Machine Learning Surrogates for Rapid Dark Matter Direct Detection Calculations

Dorian Amaral[1]
D. G. Cerdeño[2]   A. Cheek[3]   H. Schulz[1]

YTF20 December 16, 2020

[1]Durham University [2]Universidad Autónoma de Madrid
[3]Université catholique de Louvain

Machine learning models can be powerful performance-boosting tools

The Story

1. Setting The Scene

   Dark Matter

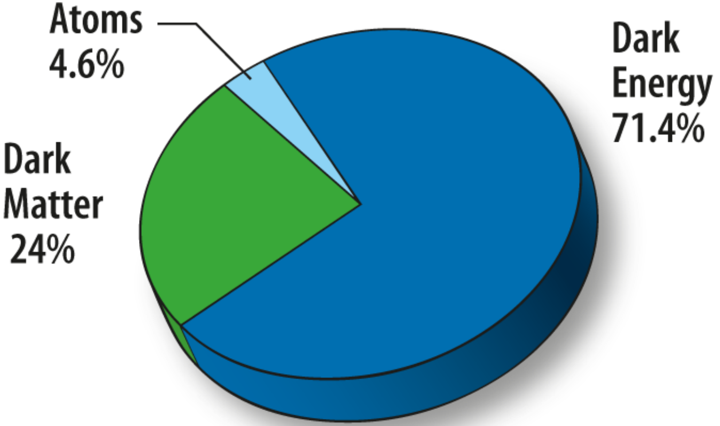
   Direct Detection

2. The Parameter Space is Big and Expensive

3. Surrogate Models can Dramatically Boost Performance

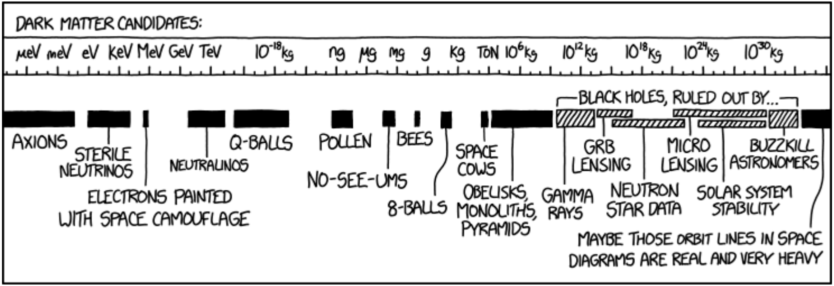
4. Machine Learning Models can be Effective Surrogates

Setting The Scene

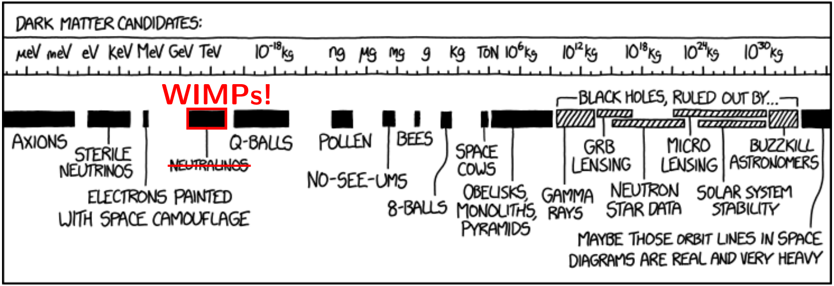


TODAY

Credit: *NASA/ESA*

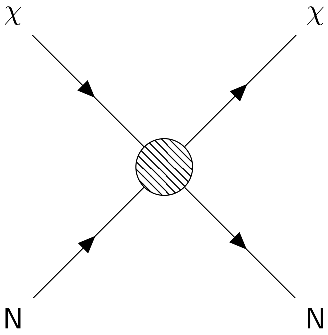

DARK MATTER CANDIDATES:
μeV meV eV KeV MeV GeV TeV 10^{-18}kg ng μg mg g Kg TON 10^6kg 10^{12}kg 10^{18}kg 10^{24}kg 10^{30}kg
BLACK HOLES, RULED OUT BY...
AXIONS
STERILE NEUTRINOS
ELECTRONS PAINTED WITH SPACE CAMOUFLAGE
NEUTRALINOS
Q-BALLS
POLLEN
NO-SEE-UMS
BEES
8-BALLS
SPACE COWS
OBELISKS, MONOLITHS, PYRAMIDS
GAMMA RAYS
GRB LENSING
NEUTRON STAR DATA
MICRO LENSING
SOLAR SYSTEM STABILITY
BUZZKILL ASTRONOMERS
MAYBE THOSE ORBIT LINES IN SPACE DIAGRAMS ARE REAL AND VERY HEAVY

Credit: *xkcd*



Credit: *xkcd + Me*

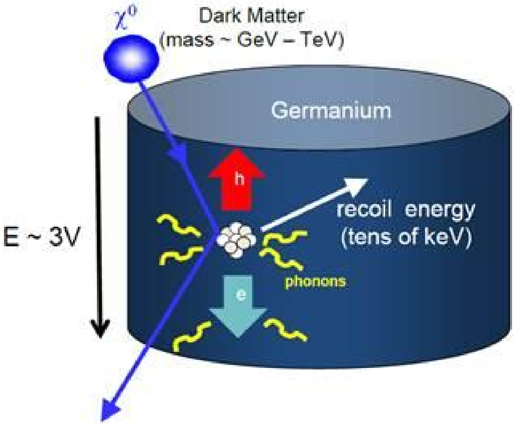
DM Scattering in a Direct Detector



χ = A BSM particle with $\sigma \sim$ weak interaction

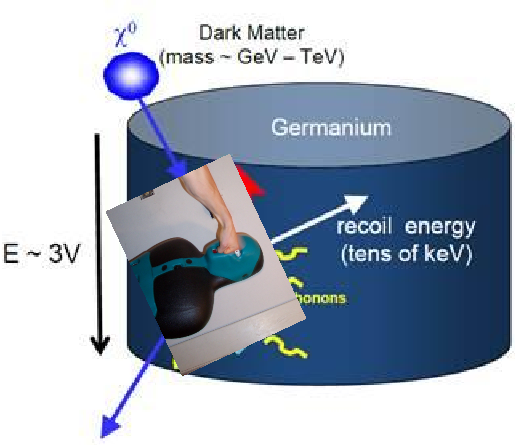
N = p, n

DM Scattering in a Direct Detector



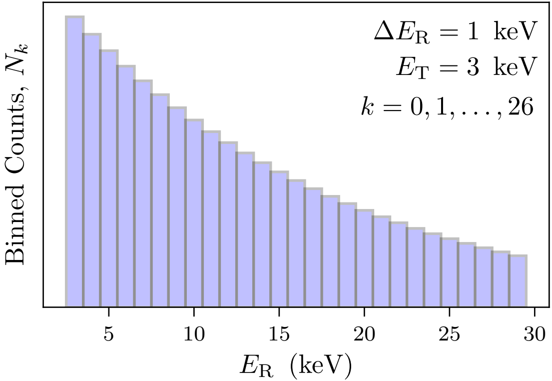
Credit: *SuperCDMS Collaboration*

DM Scattering in a Direct Detector



Credit: *SuperCDMS Collaboration + Wikipedia*

DM Scattering in a Direct Detector



$$\Updownarrow$$

Parameter space point $\vec{\Theta} = (m_\chi, \vec{c})$

Given the data from all bins, what are the DM parameters; how massive is it and how much of a punch does it pack?

The Direct Detection Parameter Space is Big and Expensive

The Count Calculation

$$N_k(\vec{\Theta}) \propto \underbrace{\boxed{\int_{E_k}^{E_{k+1}} dE_\mathrm{R}\, \varepsilon(E_\mathrm{R})}}_{\text{Binning}} \underbrace{\boxed{\int_{E'_\mathrm{R}} dE'_\mathrm{R}\, \mathrm{Gauss}(E'_\mathrm{R}, E_\mathrm{R})}}_{\text{Experimental Smearing}} \underbrace{\boxed{\int_{v_\mathrm{min}} d^3\vec{v}\, v\, f(\vec{v})\, \frac{d\sigma_{\chi T}}{dE'_\mathrm{R}}}}_{\text{Important Physics: Astro + Particle}}$$

$$\mathcal{L}(\vec{\Theta}) = \prod_k \mathrm{Poisson}(k^\mathrm{th}\ \text{Bin Data}; N_k)$$

The Hint of a Problem

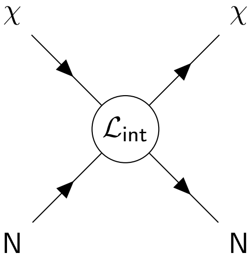
- Want to maximise $\mathcal{L}$ to get the best-fit point $\hat{\vec{\Theta}}$.
- Means evaluating it many thousands of times.
- **However, N_k is looking pretty expensive!**

Dark Matter: The EFT

$$\mathcal{L}_{\text{int}} = \sum_{N=p,n} \sum_i c_i^N \mathcal{O}_i \chi \overline{\chi} N \overline{N}$$

$$\Downarrow$$

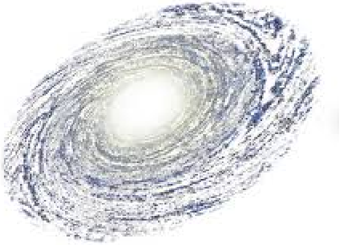
$$\frac{d\sigma_{\chi T}}{dE_{\text{R}}'} \propto \sum_{N,N'} \sum_{i,j} \underbrace{\boxed{c_i^N}}_{\vec{c} \in \vec{\Theta}} \; c_j^{N'} \underbrace{\boxed{F_{i,j}^{N,N'}}}_{\text{Interf.}}$$



Generally, can have 11 $\mathcal{O}$'s!

Dark Matter: The Halo

$f(\vec{v})$ = **Choice** of DM Halo function



Have nuissance parameters too:

- Dark matter density
- Mean velocity in halo
- Escape velocity of MW
- And possibly more...!

Direct Detection in Full

So, to recap:

- Given a dark matter signal, want to find best-fit point $\hat{\vec{\Theta}}$
- Means traversing parameter space, calculating...
- ... a three-nested integral (possibly for different targets)
- Generally, $\vec{\Theta} = (m_\chi, c_1, \ldots, c_{11})$
- Marginalising over all astrophysics

The direct detection parameter space is big and expensive to traverse!

Surrogate Models can Dramatically Boost Performance

Surrogate Models: What are They?

Simple Idea

Replace expensive function with cheap function which mimics the costly function:

$$N_k(\vec{\Theta}) \to f_k(\vec{\Theta}) \simeq N_k(\vec{\Theta})$$

Surrogate Models: How to Build a Mimic

1. Pick a surrogate model, f_k.
2. Pick some $\vec{\Theta}$'s in the parameter space.
3. For each $\vec{\Theta}$ and bin, k, calculate the true count, $N_k(\vec{\Theta})$.
4. *'Train'* f_k on this dataset.
5. Use f_k to make all future predictions quickly!

RAPIDD: Polynomial Surrogates

Simple and effective: Polynomials (RAPIDD[1])

$$f_k(\vec{\Theta}) \equiv \mathcal{P}_k(\vec{\Theta}) \equiv \sum_{n=1}^{N_{\text{coeff}}} d_{k,n}\tilde{\Theta}_n$$

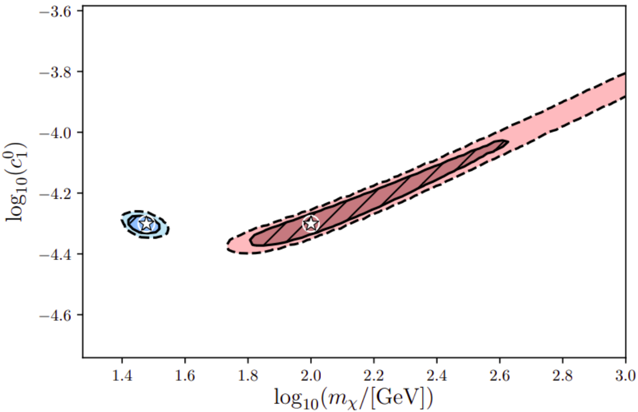
$d_{k,n}$ = coefficients

$\tilde{\Theta}$ = parameter combinations to make desired polynomial order

Training means finding those $d_{k,n}$ which best fit N_k.

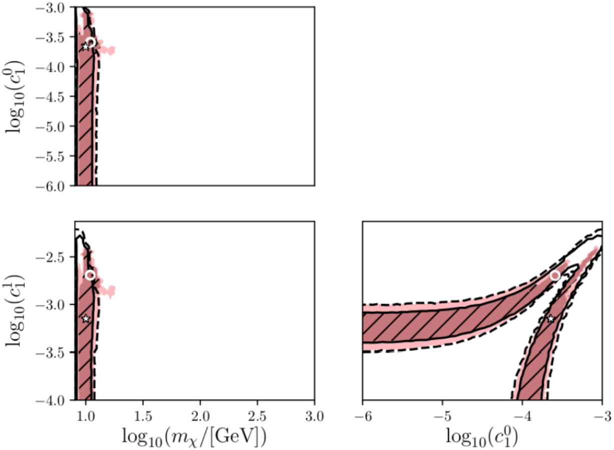
[1] Cerdeño et al., "Surrogate models for direct dark matter detection".

RAPIDD: 2D Example



Stars are best fit points for full physics code (**40 mins**). Circles are best fit points for RAPIDD (**10 s**). Contours: 1σ and 2σ confidence regions. See [1].

RAPIDD: 3D Example (Isospin violation)



Interference terms not as well modelled by polynomials, but still pretty good and definitely faster. See [1].

(Polynomial) surrogate models can dramatically boost performance while still remaining very accurate

A couple of polynomial downfalls (See [1]):

- Need special treatment to deal with interference terms.
- Don't handle discontinuities well (sudden changes to counts).
- Precision loss at very high dimensionalities.

What other surrogate models, f_k, can we pick?

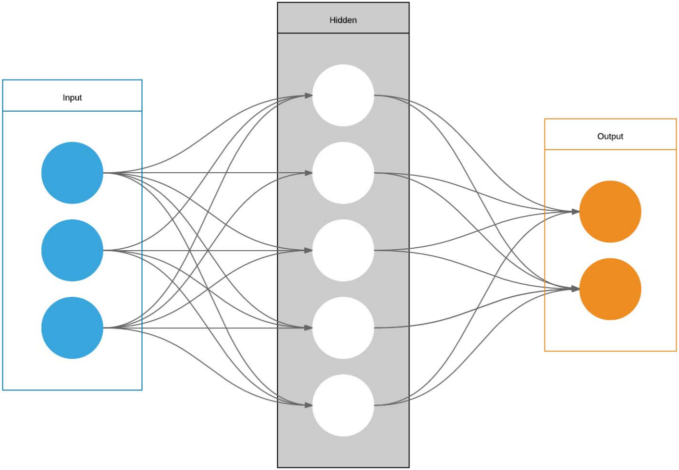
Machine Learning Models can be Effective Surrogates

What is a Machine Learning Model?

My Quick Definition

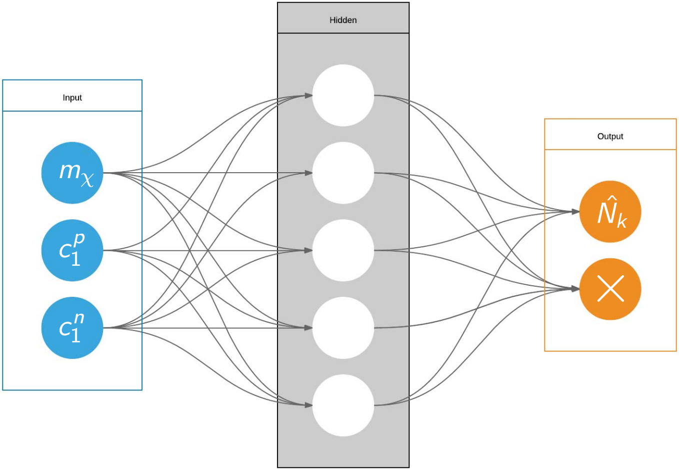
An algorithmic model which attempts to learn from data to minimise how bad it is at predicting future values.

Given a set of 'features', $\vec{\Theta}$, learn the mapping f_k which gives the target values N_k by minimising some loss function (eg MSE).



Credit: *Wikipedia*

300 neurons $\times$ 4 hidden layers



Credit: *Wikipedia* + *Me*

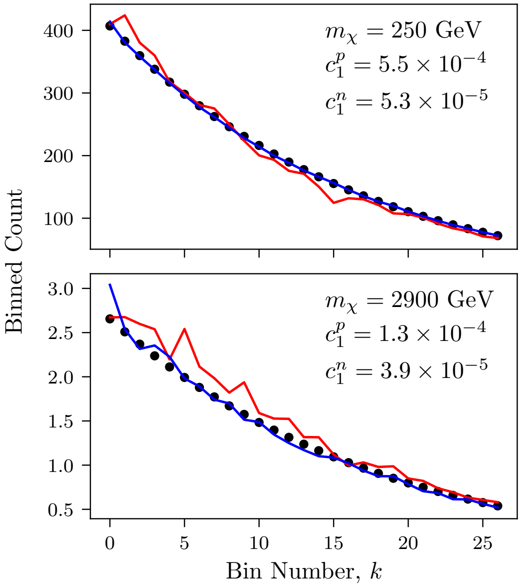
A Proof of Concept

Compare: f_k =polynomials with f_k = DNNs

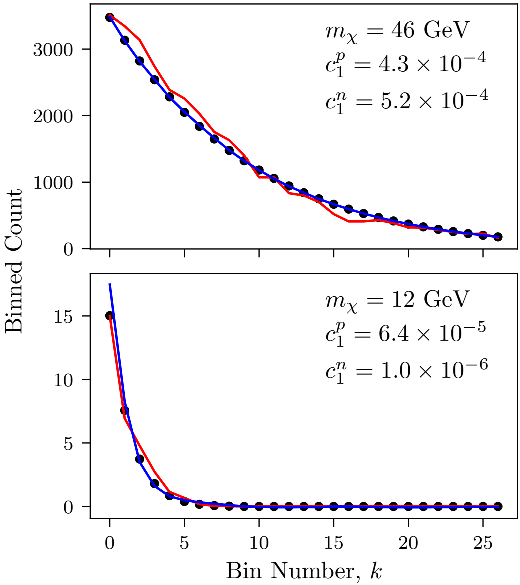
Total number of data points ($\vec{\Theta}$, true N_k): 1900

Train on 90% of data

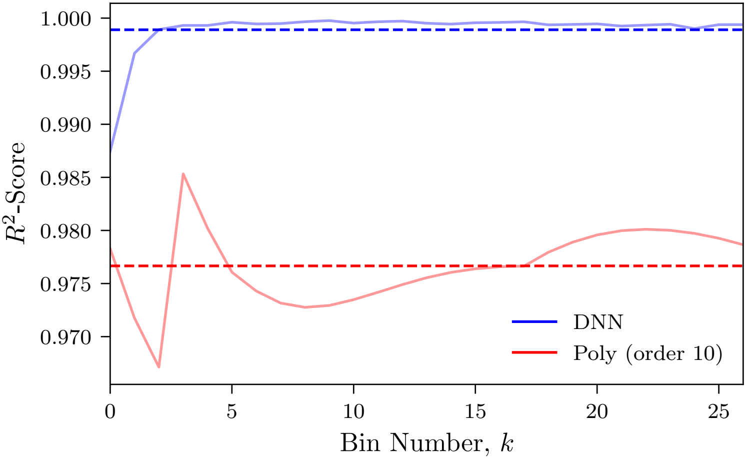
Test on remaining 10%



True DNN Poly (order 10)



True DNN Poly (order 10)



R^2-score = Fraction of data explained by model within 1σ

Evaluation times for 10000 points:

Polynomials $\sim$ 10 s DNN $\sim$ 25 s

DNN and Polynomial times comparable, and DNNs looking more accurate

So now...

What do the reconstructed best-fit points and confidence-regions look like?

The Take-Home Messages

1. Invariably, we meet big, expensive functions.
2. Surrogate models can dramatically boost performance at evaluation time.
3. Machine learning models can be very effective surrogates!

Machine learning models can be powerful performance-boosting tools