Multiview Symbolic Regression

How to learn laws from examples

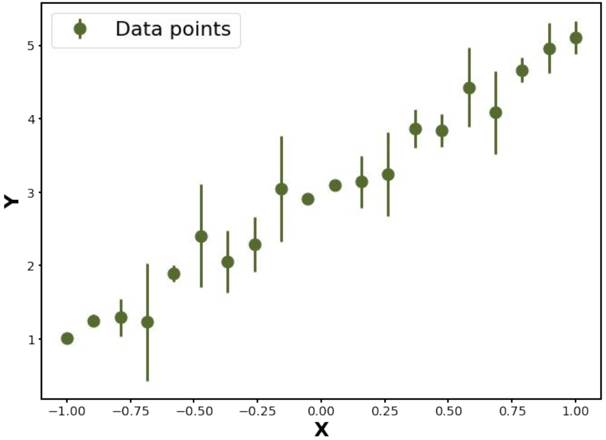
Etienne Russeil - *LPC Université Clermont Auvergne, France*
Fabricio Olivetti - *CMCC Federal University of ABC, Brazil*
Emille Ishida - *LPC Université Clermont Auvergne, France*
Konstantin Malanchev - *University of Illinois Urbana–Champaign, USA*
Emmanuel Gangler - *LPC Université Clermont Auvergne, France*

II

Symbolic Regression

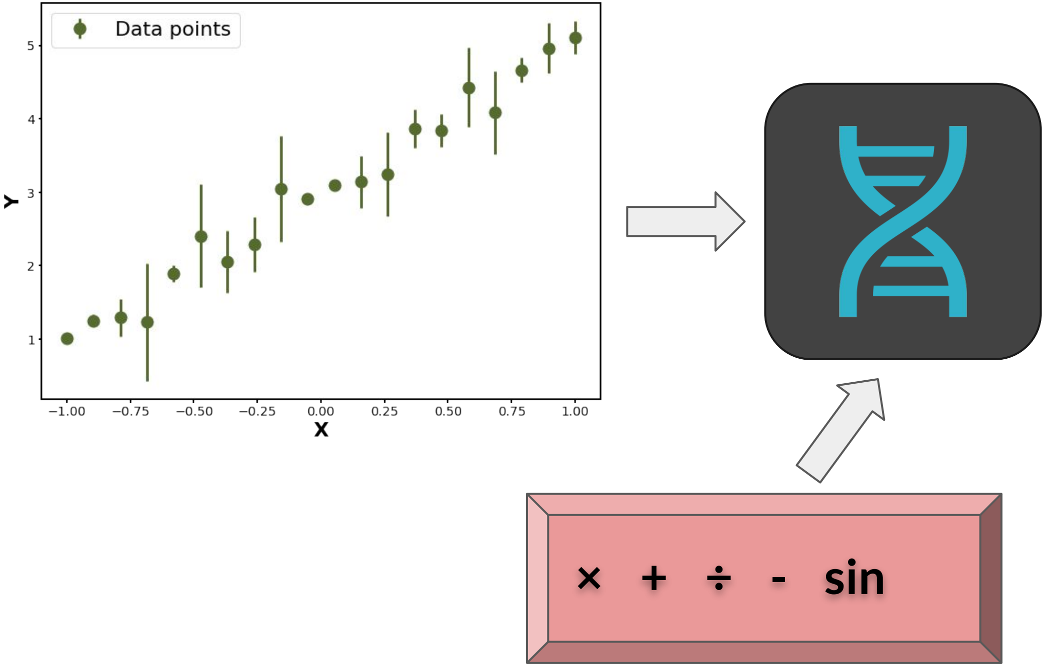
Traditional Symbolic Regression

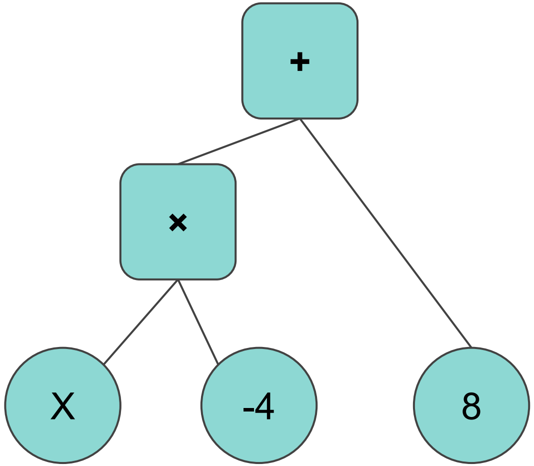
DATA SET



Traditional Symbolic Regression

DATA SET

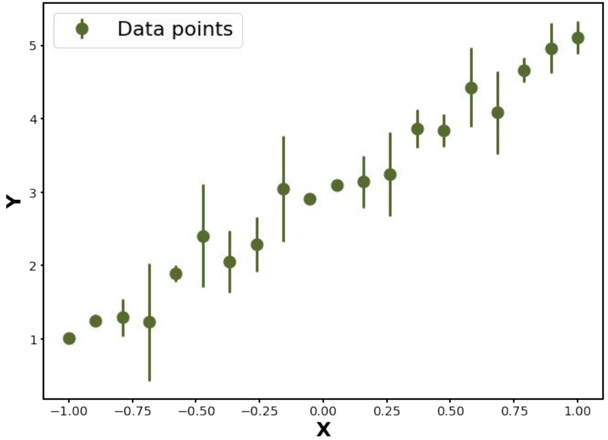




As a first step the algorithm will randomly generate many different equations

Traditional Symbolic Regression

DATA SET



RANDOM EQUATIONS

f(X) = sin(X) + 2

f(X) = X² - 1

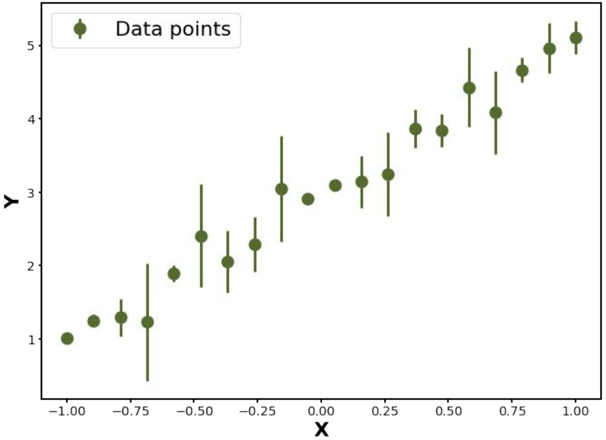
f(X) = 42

f(X) = -4X + 8

f(X) = 2X

Traditional Symbolic Regression

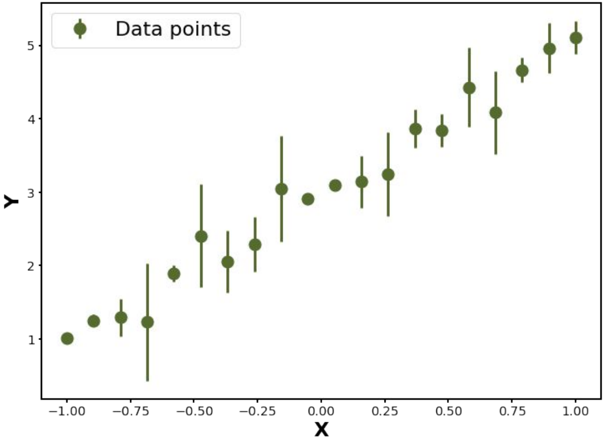
DATA SET



RANDOM EQUATIONS	COST FUNCTION
$f(X) = \sin(X) + 2$	COST = 12
$f(X) = X^2 - 1$	COST = 24
$f(X) = 42$	COST = 43
$f(X) = -4X + 8$	COST = 7
$f(X) = 2X$	COST = 3

Traditional Symbolic Regression

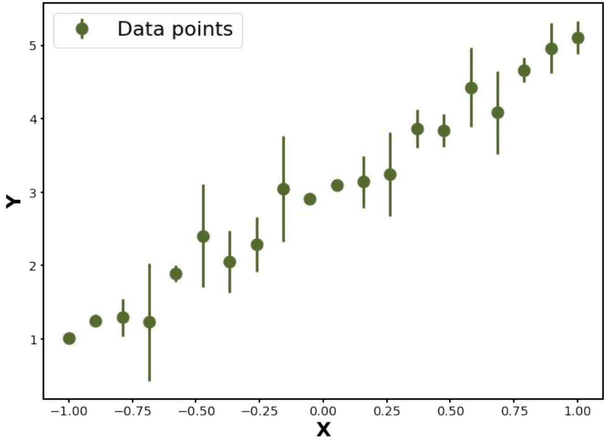
DATA SET



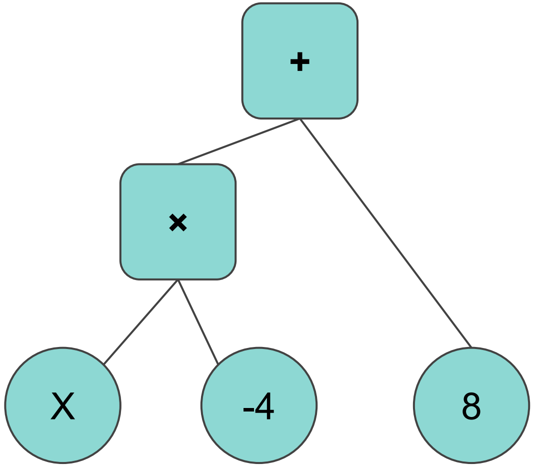
RANDOM EQUATIONS	COST FUNCTION
$f(X) = \sin(X) + 2$	COST = 12
$f(X) = X^2 - 1$	COST = 24
$f(X) = 42$	COST = 43
$f(X) = -4X + 8$	COST = 7
$f(X) = 2X$	COST = 3

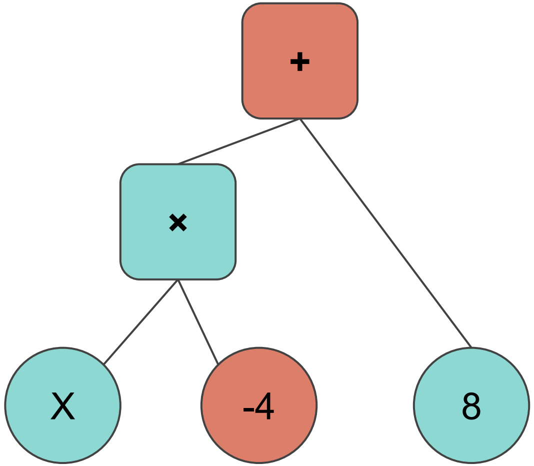
Traditional Symbolic Regression

DATA SET

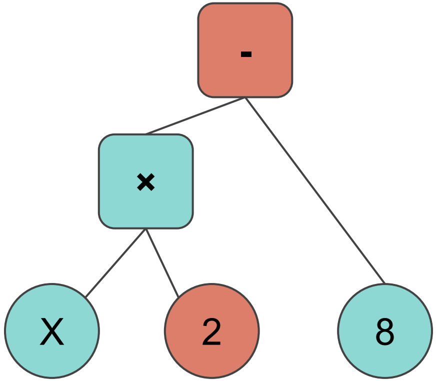


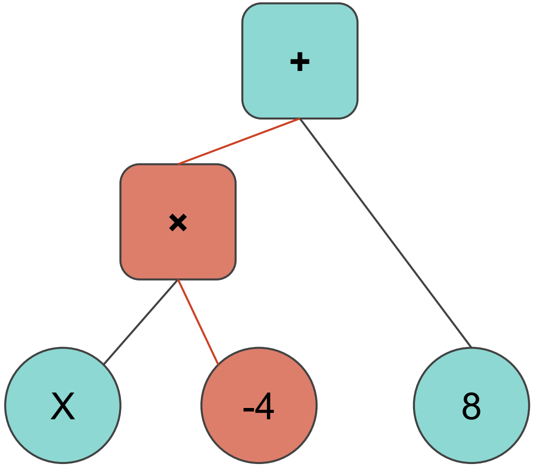
Apply random mutations
to the best candidates


+
×
X
-4
8

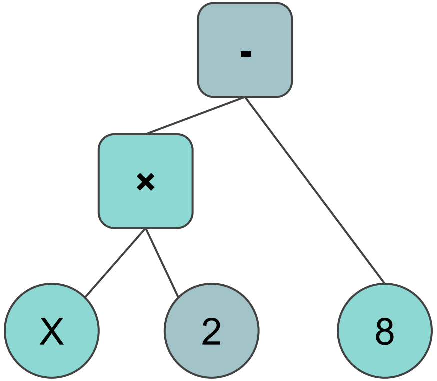


Point mutations

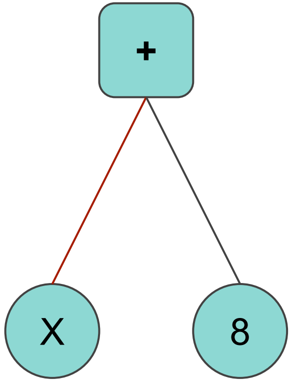


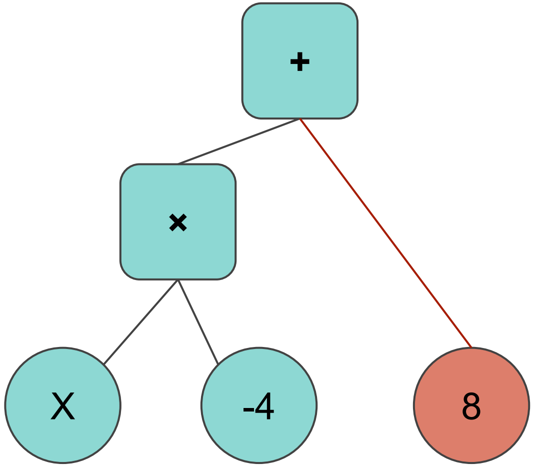


Point mutations

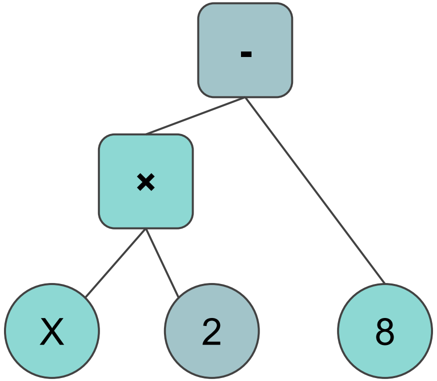


Hoist mutations

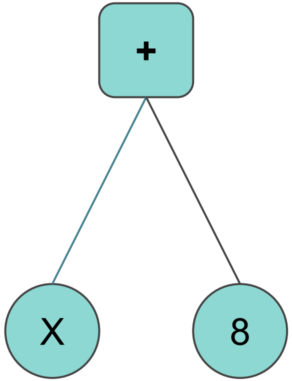




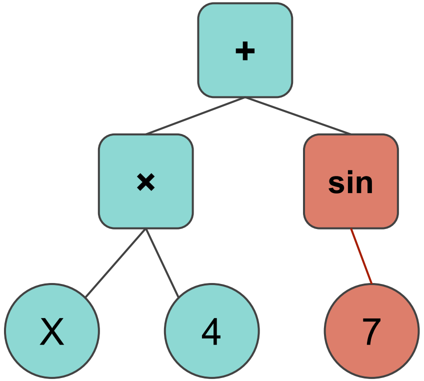
Point mutations

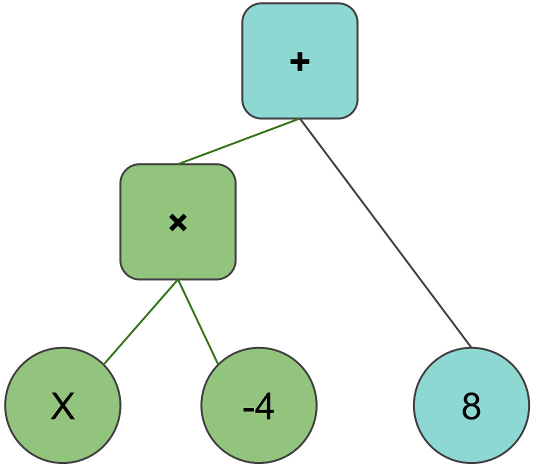


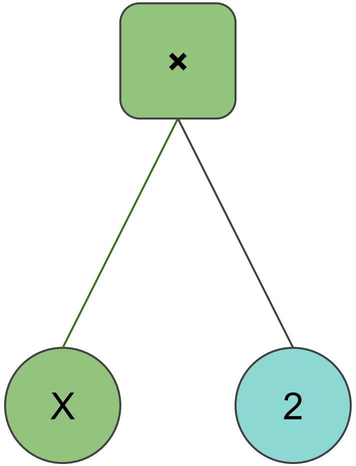
Hoist mutations



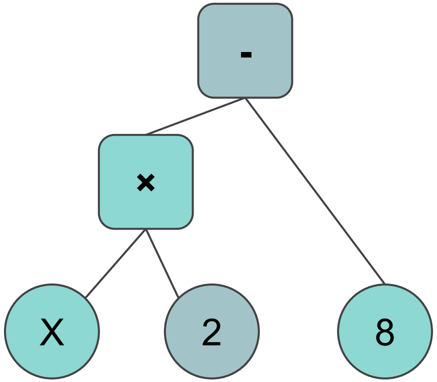
Subtree mutations



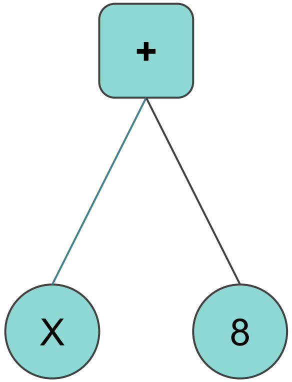




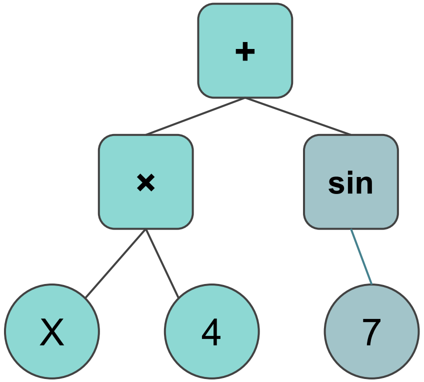
Point mutations



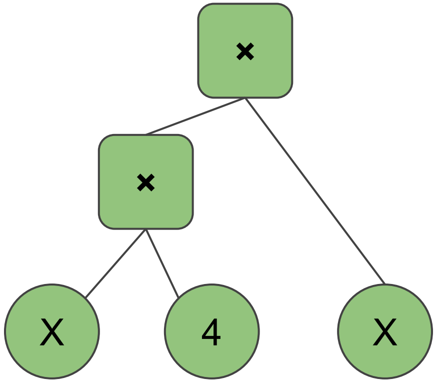
Hoist mutations



Subtree mutations

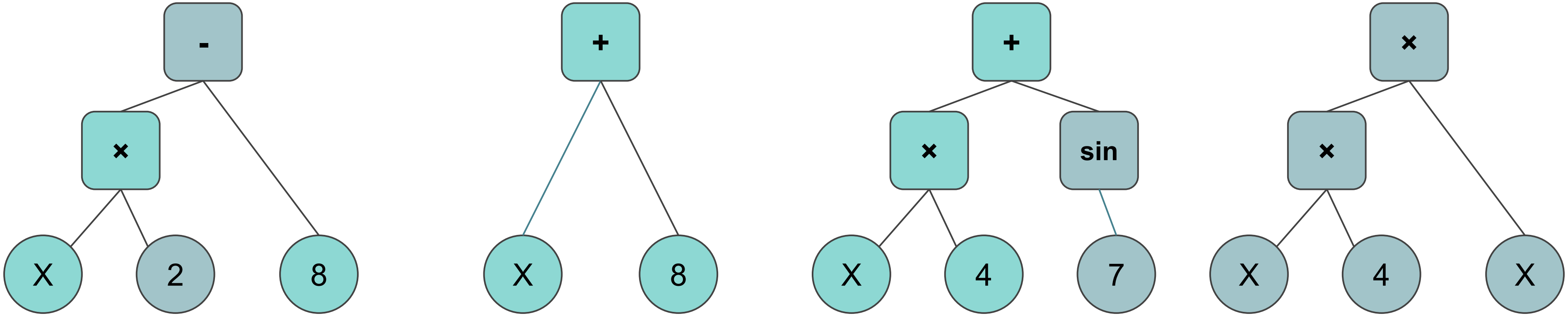


Crossover



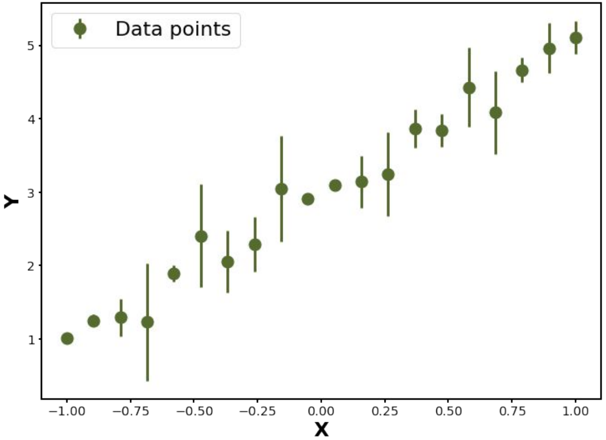
Create a new population from the previous best candidates

Point mutations · Hoist mutations · Subtree mutations · Crossover



Traditional Symbolic Regression

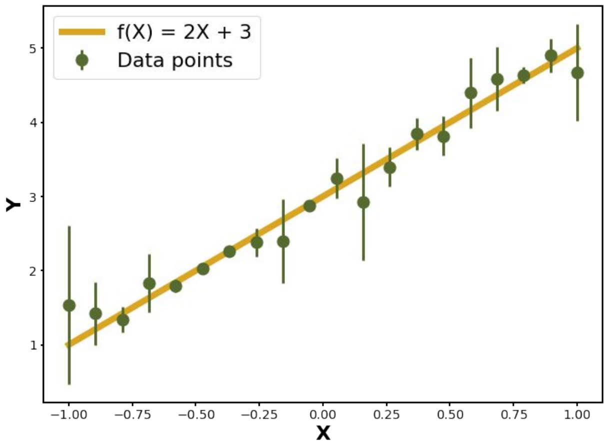
DATA SET



EVOLVED EQUATIONS	COST FUNCTION
f(X) = 2X - 8	COST = 10
f(X) = X	COST = 7
f(X) = 2X + 2	COST = 0.5
f(X) = 2X	COST = 7
f(X) = 1/X	COST = 42

Traditional Symbolic Regression

DATA SET



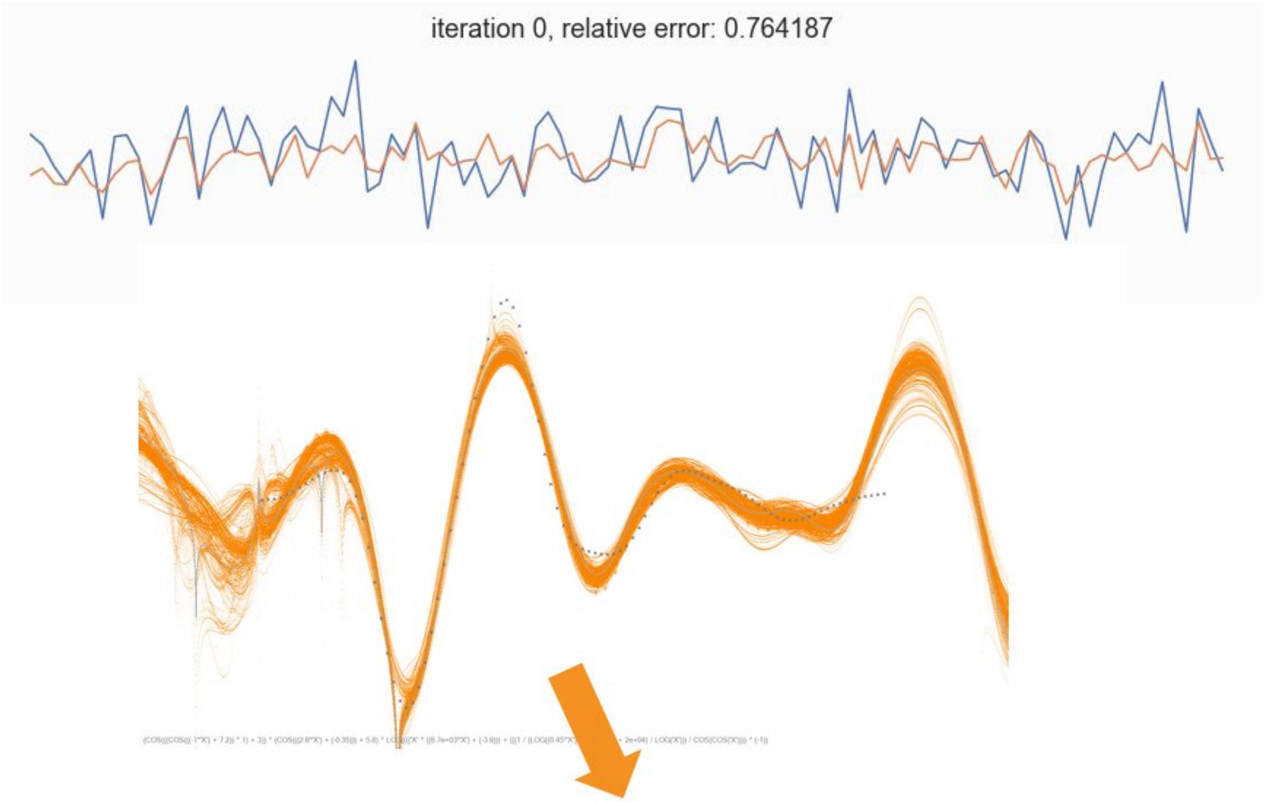
After many generation

Best answer

$\boxed{f(X) = 2\,X + 3}$

COST ~ 0

Traditional Symbolic Regression

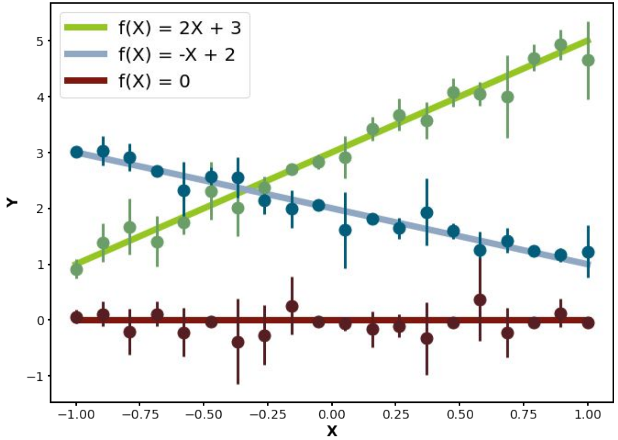


$$e^{-x}x^3\cos(x)\sin(x)(cos(x)sin(x)^2-1)$$

https://github.com/heal-research/operon

Traditional Symbolic Regression : Limitation

DATA SETS



Best answers

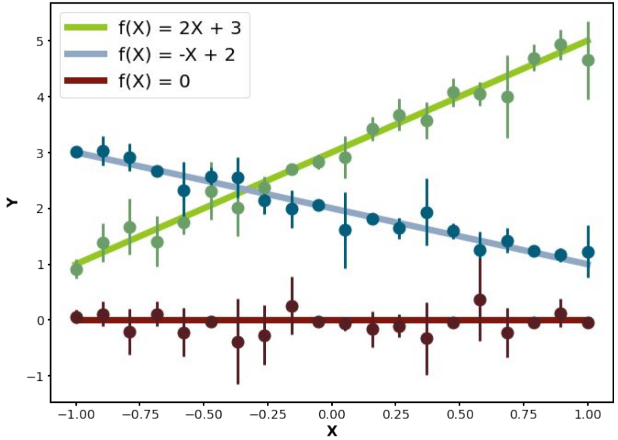
f(X) = 2 X + 3

f(X) = -X + 2

f(X) = 0

Traditional Symbolic Regression : Limitation

DATA SETS



Best answers

f(X) = 2 X + 3

f(X) = -X + 2

f(X) = 0

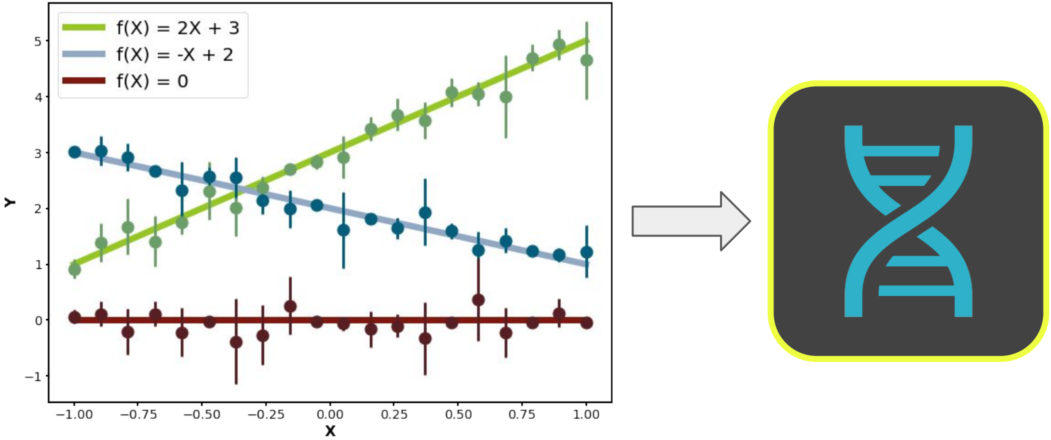
Could it find f(X) = AX + B ?

III

MultiView Symbolic Regression

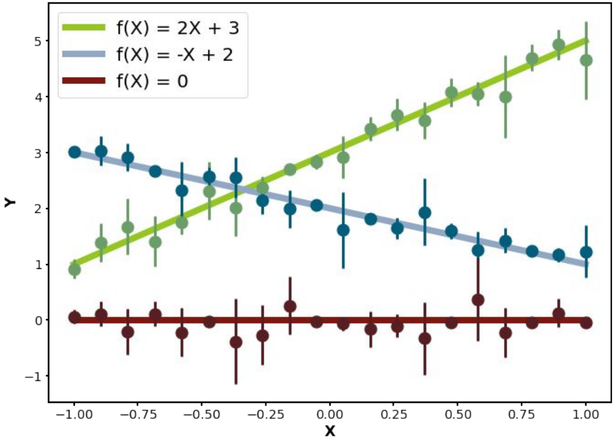
Multiview Symbolic Regression (MvSR)

DATA SETS



Multiview Symbolic Regression (MvSR)

DATA SETS



RANDOM EQUATIONS

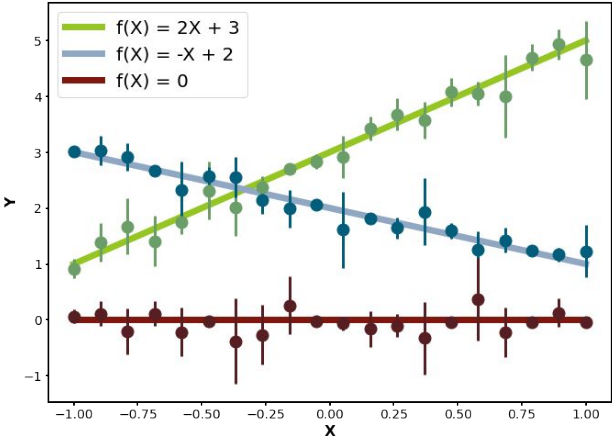
$f(X) = \sin(X) + \mathbf{A}$

$f(X) = \mathbf{A} + \mathbf{B}\, X^2$

$f(X) = \mathbf{A}$

Multiview Symbolic Regression (MvSR)

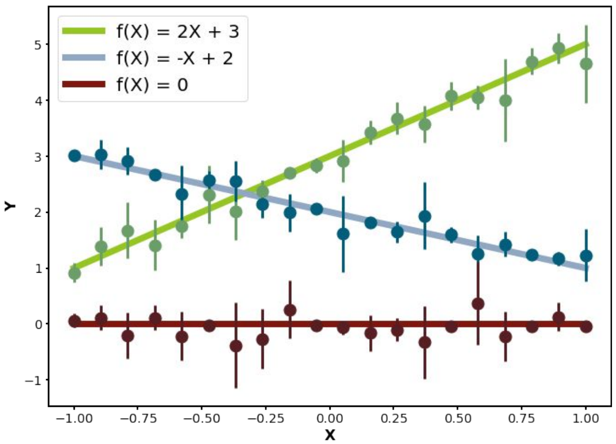
DATA SETS



RANDOM EQUATIONS	COST FUNCTION AFTER MINIMIZATION
f(X) = sin(X) + **A**	COST = 24, 32, 7
f(X) = **A** + **B** X^2	COST = 17, 8, 0
f(X) = **A**	COST = 19, 10, 0

Multiview Symbolic Regression (MvSR)

DATA SETS



RANDOM EQUATIONS

COST FUNCTION AFTER MINIMIZATION

f(X) = sin(X) + **A** — COST = 24, 32, 7

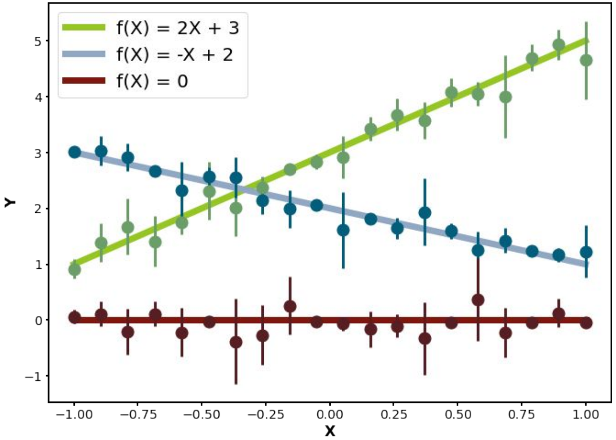
f(X) = **A** + **B** X^2 — COST = 17, 8, 0

f(X) = **A** — COST = 19, 10, 0

Average COST

Multiview Symbolic Regression (MvSR)

DATA SETS



After many generation

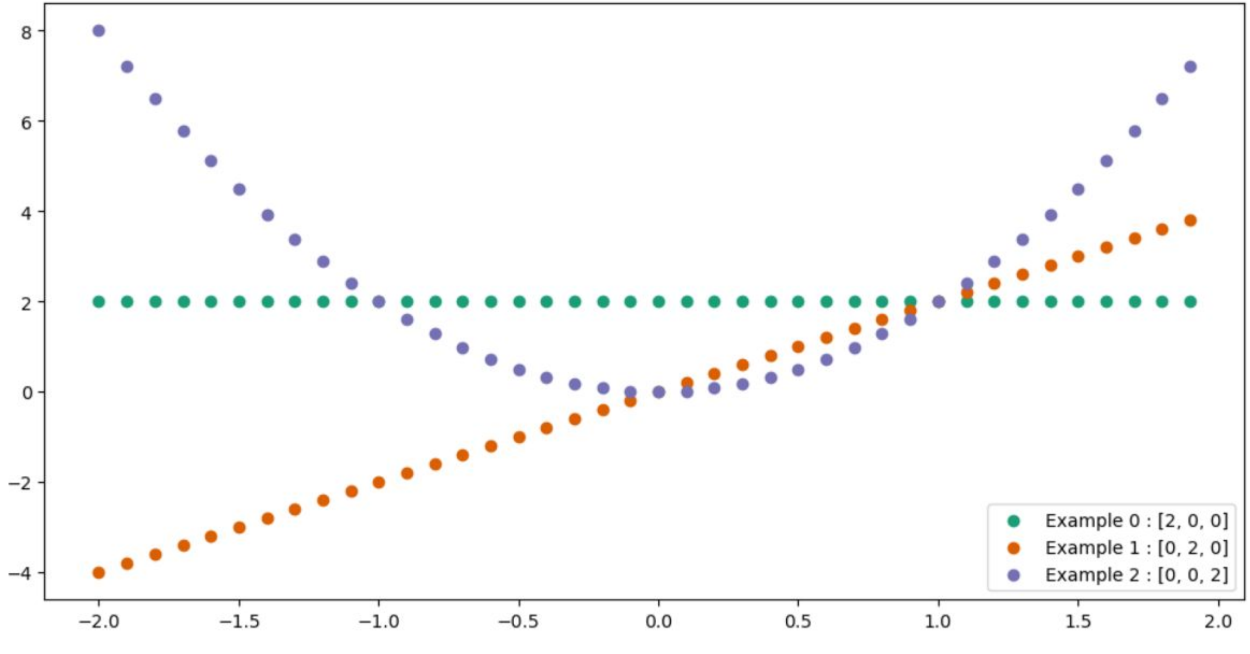
Best answer

f(X) = A X + B

COST = 0, 0, 0

Multiview Symbolic Regression (MvSR)

Toy data illustration



$$f(X) = A + BX + CX^2$$

Multiview Symbolic Regression (MvSR)

Strong points

- Directly reconstructs parametric equations
- Make sense of partial multiple information
- Much harder to overfit
- Allow for a control of the number of free parameters

Multiview Symbolic Regression (MvSR)

Strong points

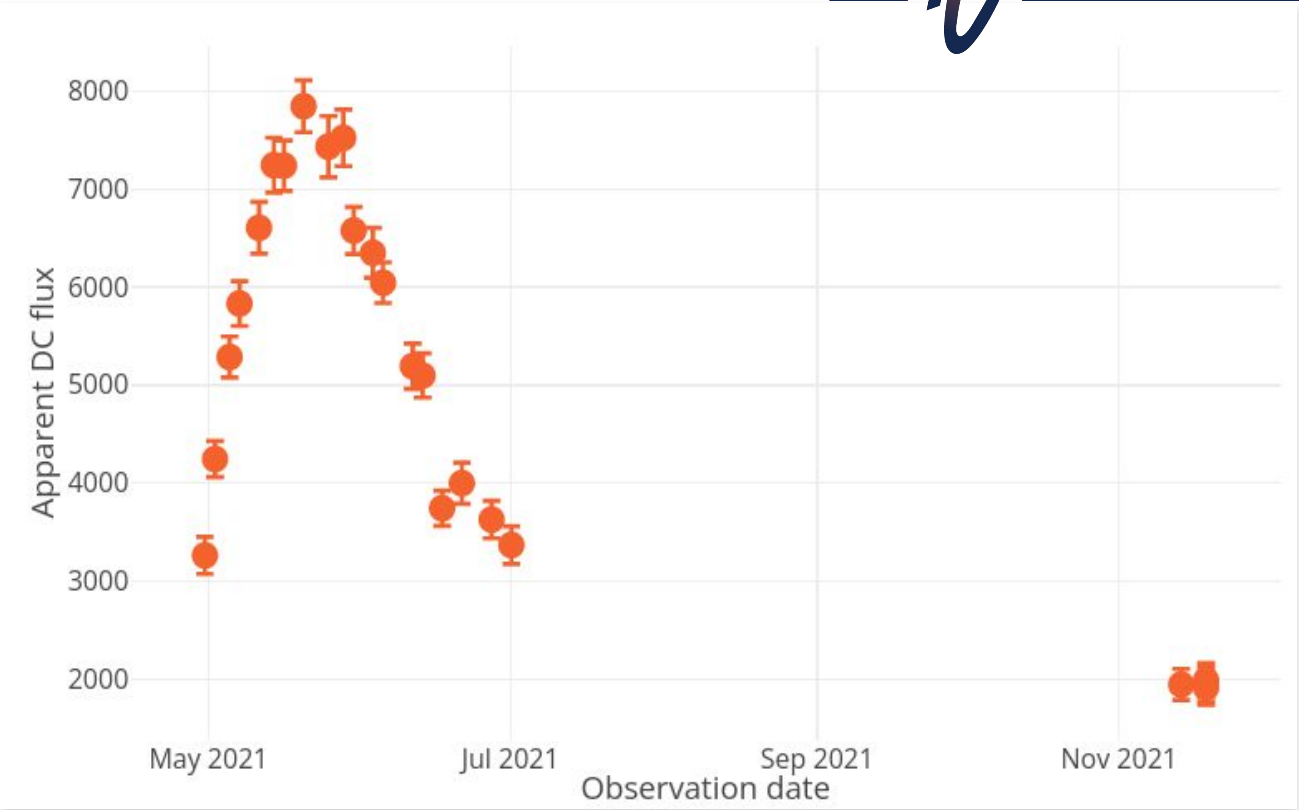
- Directly reconstructs parametric equations
- Make sense of partial multiple information
- Much harder to overfit
- Allow for a control of the number of free parameters

Currently working on a paper with a partial implementation of the idea

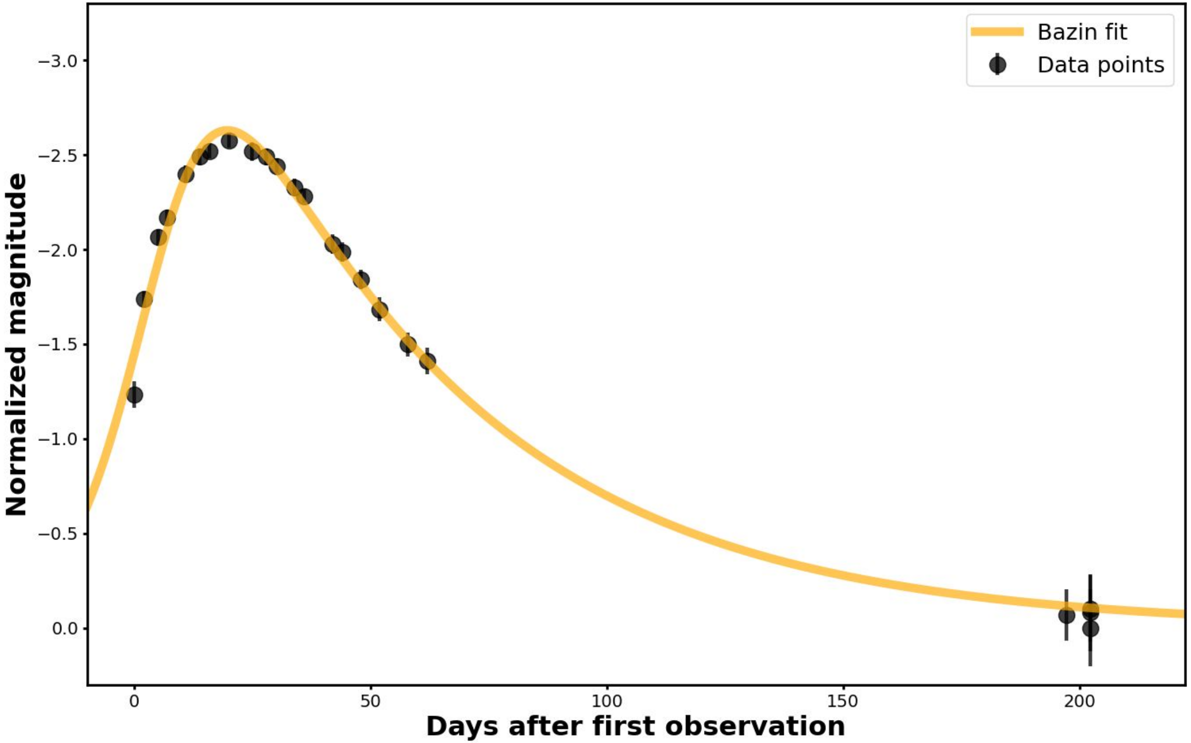
IV

Astrophysical context

Astrophysical use case



Astrophysical use case



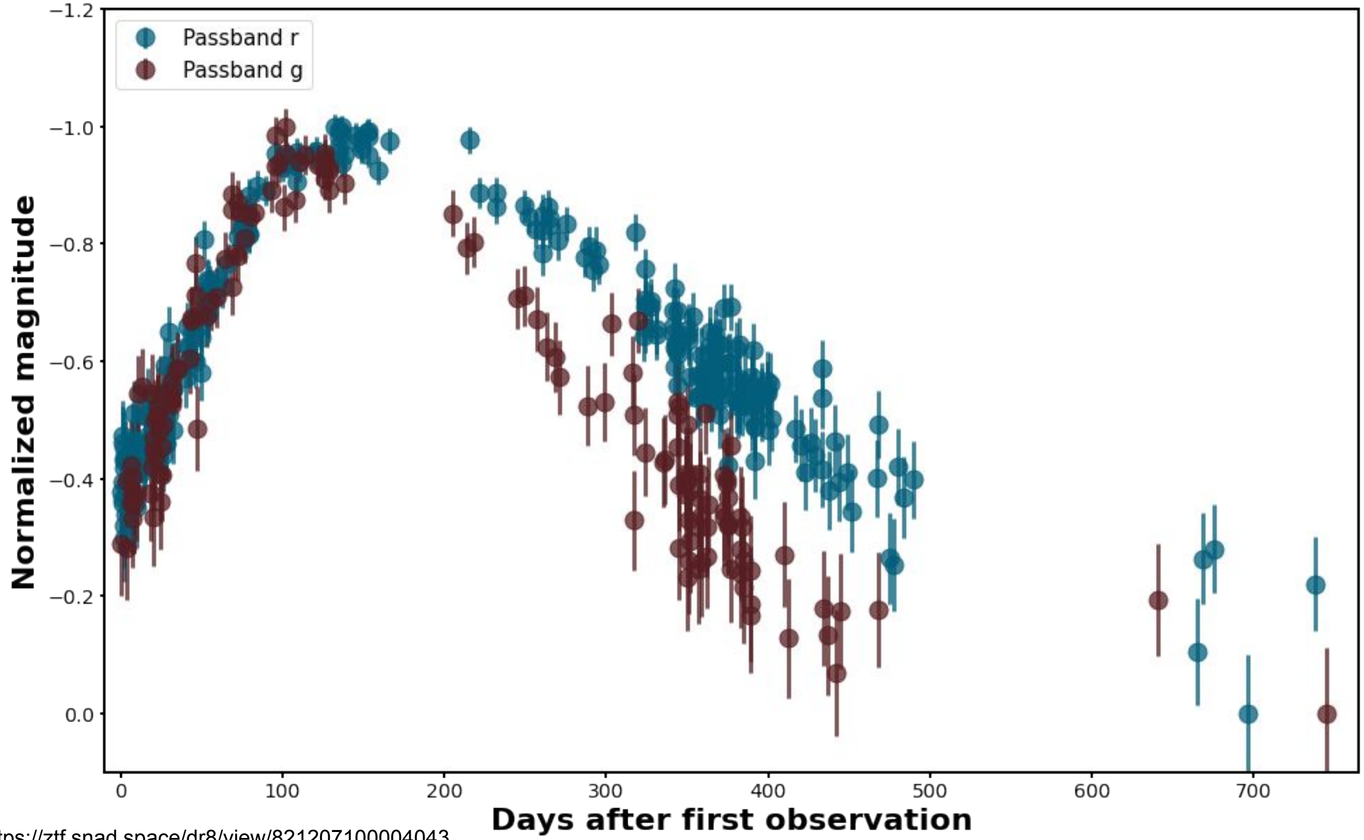
Bazin function :

$$f(t) = a \times \frac{e^{-\frac{(t-t0)}{t_{fall}}}}{1 + e^{\frac{(t-t0)}{t_{rise}}}}$$

Anything better ?

Astrophysical use case

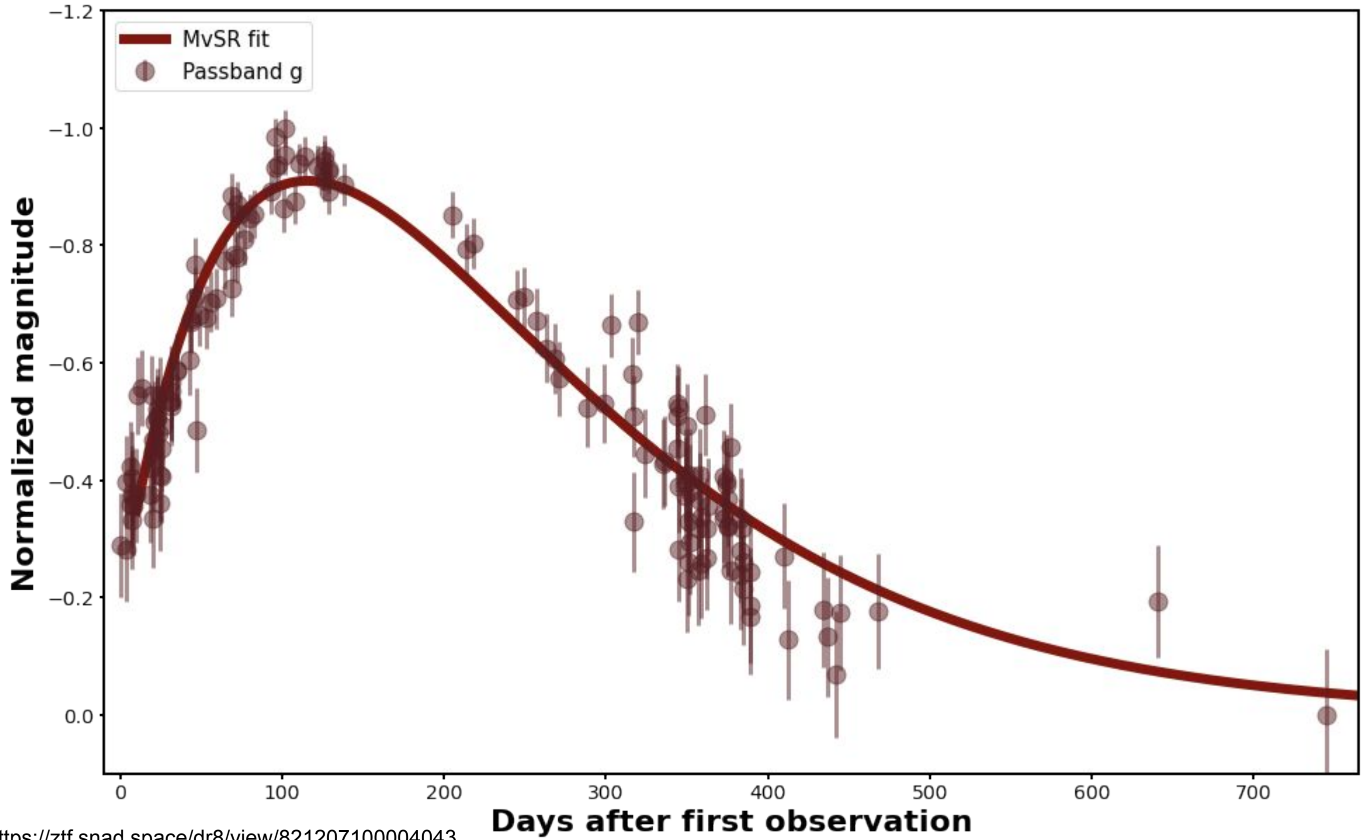
SNAD 160



https://ztf.snad.space/dr8/view/821207100004043

Astrophysical use case

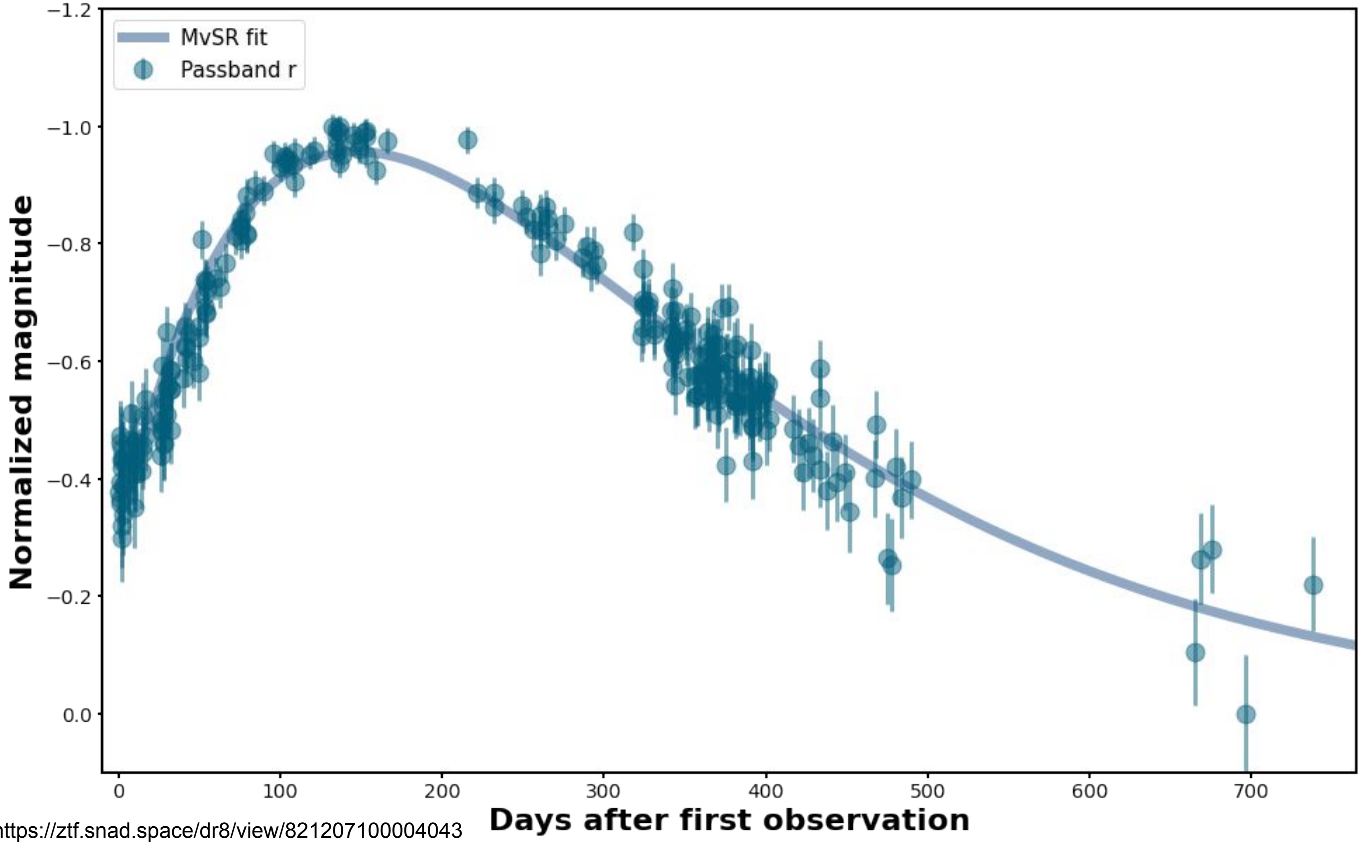
$$f(t) = A(t - t_0) \times e^{B(t-t0)}$$



https://ztf.snad.space/dr8/view/821207100004043

Astrophysical use case

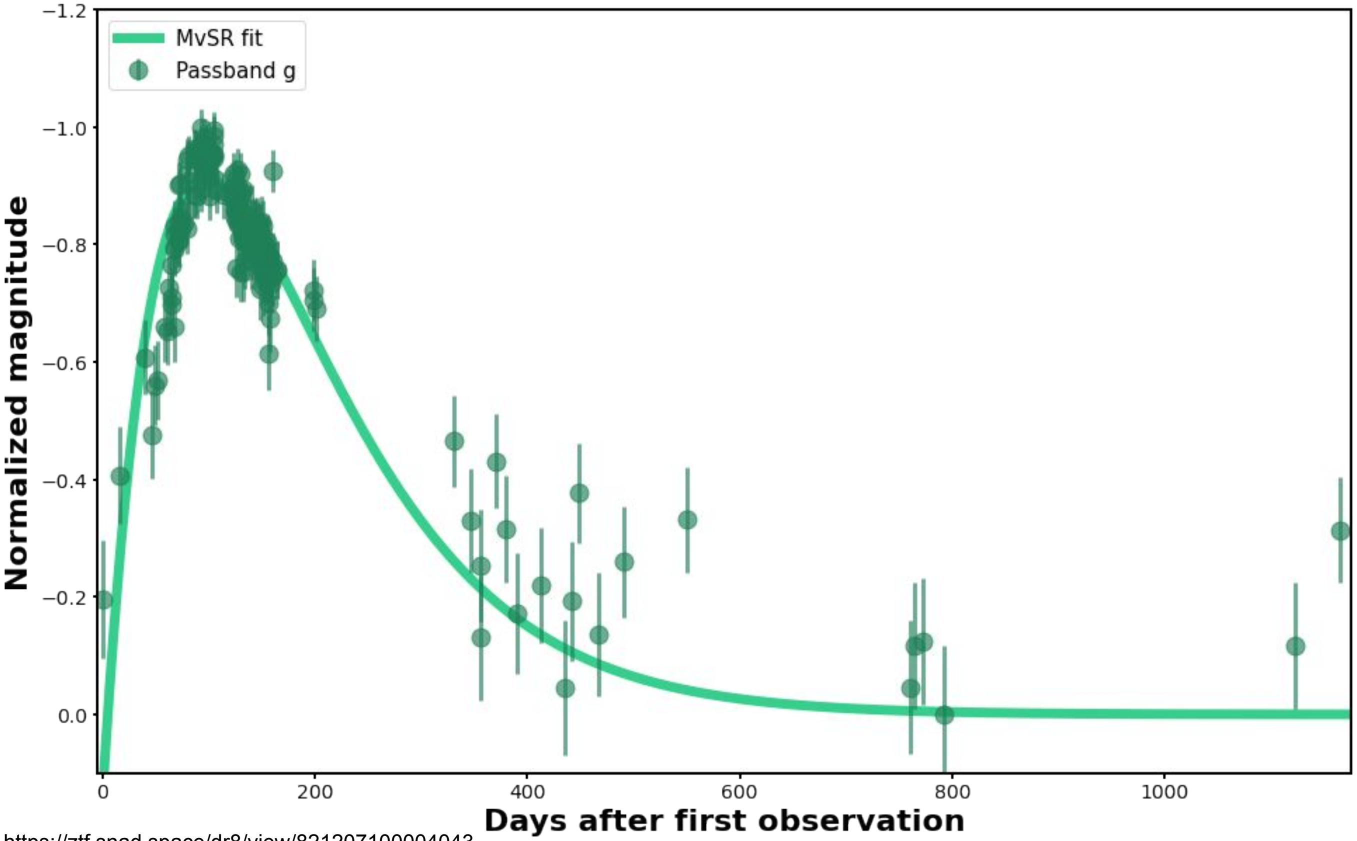
$$f(t) = A(t - t_0) \times e^{B(t-t0)}$$



https://ztf.snad.space/dr8/view/821207100004043

Astrophysical use case

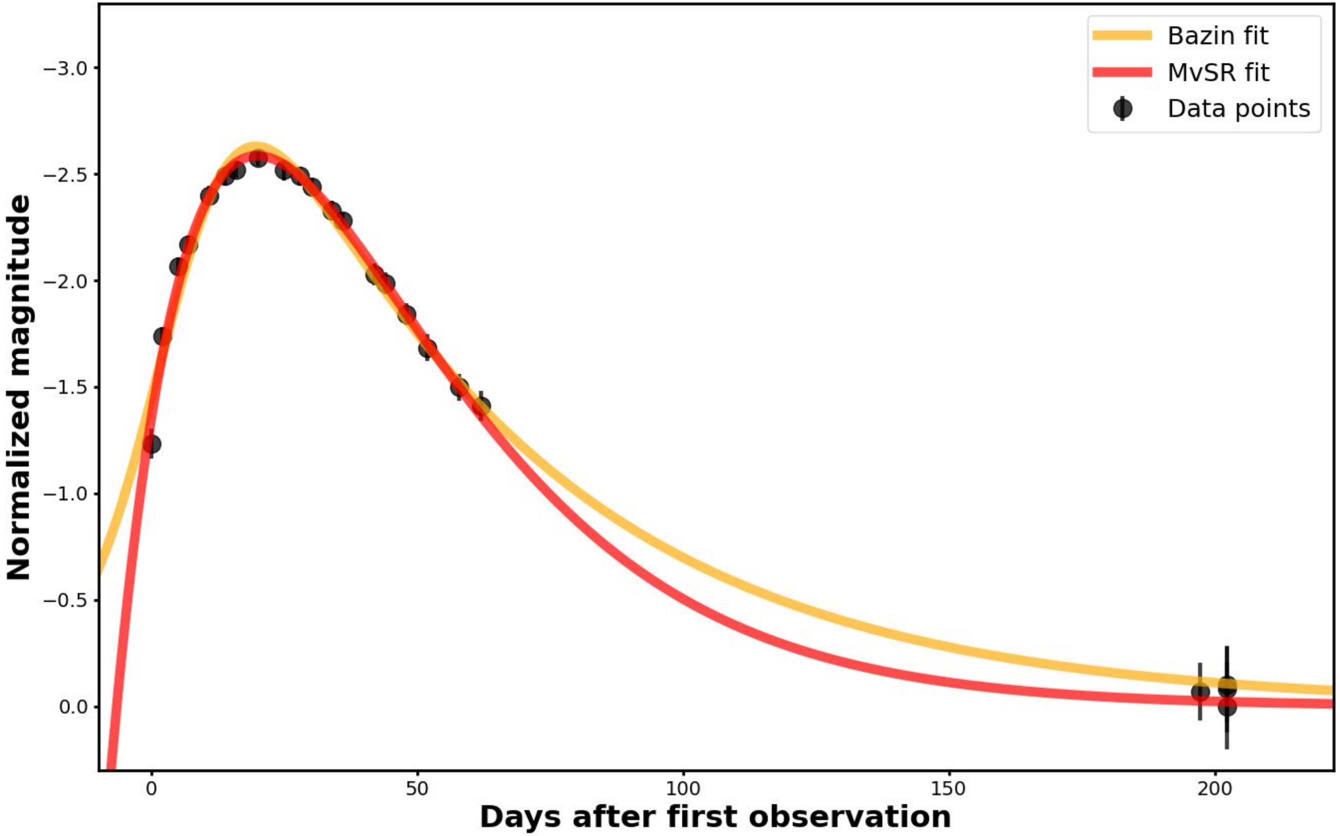
$$f(t) = A(t - t_0) \times e^{B(t-t0)}$$



https://ztf.snad.space/dr8/view/821207100004043

Astrophysical use case

$$f(t) = A(t - t_0) \times e^{B(t-t0)}$$



Astrophysical use case

$$f(t) = A(t - t_0) \times e^{B(t-t0)}$$

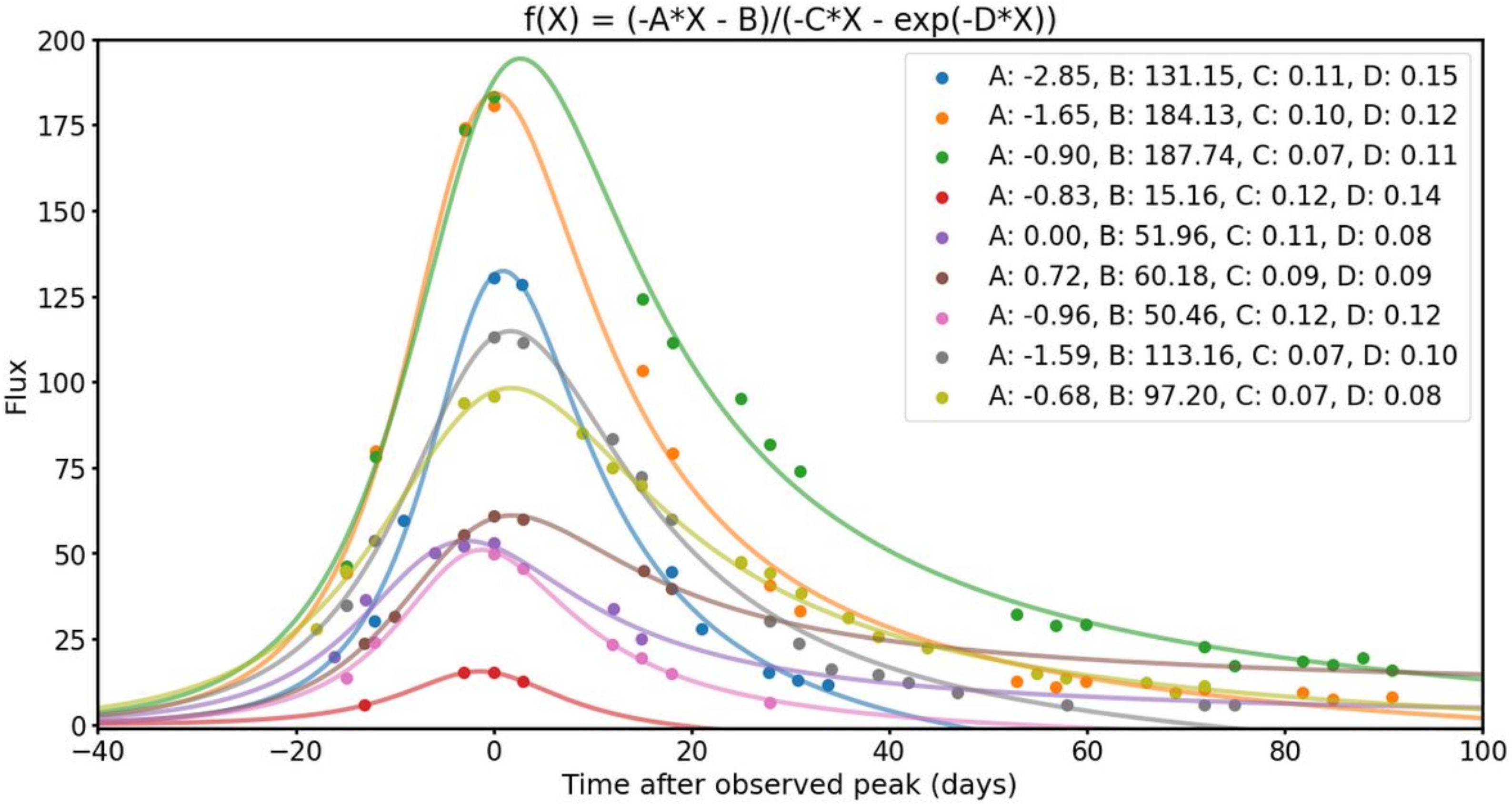
Already used inside Fink !

Superluminous supernovae classifier for the ELAsTiCC challenge

Gives better classification result than using a Bazin fit

Astrophysical use case

Preliminary work !



VI

Conclusion

Conclusion

- MvSR is working and we are writing a paper to present it formally
- For now MvSR implementation is just a proof of concept, more work to come
- Can be used in astrophysics to optimally describe light curves
- Future work: analysis of the different astrophysical functions proposed by MvSR

Thank you for your attention