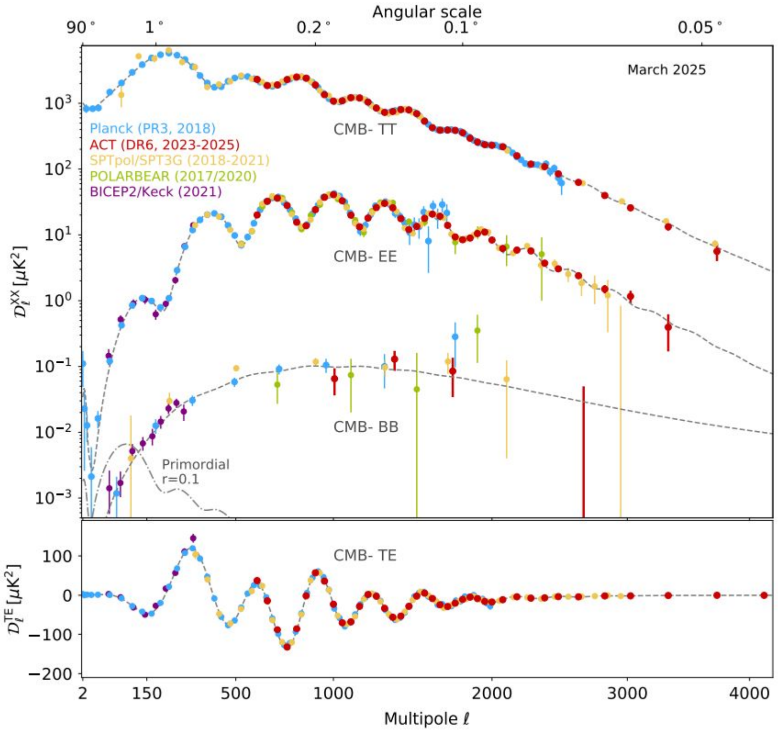
Accelerating Modern Cosmology with Symbolic Regression

David M.J. Vokrouhlicky in collaboration with C. Skordis, D.Bartlett, P.Ferreira, H.Desmond

Co-funded by the European Union

MINISTRY OF EDUCATION, YOUTH AND SPORTS

FORTE



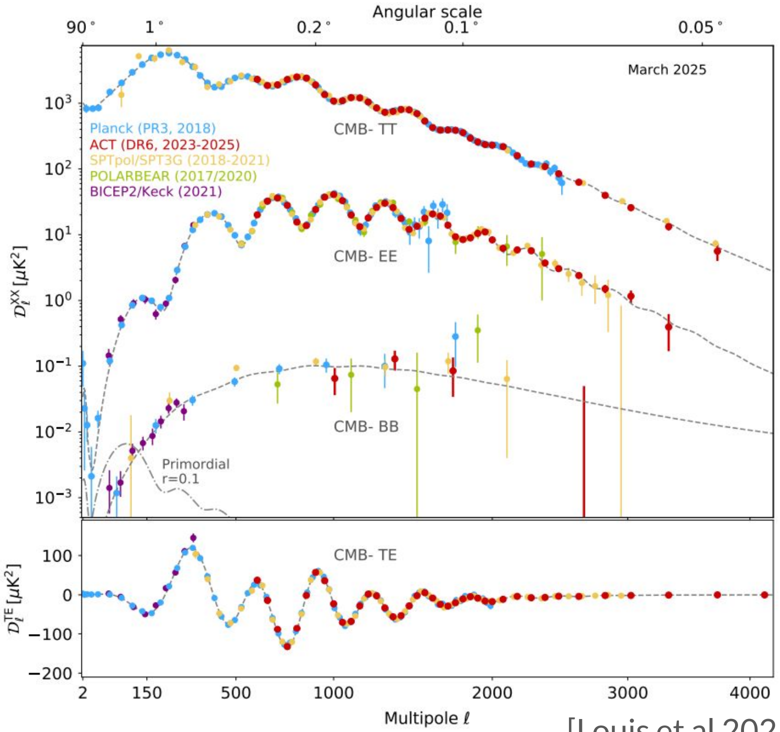
[Louis et al 2025]

$\longrightarrow$

$$
\begin{aligned}
\Omega_m &= \text{total matter density} \\
\Omega_b &= \text{baryonic matter density} \\
\Omega_\Lambda &= \text{dark energy density} \\
H_0 &= \text{the Hubble parameter} \\
\tau &= \text{the optical depth} \\
n_s &= \text{spectral index of initial spectrum} \\
A &= \text{amplitude of initial spectrum}
\end{aligned}
$$

$$
\begin{aligned}
\Omega_m &= \text{total matter density}\\
\Omega_b &= \text{baryonic matter density}\\
\Omega_\Lambda &= \text{dark energy density}\\
H_0 &= \text{the Hubble parameter}\\
\tau &= \text{the optical depth}\\
n_s &= \text{spectral index of initial spectrum}\\
A &= \text{amplitude of initial spectrum}
\end{aligned}
$$

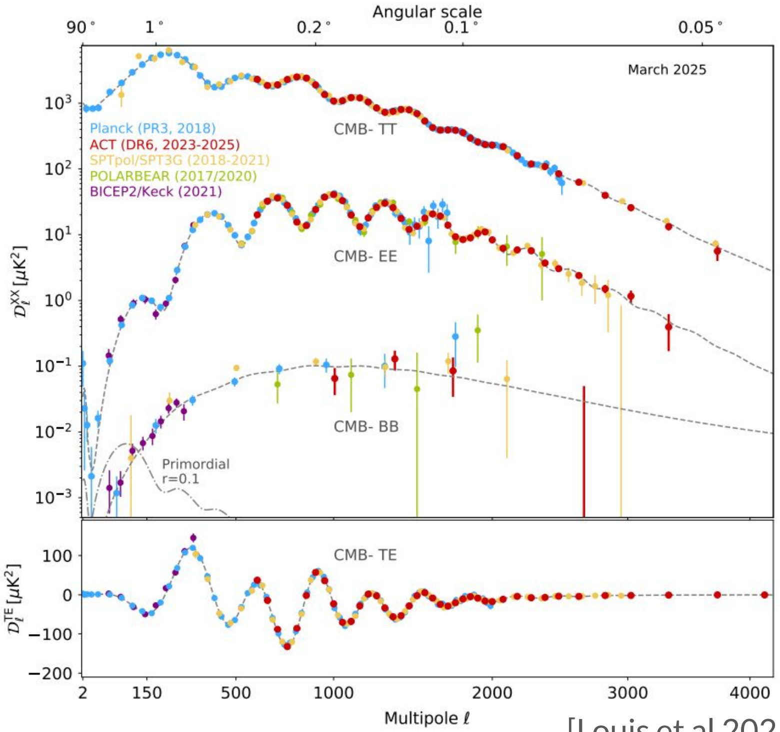
$\longrightarrow$



[Louis et al 2025]

$$\Omega_m = \text{total matter density}$$
$$\Omega_b = \text{baryonic matter density}$$
$$\Omega_\Lambda = \text{dark energy density}$$
$$H_0 = \text{the Hubble parameter}$$
$$\tau = \text{the optical depth}$$
$$n_s = \text{spectral index of initial spectrum}$$
$$A = \text{amplitude of initial spectrum}$$

$\longrightarrow$



[Louis et al 2025]

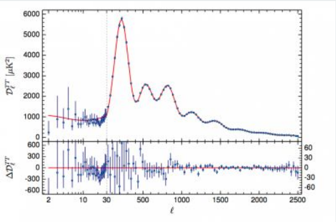
MCMC requires $O(10^5)$ evaluations + Boltzmann codes take order 1min to calculate observables $\Rightarrow$ Faster simpler way to calculate observables needed

CMB anisotropies

Boltzmann codes [Lesgourges et al, Lewis et al]

$A_s\ \Omega_m\ \Omega_b\ h\ n_s$

Class



σ_8, …

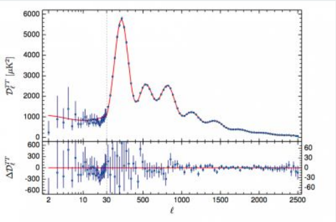
Precise, but slow for inference

CMB anisotropies

Boltzmann codes [Lesgourges et al, Lewis et al]

$A_s\ \Omega_m\ \Omega_b\ h\ n_s$

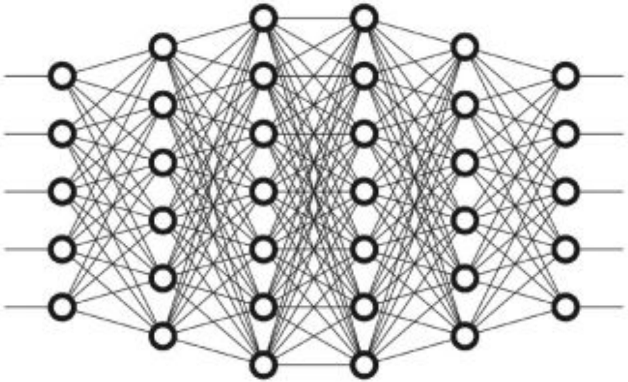
Class



σ_8, …

Precise, but slow for inference

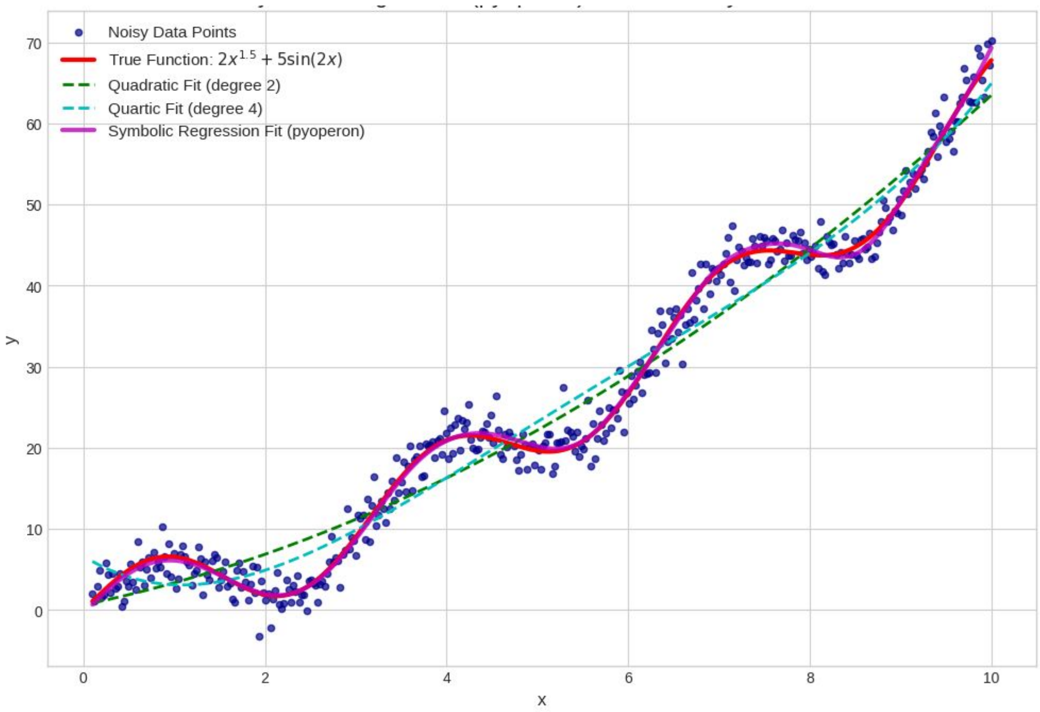
Neural Network emulators [Bolliet et al]



Faster, but do not have CPL+Σm_ν, harder to maintain

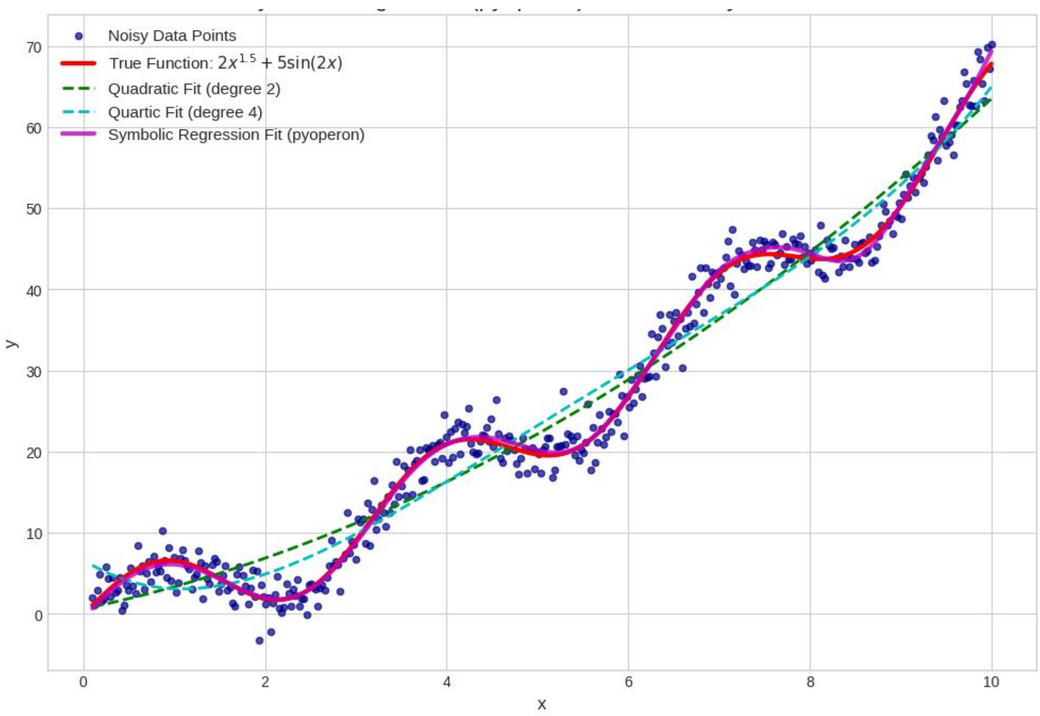
Symbolic regression

- Discover general analytical functions instead of fitting parameters in fixed ones



Symbolic regression

- Discover general analytical functions instead of fitting parameters in fixed ones



True function: $2x^{1.5} + 5\sin(2x)$

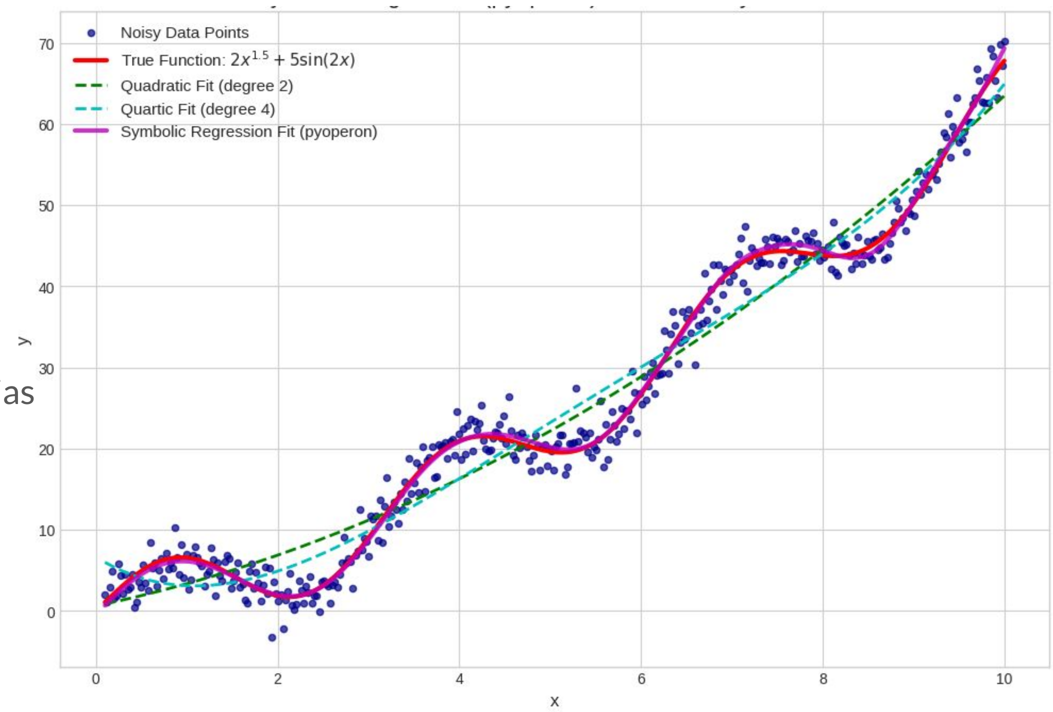
Quadratic Fit: $y = 0.396x^2 + 2.332x + 0.617$

Symbolic fit: (0.256 + (1.257 * ((sin((1.997 * X1)) - ((-1.354) * X1)) ^ 1.504)))

Symbolic regression

Benefits

- Simple to use and maintain
- Potential for interpretability
- More general than fitting - no confirmation bias
- Can outperform Neural Networks in small training datasets



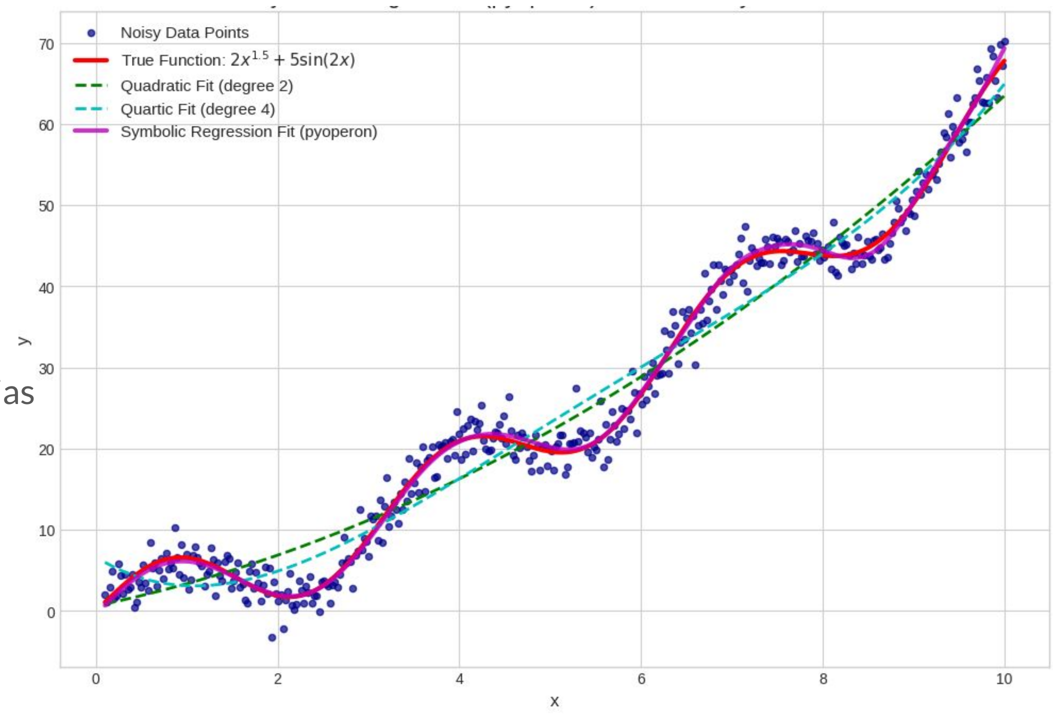
Symbolic regression

Benefits

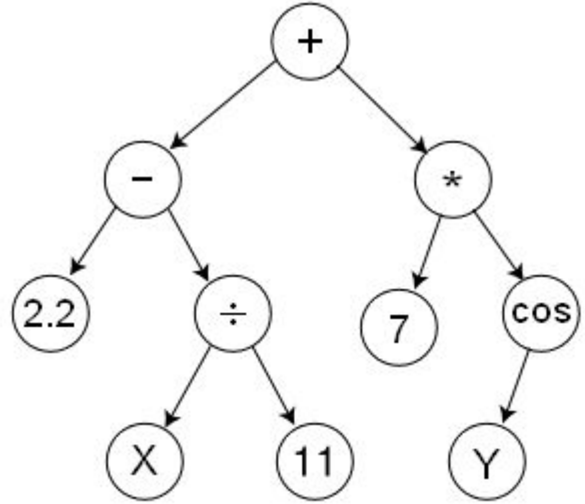
- Simple to use and maintain
- Potential for interpretability
- More general than fitting - no confirmation bias
- Can outperform Neural Networks in small training datasets

Caveats:

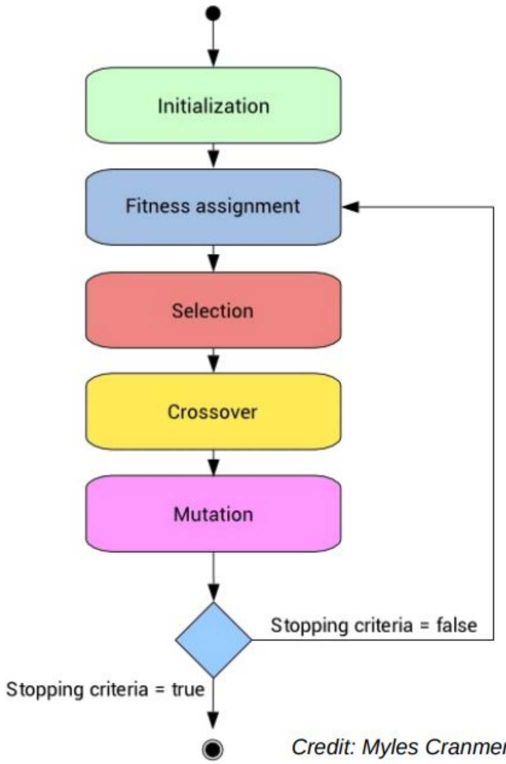
- Bad scaling with search space, dataset size
- Incompatible with fast optimization methods
- Usually worse fit than full NN

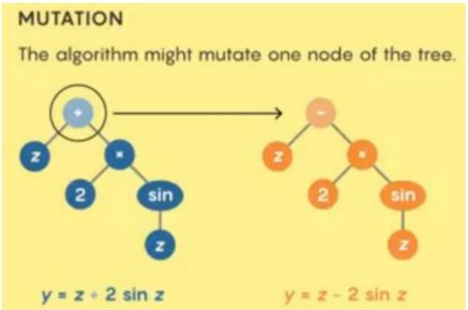


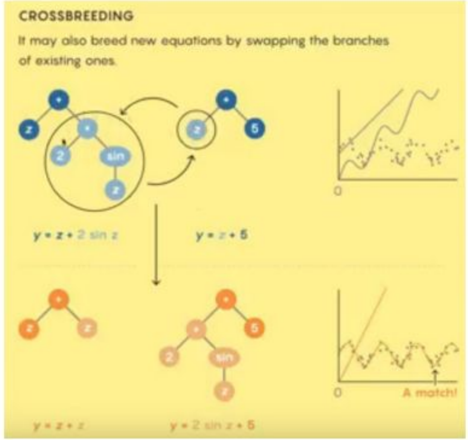
In practice



$$\left(2.2 - \left(\frac{X}{11}\right)\right) + \left(7 * \cos(Y)\right)$$



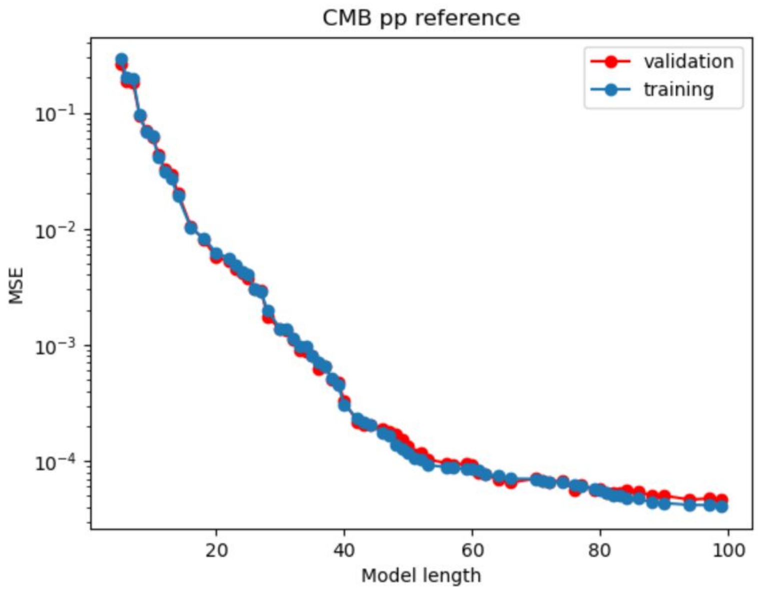




Credit: Myles Cranmer

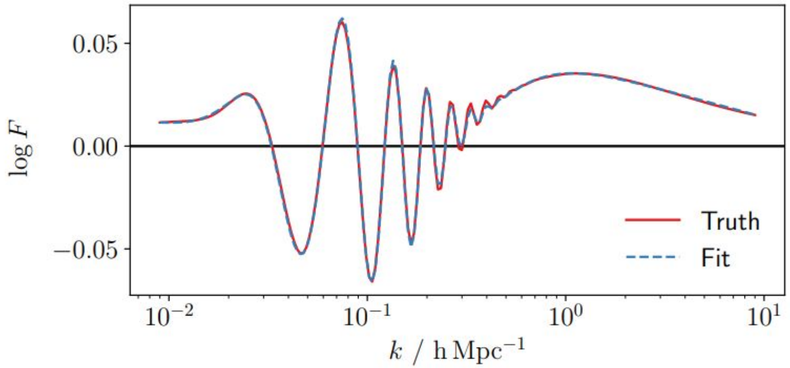
SR as two parameter optimization

- There are always very complex functions that can have zero error
  - **Optimize for both simplicity and accuracy**
- Test training and validation
- Result - **Pareto front** of optimal models for each complexity
- Choose based on desired error and expression



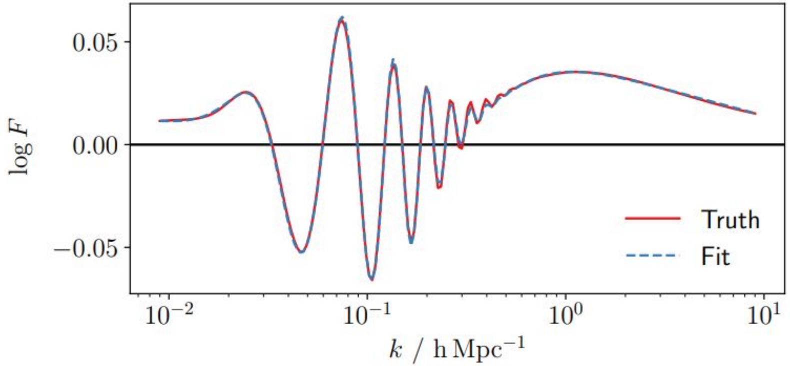
Main use-cases of SR

Full emulation of observables [Bartlett et al]



Main use-cases of SR

Full emulation of observables [Bartlett et al]



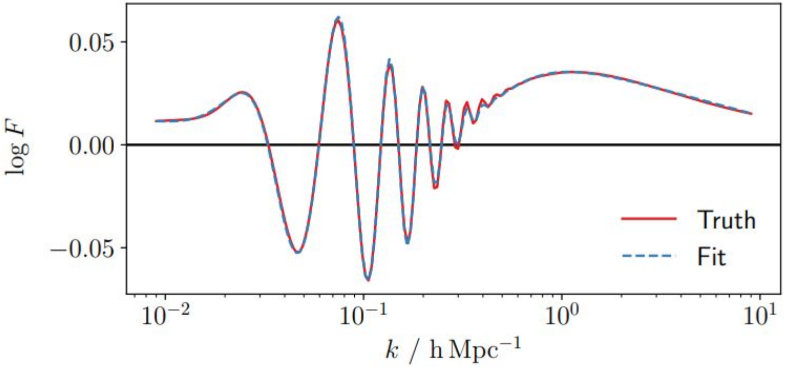
Emulating parts of a numerically expensive calculation [Farakou et al]

$$P_{1,3}(\vec{k},a)\delta_D(\vec{k}+\vec{k'}) =< \delta_1(\vec{k},a),\delta_3(\vec{k'},a) >=$$
$$= \int \frac{d^3q_1}{(2\pi)^3}\frac{d^3q_2}{(2\pi)^3}\frac{d^3q_3}{(2\pi)^3}\delta_D^3(\vec{k'}-\vec{q}_1-\vec{q}_2-\vec{q}_3)$$
$$\times\, Complicated - Kernel(\vec{q}_1,\vec{q}_2,\vec{q}_3)$$
$$\times < \delta_1(\vec{k},a)\delta_1(\vec{q}_1,a)\delta_1(\vec{q}_2,a)\delta_1(\vec{q}_3,a) >$$

$$\boxed{\text{Symbolic Regression} \rightarrow P^{X-loop}_{complicated-integral}(k, A_s, n_s, \Omega_{CDM}\Omega_b, h)}$$

Main use-cases of SR

Full emulation of observables [Bartlett et al]

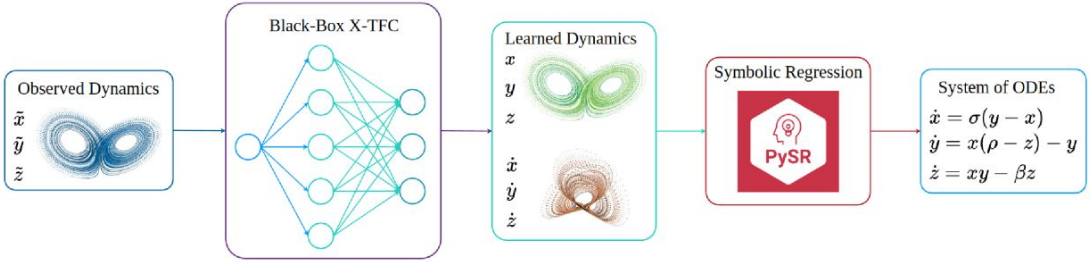


Emulating parts of a numerically expensive calculation [Farakou et al]

$$P_{1,3}(\vec{k},a)\delta_D(\vec{k}+\vec{k'}) =< \delta_1(\vec{k},a), \delta_3(\vec{k'},a) >=$$
$$= \int \frac{d^3q_1}{(2\pi)^3}\frac{d^3q_2}{(2\pi)^3}\frac{d^3q_3}{(2\pi)^3}\delta_D^3(\vec{k'}-\vec{q}_1-\vec{q}_2-\vec{q}_3)$$
$$\times\ Complicated - Kernel(\vec{q}_1,\vec{q}_2,\vec{q}_3)$$
$$\times < \delta_1(\vec{k},a)\delta_1(\vec{q}_1,a)\delta_1(\vec{q}_2,a)\delta_1(\vec{q}_3,a) >$$

$$\text{Symbolic Regression} \rightarrow P^{X-loop}_{complicated-integral}(k, A_s, n_s, \Omega_{CDM}\Omega_b, h)$$

Finding Fully New Physical laws [De Florio et al]



Our applications:

1. **CMBolic**: suite of CMB anisotropy emulators
2. Matter power spectra in the **Generalized Dark Matter** framework

CMBolic

- **Fully analytic** emulation of CMB lensing, temperature and polarization anisotropies
- Trained on hundreds of cosmologies from Class with Act DR6 inspired accuracy settings

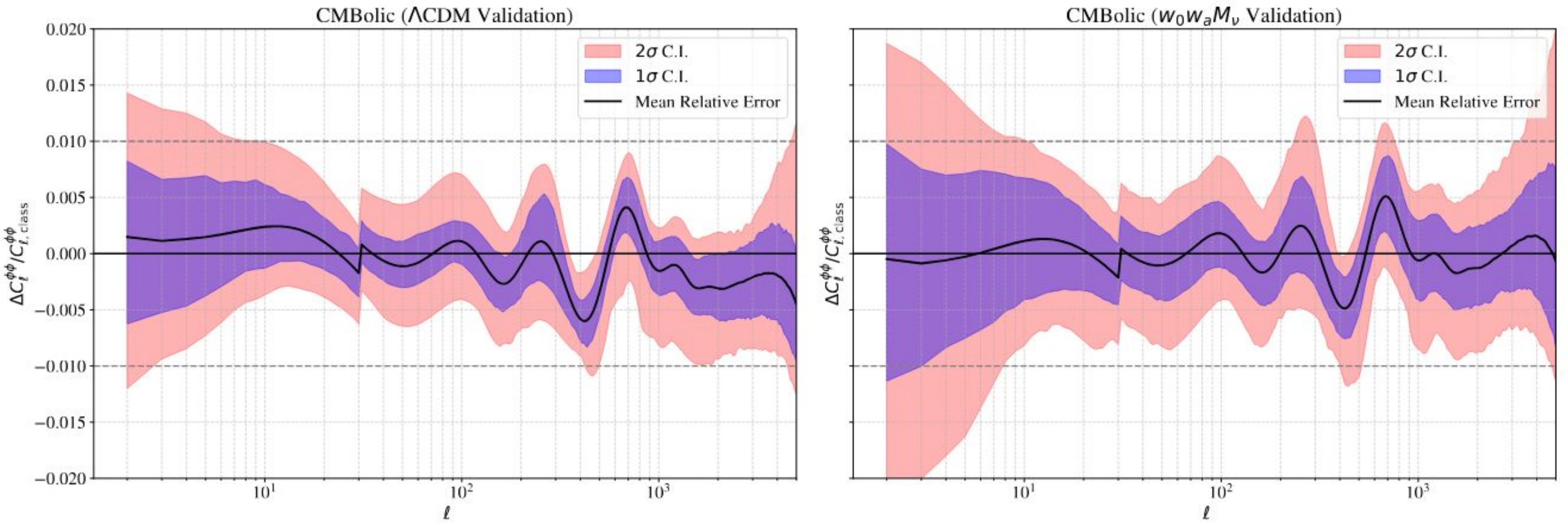
CMBolic

- **Fully analytic** emulation of CMB lensing, temperature and polarization anisotropies
- Trained on hundreds of cosmologies from Class with Act DR6 inspired accuracy settings
- Includes CPL dark energy and massive neutrinos
- Will be **fastest** CMB emulator available
- Lensing emulator paper out soon, temperature and polarization will follow

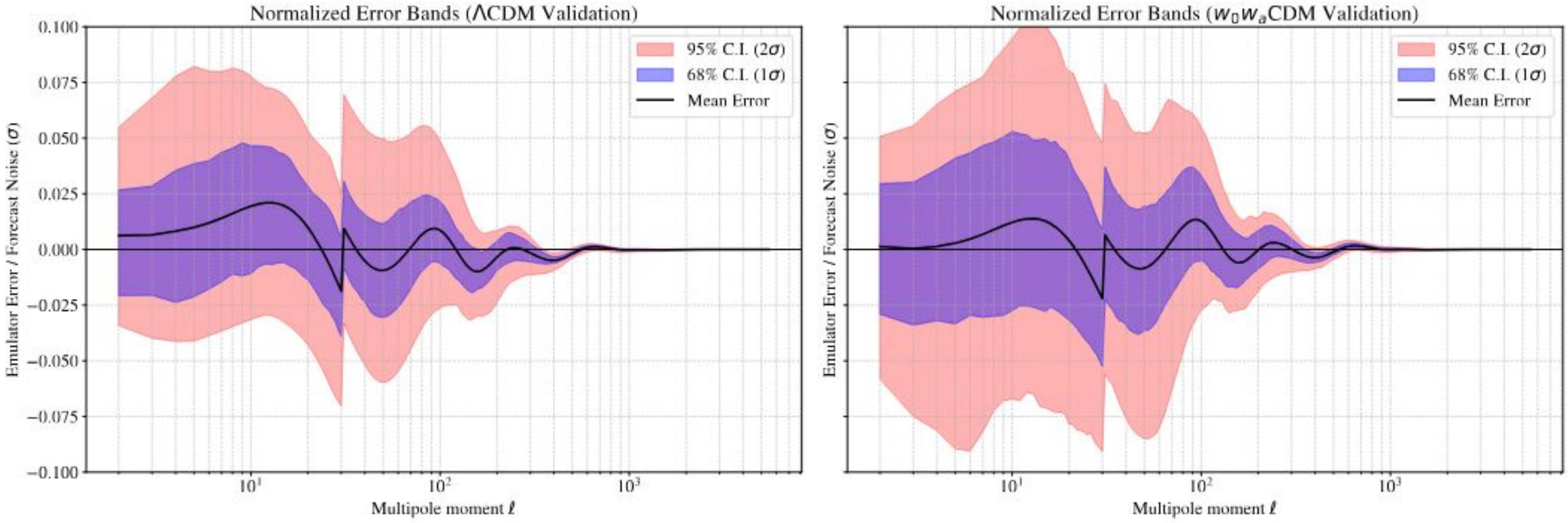
Parameter	Lower bound	Upper bound
$10^9 A_s$	1.7	2.5
Ω_m	0.24	0.40
Ω_b	0.04	0.06
h	0.6	0.8
n_s	0.92	1.01
w_0	-1.15	-0.85
w_a	-0.3	0.3
M_ν	0	0.4

$$\log\left(\ell(\ell+1)\frac{C_\ell^{PP}}{A_s}\right) = C_{16} + \frac{C_{27}\left(\frac{C_{10}\log(\ell)}{\sqrt{\left(C_{14}M_u+\frac{C_{34}\Omega_m}{\sqrt{C_{37}\Omega_b^2+1}}+\frac{C_{15}\log(\ell)+C_{28}}{\sqrt{(C_{26}\log(\ell)+C_6)^2+1}}\right)^2+1}} + C_{29}\Omega_m + C_{30}M_u + C_8\log(\ell) - \frac{C_{31}\log(\ell)+C_{33}M_u}{\sqrt{(C_{12}\Omega_m+C_9h)^2+1}}\right)}{\sqrt{\left(C_{13}n_s + C_{17}w_0 + C_{18}A_s + \frac{C_{19}\log(\ell)}{\sqrt{C_{21}A_s^2+1}} + C_{23}h + C_{25}\log(\ell) + C_5 - \cos\left(C_{24}\log(\ell)\right)\right)^2 + 1}} + \frac{C_{27}\left(C_{38} + \frac{(C_0n_s+C_1\log(\ell)+C_{40}w_a)e^{C_7\Omega_m}}{\sqrt{(C_{35}w_a+C_{36}w_0+\log\left(C_{39}\log(\ell)\right))^2+1}}\right)}{\sqrt{1 + \frac{\left(C_2n_s+C_3+\frac{C_4\log(\ell)}{\sqrt{(C_{11}\Omega_m+C_{20}\log(\ell)+C_{32}\Omega_b)^2+1}}\right)^2}{\cos^2\left(C_{22}\log(\ell)\right)+1}}}$$

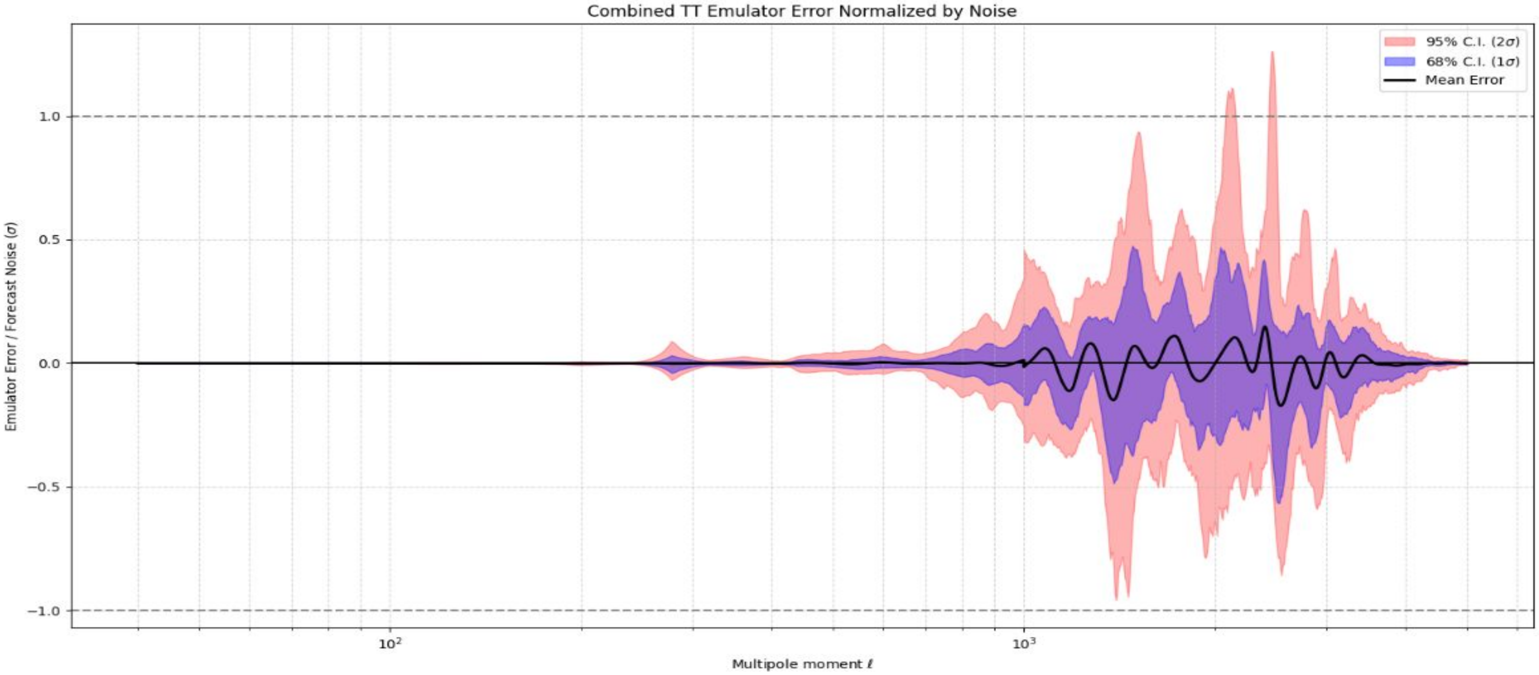
Emulation errors - lensing C_l^{PP}



Normalized to S4 noise forecast



Preliminary - C_l^{TT} error noise normalized

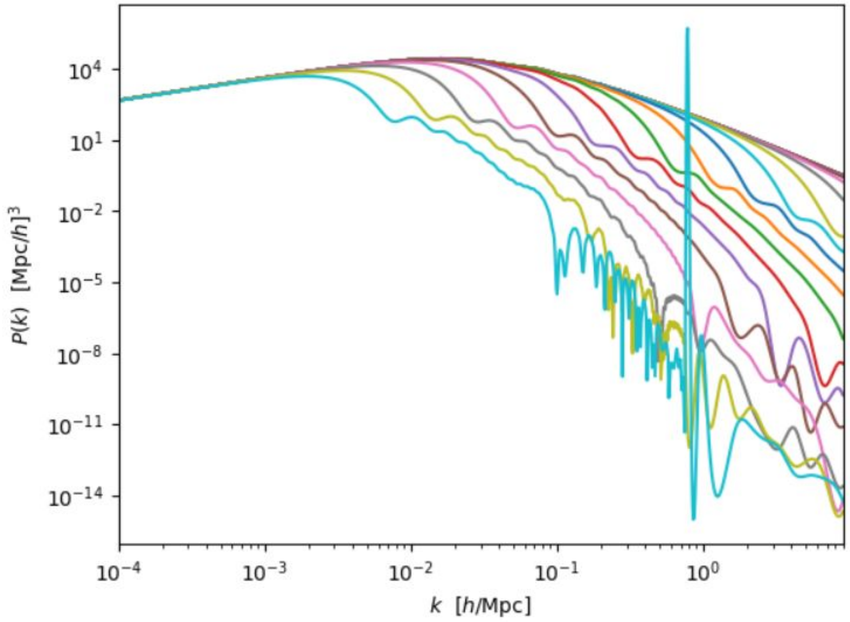


Generalized Dark Matter

- Wayne Hu 1998 - framework of warm DM models defined on the linear perturbation level
- Imperfect fluid with pressure and shear viscosity

$$w, c_s, c_v$$

- Many DM models are GDM at linear level
- Emulate matter power spectra for this theory



Generalized Dark Matter

- Modify Eisenstein-Hu formula to include GDM parameters in expressions
- Subtract this estimate and fit residuals

$$z_{eq} = C_1 + C_2 h (C_3 h \Omega_m)^{exp(C_4 * w)}$$

[Eisenstein, Hu 1998]

The transfer function is written as a sum of the baryon and cold dark matter contributions at the drag epoch

$$T(k) = \frac{\Omega_b}{\Omega_0} T_b(k) + \frac{\Omega_c}{\Omega_0} T_c(k) . \tag{8}$$

The CDM transfer function can be solved exactly in terms of hypergeometric functions that are more conveniently approximated by the following form:

$$T_c \to \alpha_c \frac{\ln 1.8\beta_c q}{14.2 q^2} , \tag{9}$$

$$q = \left(\frac{k}{\mathrm{Mpc}^{-1}}\right) \Theta_{2.7}^2 (\Omega_0 h^2)^{-1} = \frac{k}{13.41 k_{\mathrm{eq}}} , \tag{10}$$

where α_c and β_c are fitted by

$$\begin{aligned} \alpha_c &= a_1^{-\Omega_b/\Omega_0} a_2^{-(\Omega_b/\Omega_0)^3} , \\ a_1 &= (46.9\Omega_0 h^2)^{0.670}[1 + (32.1\Omega_0 h^2)^{-0.532}] , \\ a_2 &= (12.0\Omega_0 h^2)^{0.424}[1 + (45.0\Omega_0 h^2)^{-0.582}] , \end{aligned} \tag{11}$$

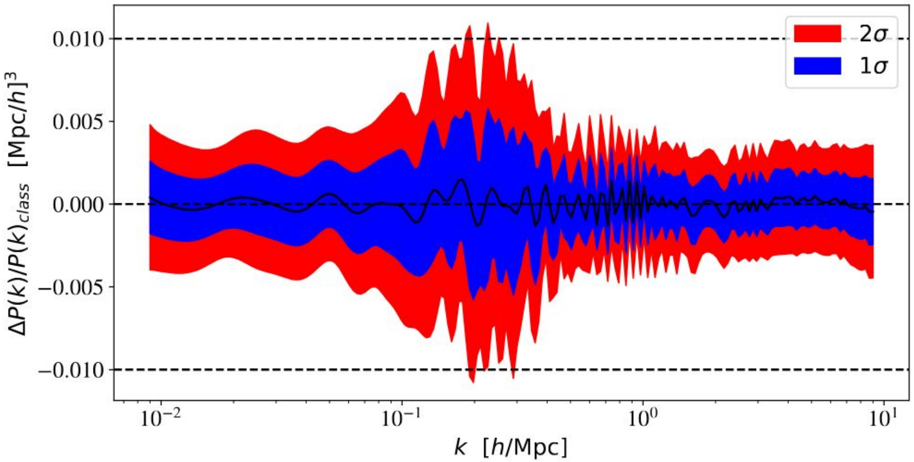
$$\begin{aligned} \beta_c^{-1} &= 1 + b_1[(\Omega_c/\Omega_0)^{b_2} - 1] , \\ b_1 &= 0.944[1 + (458\Omega_0 h^2)^{-0.708}]^{-1} , \\ b_2 &= (0.395\Omega_0 h^2)^{-0.0266} . \end{aligned} \tag{12}$$

As $\Omega_b/\Omega_0 \to 0$, $\alpha_c, \beta_c \to 1$. Equation (9) shows the familiar ln $(k)/k^2$ dependence of the small-scale CDM transfer function.

Preliminary

- Separate emulators for various regimes of the theory:
  - Include w
  - Include w, c_s
  - Include w, c_s, c_v in damped regime
  - Include w, c_s, c_v in undamped regime

wGDM emulator



Summary

- **Symbolic Regression** is an interesting and powerful techniques applicable for many problems in physics
  - Full observable emulation
  - Speed up computationally intensive subparts
  - Discover new physic laws
- **CMBolic** - a new emulator for CMB lensing, temperature and polarisation anisotropies, fully public coming out soon
- **Generalized Dark Matter** power spectra emulator coming out soon, possibility to test many modified gravity models at once