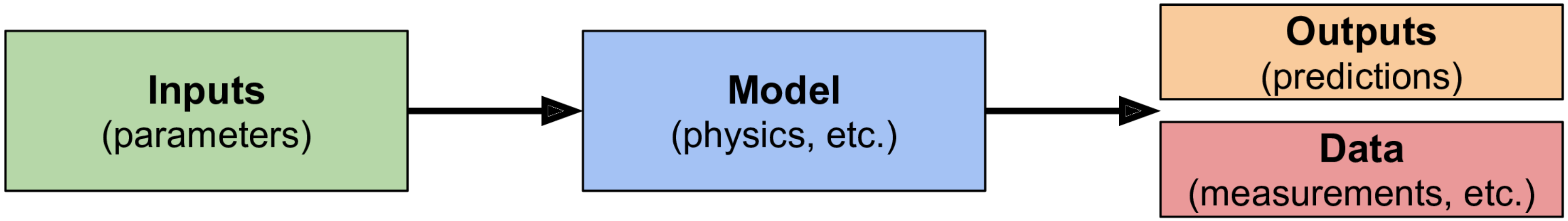
This is an extremely successful approach and is precisely how nuclear theory modeling for nuclear data works

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

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

Figure by **Mumpower**

FORWARD PROBLEM... PROBLEMS...



There can be many challenges with this approach

For instance, how do we update our model if we don't match data?

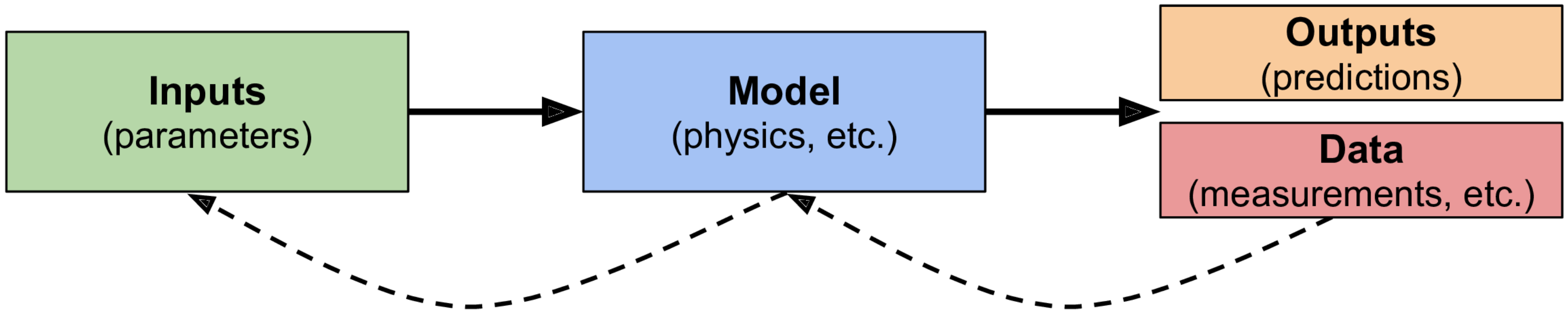
This can be challenging

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 DATA 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)

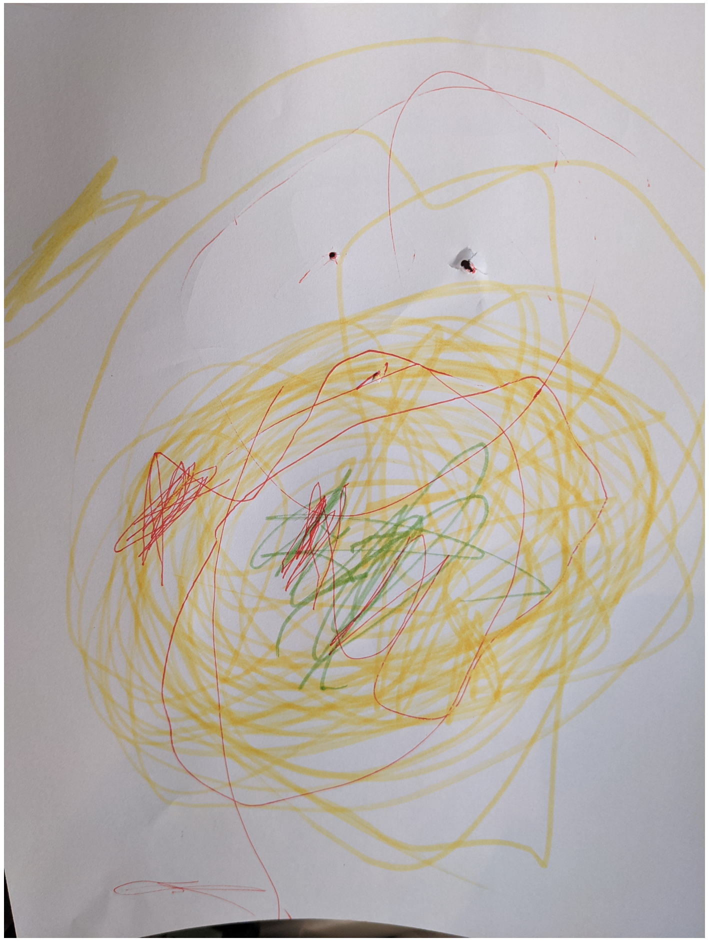
My son Zachary has been very keen on learning his letters from an early age

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'!

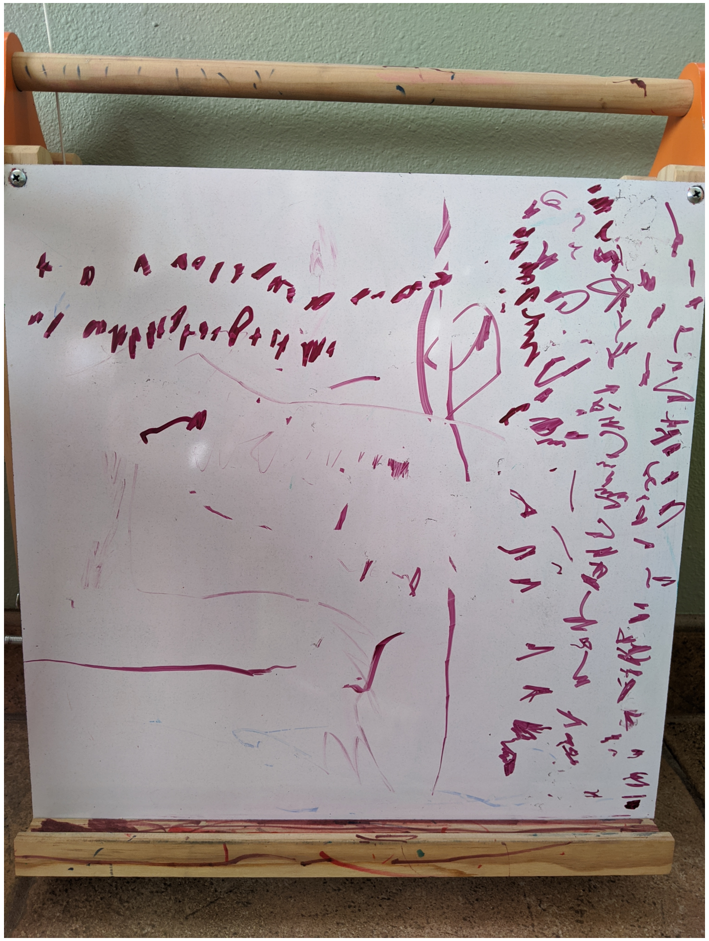
AN EXAMPLE OF HOW THIS WORKS (FROM MY LIFE)



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!

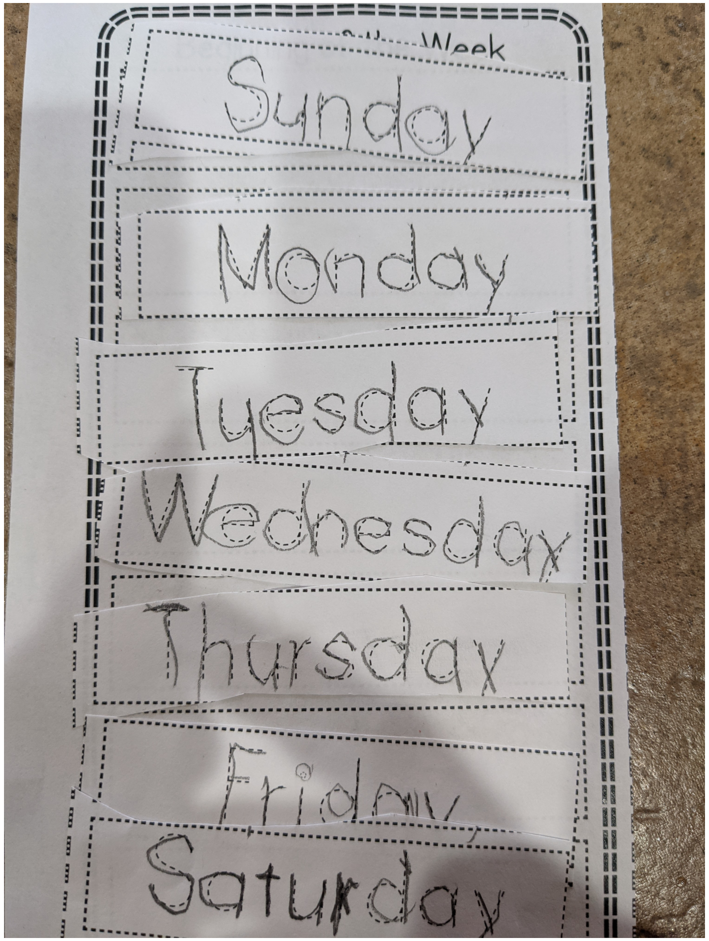
Drawing by Z. R. Mumpower

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 NUCLEAR DATA...

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

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

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

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

But what about $\vec{x}$?

Why not a *physically motivated* feature space!?

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

This is a **completely different approach** than past work

Past work relies on matching model residuals: $\vec{y}$ = (model output) - data

Why not use data as data, rather than mixing model output and data!? (this is difficult to interpret and understand)

Lovell, Mohan, Sprouse & **Mumpower** submitted (2021)

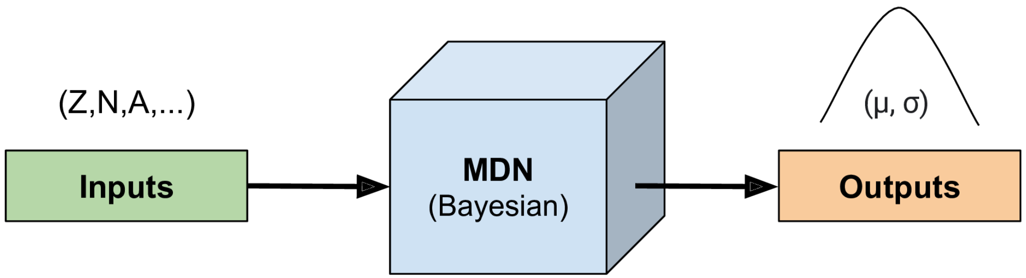
NETWORK FEATURE SPACES ($\vec{x}$)

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** submitted (2021)

MIXTURE DENSITY NETWORK



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

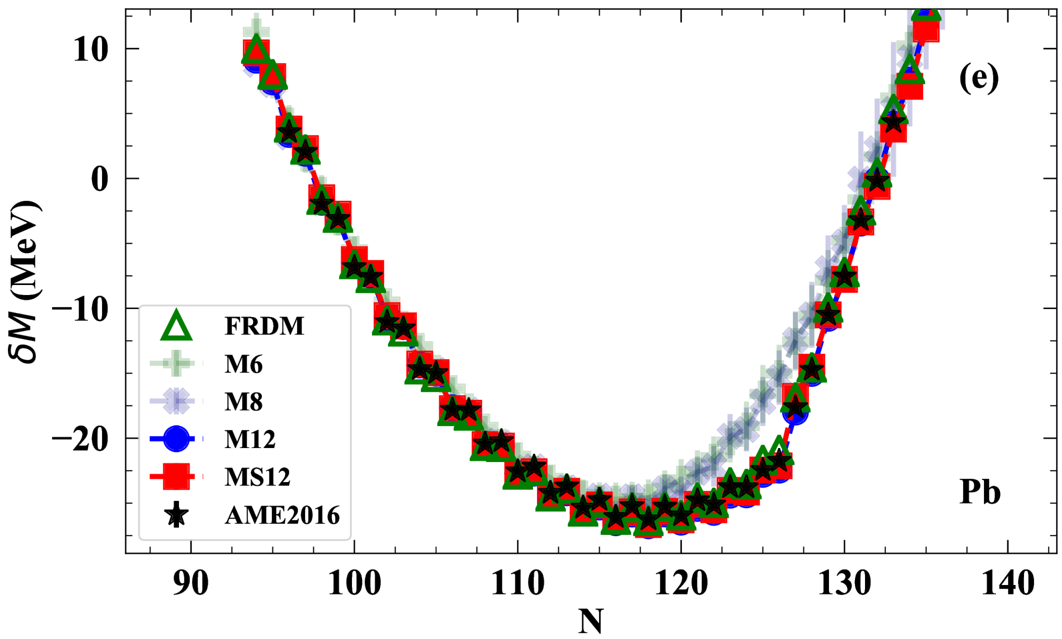
Our network outputs are therefore **statistical**

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

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

Lovell, Mohan, Sprouse & **Mumpower** submitted (2021)

RESULTS: PREDICTING ATOMIC MASSES



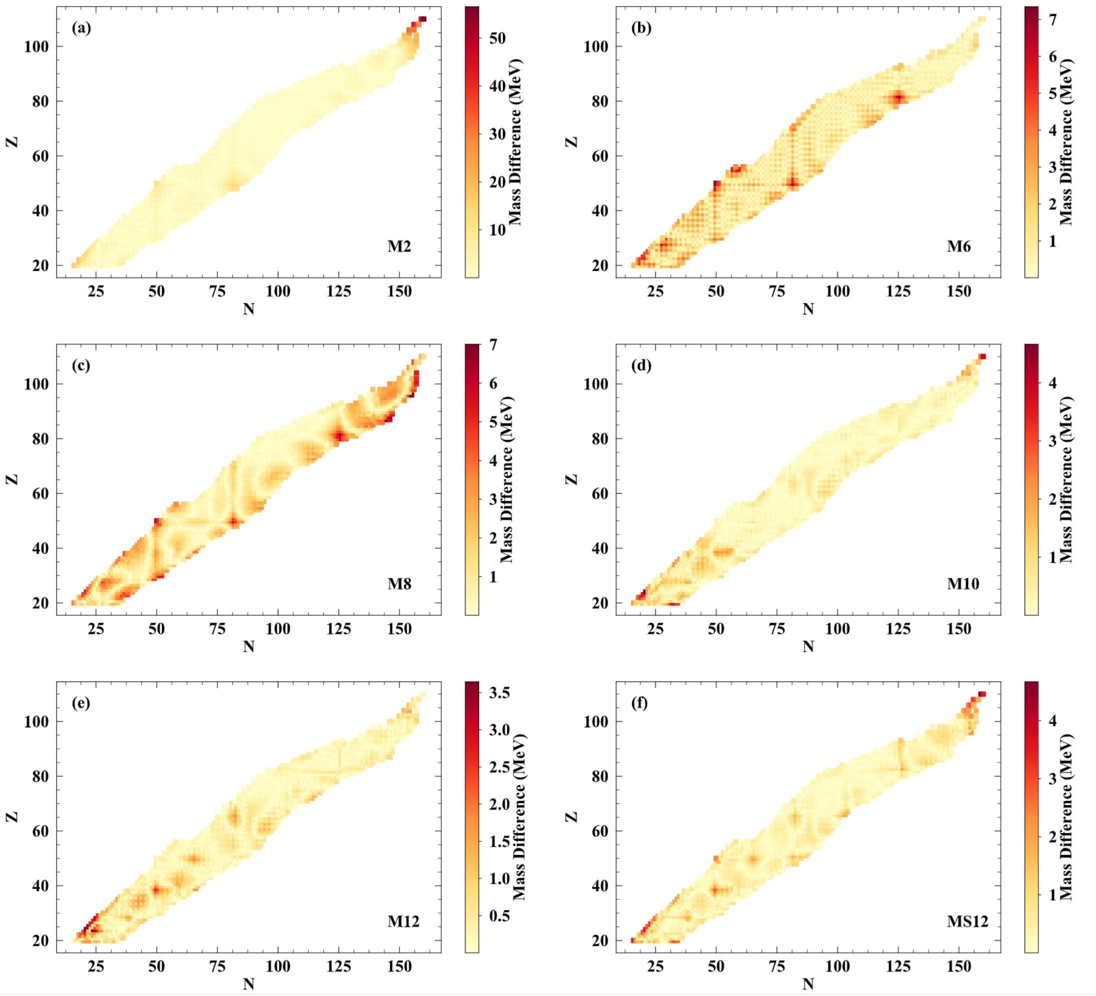
Here we show results along the lead (Pb) $[Z = 82]$ isotopic chain

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

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

Lovell, Mohan, Sprouse & **Mumpower** submitted (2021)

RESULTS: PREDICTING 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** submitted (2021)

RESULTS: PREDICTING 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)

We do not need deep neural networks to describe atomic masses; nor for extrapolations!

Lovell, Mohan, Sprouse & **Mumpower** submitted (2021)

My collaborators

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

▣ Postdoc

SUMMARY

We are entering the era of *computational* nuclear data

Recent advances:

Los Alamos is developing a state-of-the-art computational nuclear data framework

This framework can combine measurements, observations and theoretical modeling to produce nuclear data with well-defined uncertainties that can be easily interpreted and shared with the community

As an example, we showed in this talk binding energies of atomic nuclei with a Bayesian neural network

Our methods are general and can be applied to any physical quantity or system

Results / Data / Papers @ MatthewMumpower.com