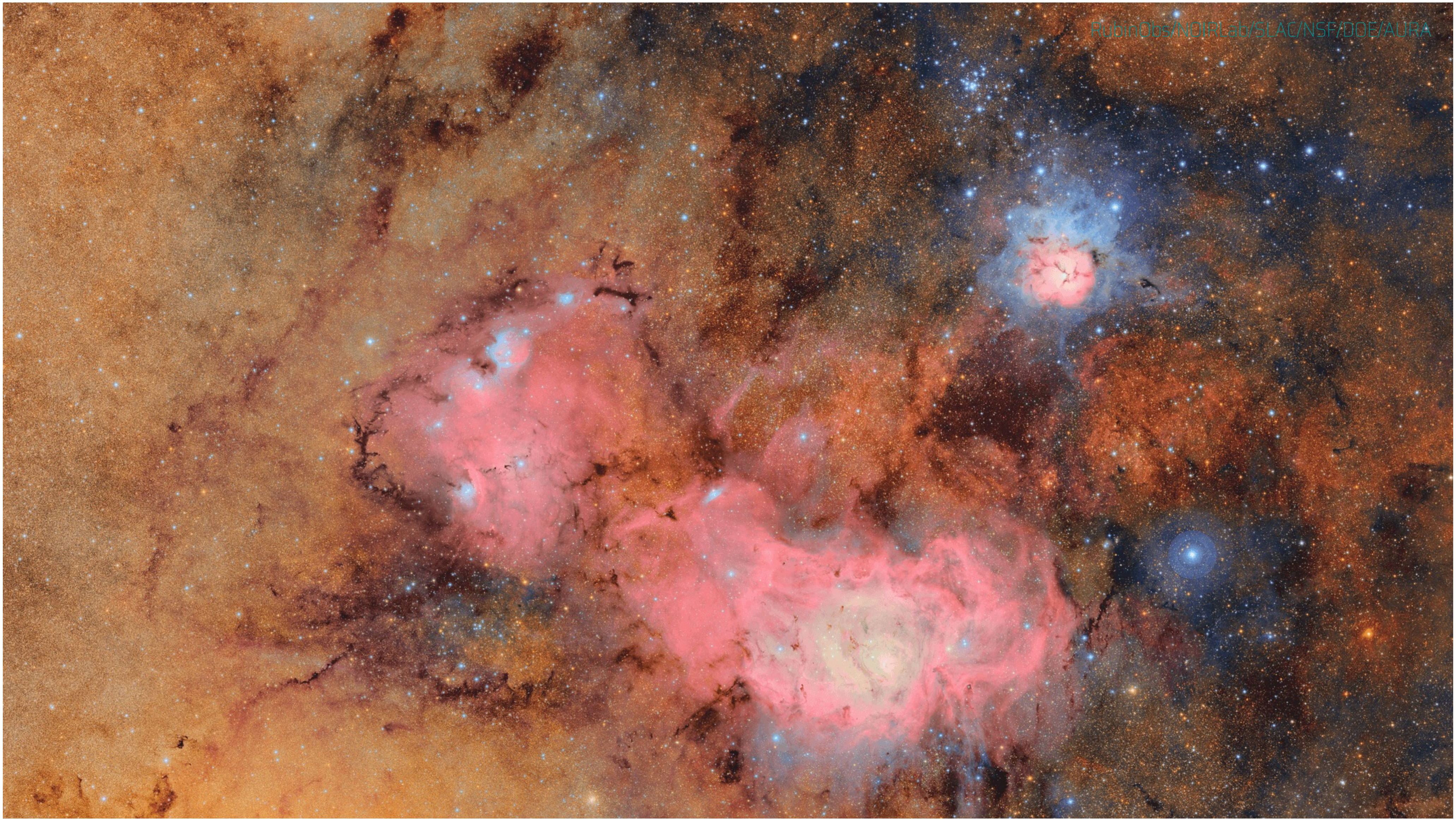
Transient hunting in large-sky surveys: using Neural Processes to tackle sparse, multi-wavelength light curves

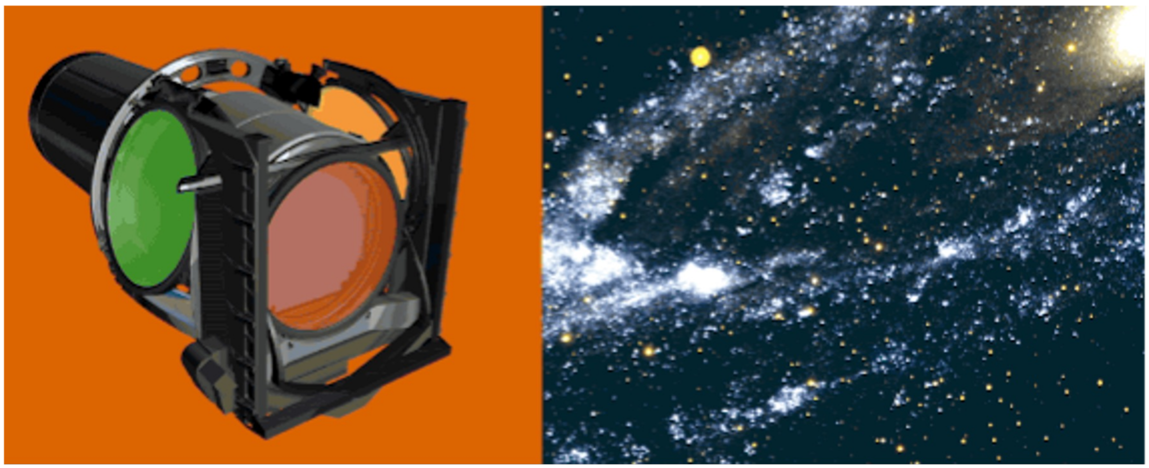
RAPID Response

Siddharth Chaini
NASA FINESST Fellow & UD Fellow of Excellence
University of Delaware


RubinObs/NOIRLab/SLAC/NSF/DOE/AURA


RubinObs/NOIRLab/SLAC/NSF/DOE/AURA

The Multiband Promise of LSST



SLAC National Accelerator Laboratory

LSST's 6 bands (*ugrizy*)
⇒ some colour information

The Multiband Promise of LSST + Synergies

LSST's 6 bands (*ugrizy*)

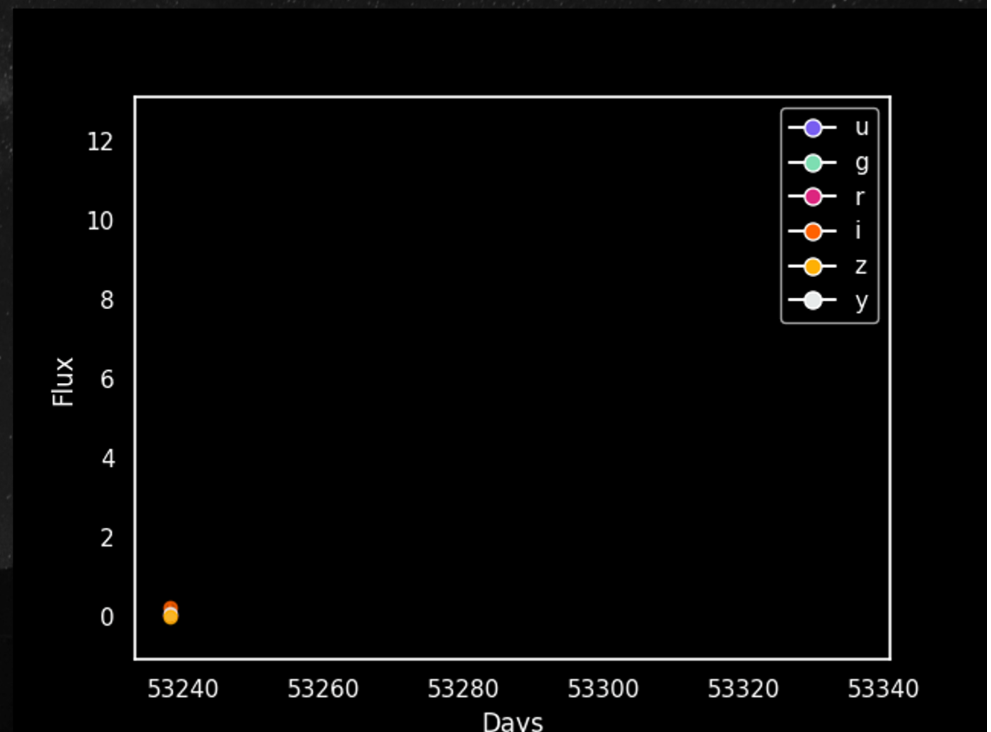
+ synergistic co-observations from

- ZTF
- Roman
- Euclid
- DECAM (e.g. Shadow)
- Argus array
- LS4
- etc.

Discovery Machine

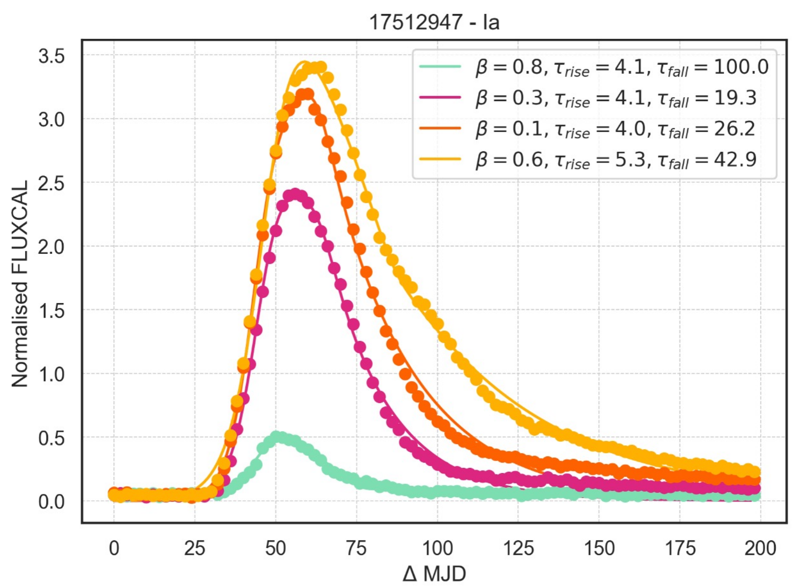
Rubin will produce millions of light curves!

- ~10M QSO Mary Loli+21
- ~3.3M SN II Hložek+20
- ~800k SNIa (z<1) “well observed” Griz+24
- ~580k SN Ibc Hložek+20
- ~300k SNIa (z<0.3) Griz+24
- ~50k Tidal Disruption Events Brickman+ 20
- ~10k SuperLuminous Supernovae Villar+ 20
- ~400 strongly lensed SN Ia Ardense+24
- ~50 kilonovae Setzer+19, Andreoni+19 (+ ToO)



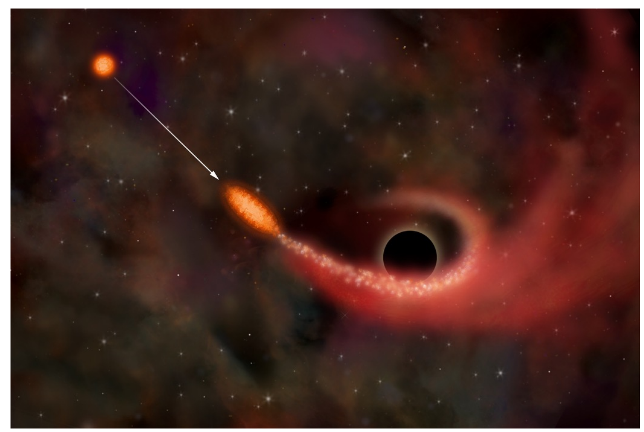
Recipe for ML for science in the time-domain

1. Representing our data:
   a. Feature extraction (e.g. Sánchez Sáez+21, Malanchev+21, etc.)
   b. Representation Learning - deep learning (e.g. Villar+21, Parker+24, etc.)



Recipe for ML for science in the time-domain

1. Representing our data:
    a. Feature extraction (e.g. Sánchez Sáez+21, Malanchev+21, etc.)
    b. Representation Learning - deep learning (e.g. Villar+21, Parker+24, etc.)
2. Apply ML for science!
    - Classification & Characterization
    - Redshift Estimation
    - Systematic Searches
    - Population studies & hypothesis testing
    - Anomaly detection
    - [YOUR SCIENCE CASE HERE] ...



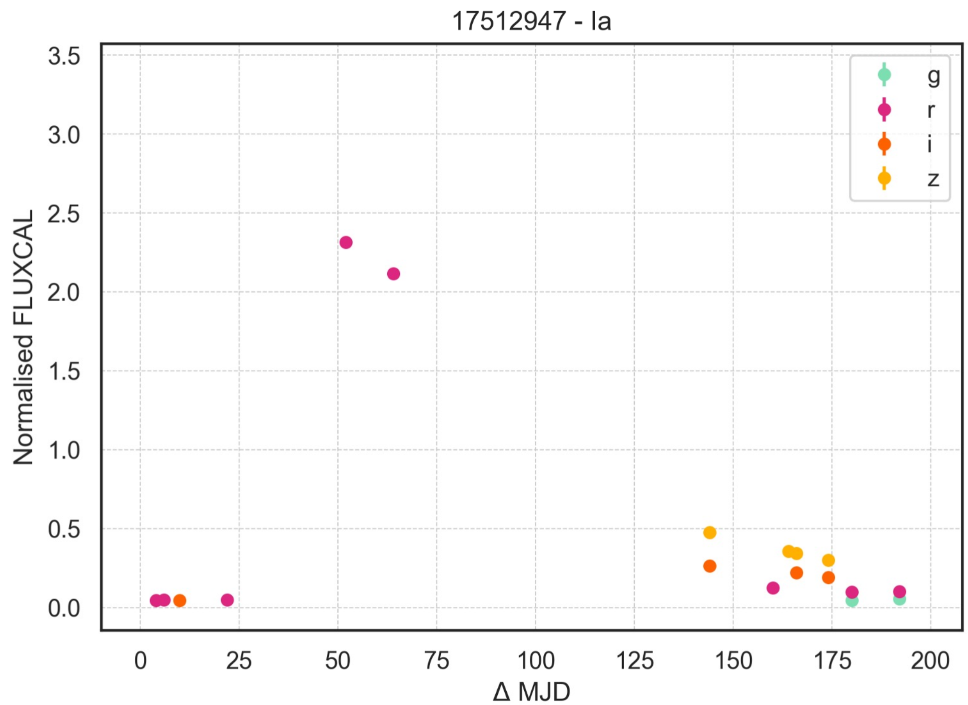
NASA/CXC/M.Weiss

PHOTOMETRIC LSST ASTRONOMICAL TIME-SERIES CLASSIFICATION CHALLENGE (PLASTICC)

M.A.L.L.O.R.N

Multiband light curves, but sparse

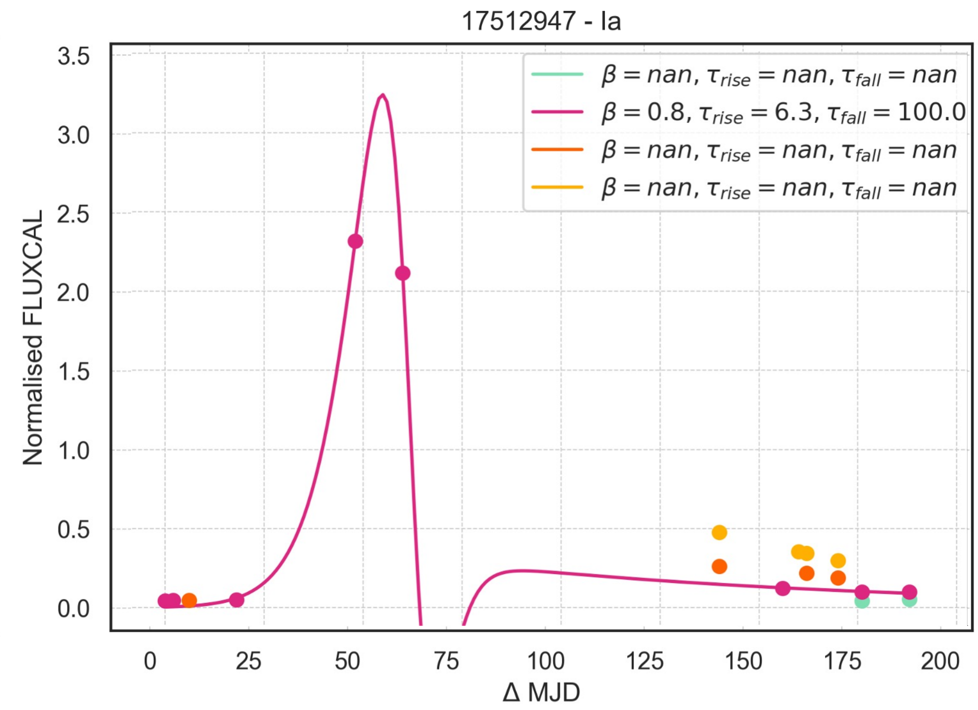
- But, a huge challenge is the sparsity of these light curves



Based on LSST OpSims
Baseline v5.0.0

Multiband light curves, but sparse

- But, a huge challenge is the sparsity of these light curves



Our fits fail if there's not enough points :((

Based on LSST OpSims Baseline v5.0.0

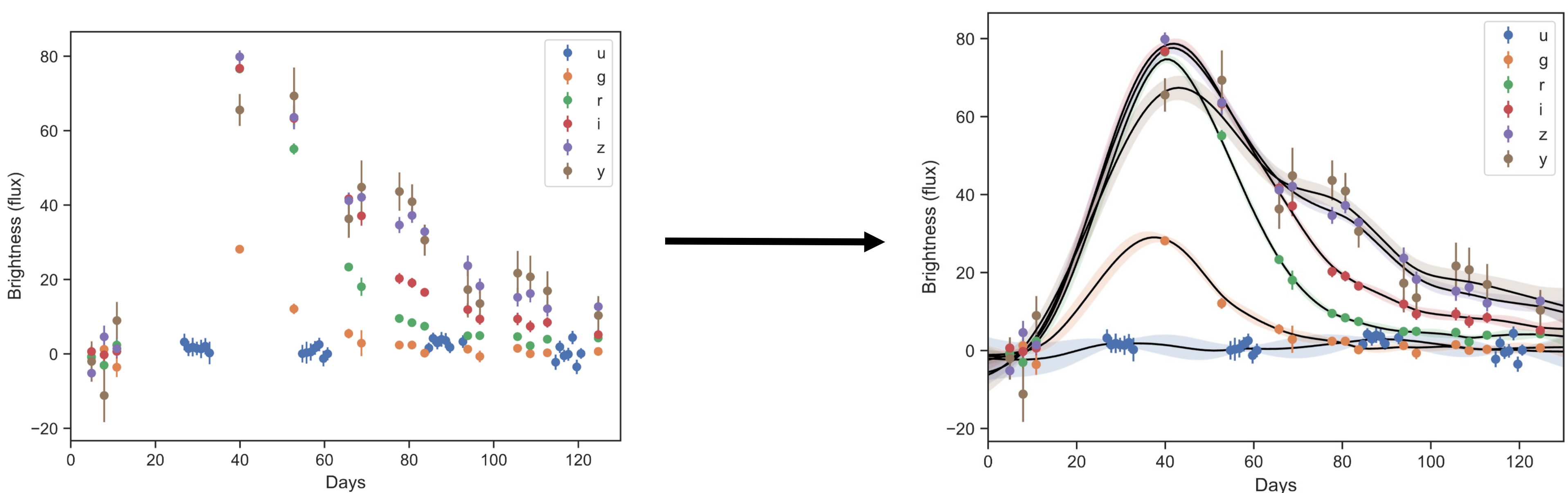
Recipe for ML for science in TVS

Can improve all downstream tasks below!!

0. Interpolate light curves!
1. Representing our data:
    a. Feature extraction (e.g. Sánchez Sáez+21, Malanchev+21, etc.)
    b. Representation Learning - deep learning (e.g. Villar+21, Parker+24, etc.)
2. Apply ML for science!
    - Classification & Characterization
    - Redshift Estimation
    - Systematic Searches
    - Population studies & hypothesis testing
    - Anomaly detection
    - [YOUR SCIENCE CASE HERE] ...

GPs as interpolators

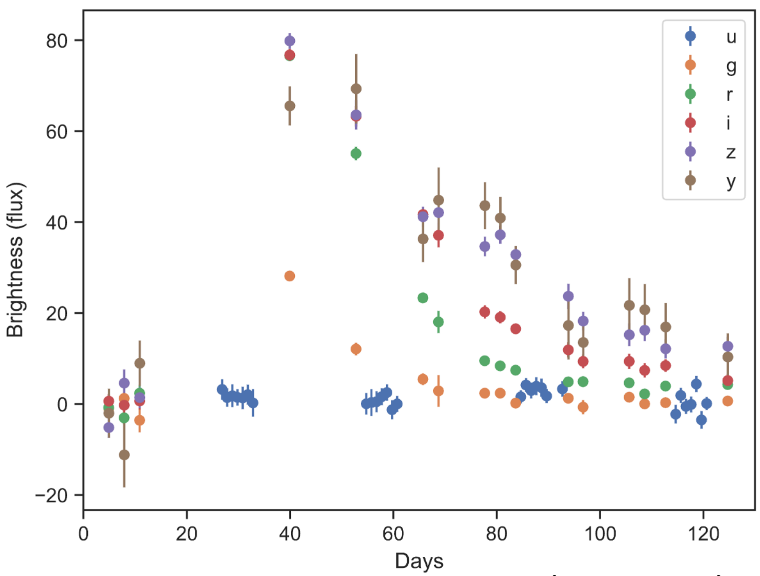
- Gaussian Processes to interpolate light curves and then feature extraction [e.g, **Boone+19**] or neural nets [e.g, Villar+20, Qu+21, etc.]

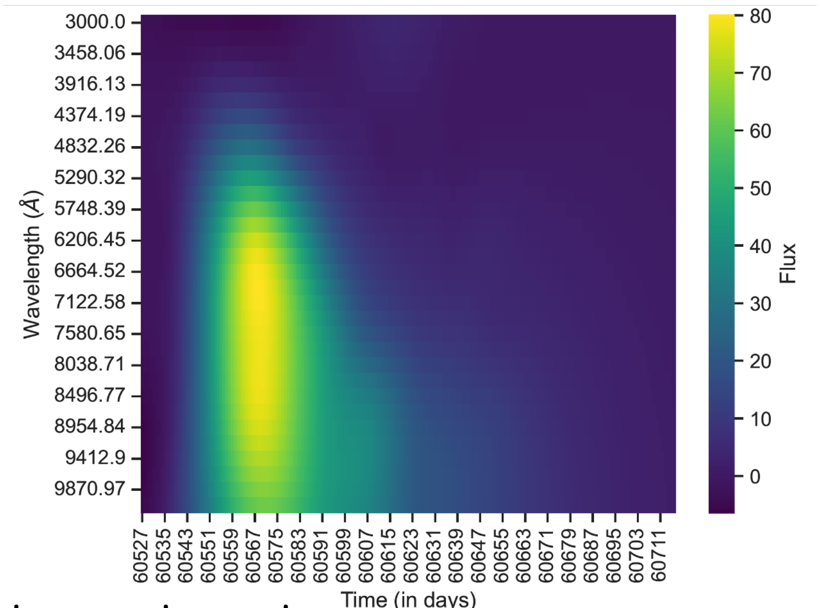


Interpolate over time

GPs as interpolators

- Gaussian Processes to interpolate light curves and then feature extraction [e.g, Boone+19] or neural nets [e.g, Villar+20, **Qu+21**, etc.]





Interpolate over time and wavelength

GPs as interpolators

- Gaussian Processes to interpolate light curves and then feature extraction [e.g, Boone+19] or neural nets [e.g, Villar+20, Qu+21, etc.]

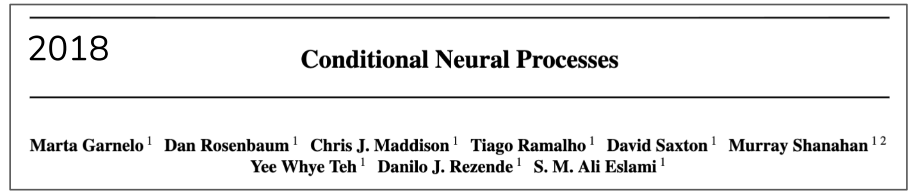
But GPs require you to choose a kernel function to model covariance

$$K_{3/2}(x_1, x_2; \alpha, l) = \alpha^2 \left(1 + \sqrt{3\frac{(x_1 - x_2)^2}{l^2}}\right) \exp\left(-\sqrt{3\frac{(x_1 - x_2)^2}{l^2}}\right)$$

Matern 3/2 kernel

But no single kernel will work for everything!!

The Neural Process Family

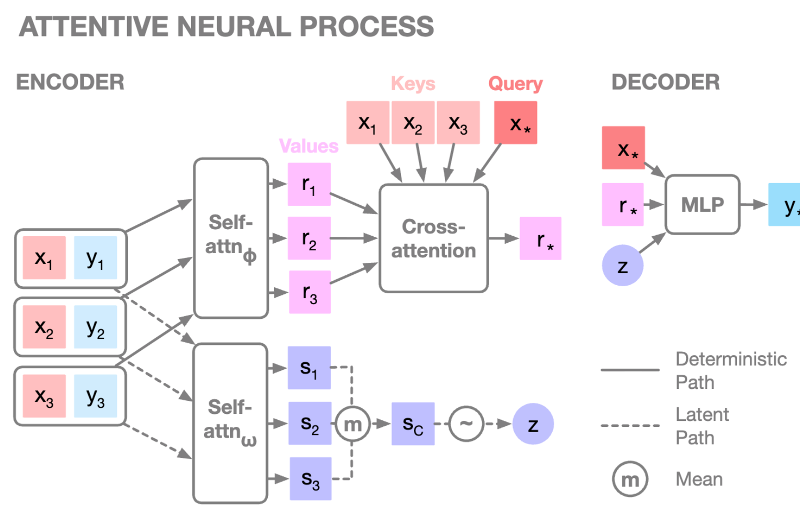

2018

Conditional Neural Processes

Marta Garnelo[1] Dan Rosenbaum[1] Chris J. Maddison[1] Tiago Ramalho[1] David Saxton[1] Murray Shanahan[1 2] Yee Whye Teh[1] Danilo J. Rezende[1] S. M. Ali Eslami[1]

Neural Processes (NPs) are a family of models that learn distributions over functions (like GPs) with the help of neural networks.

- NPs combine the best parts of the NN & GP approaches
  - NNs: Very quick, flexible and data driven: exceptional at learning patterns
  - GPs: Produce distributions over functions with uncertainties
- Akin to a GP "learning" the kernel with a neural network

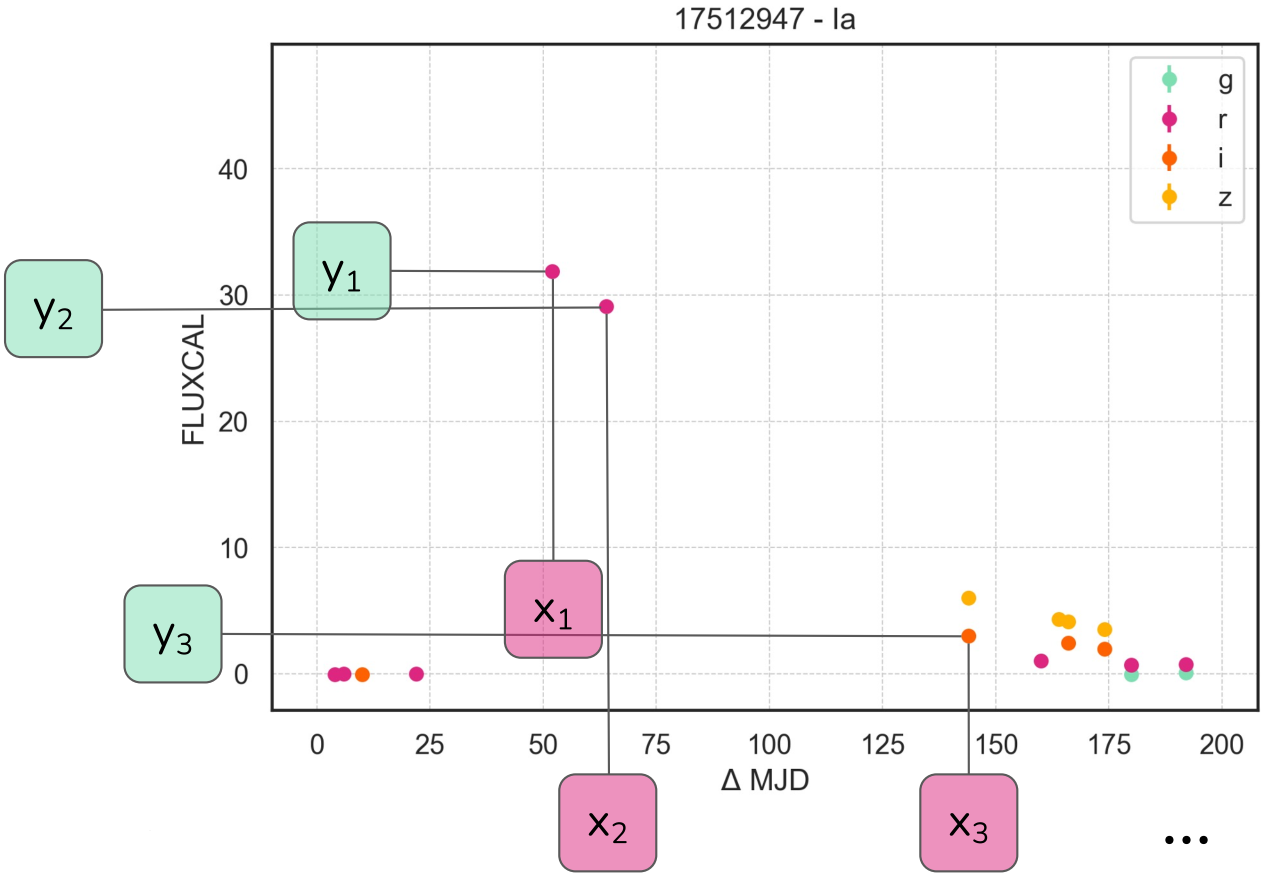


Credit: Kim+2019

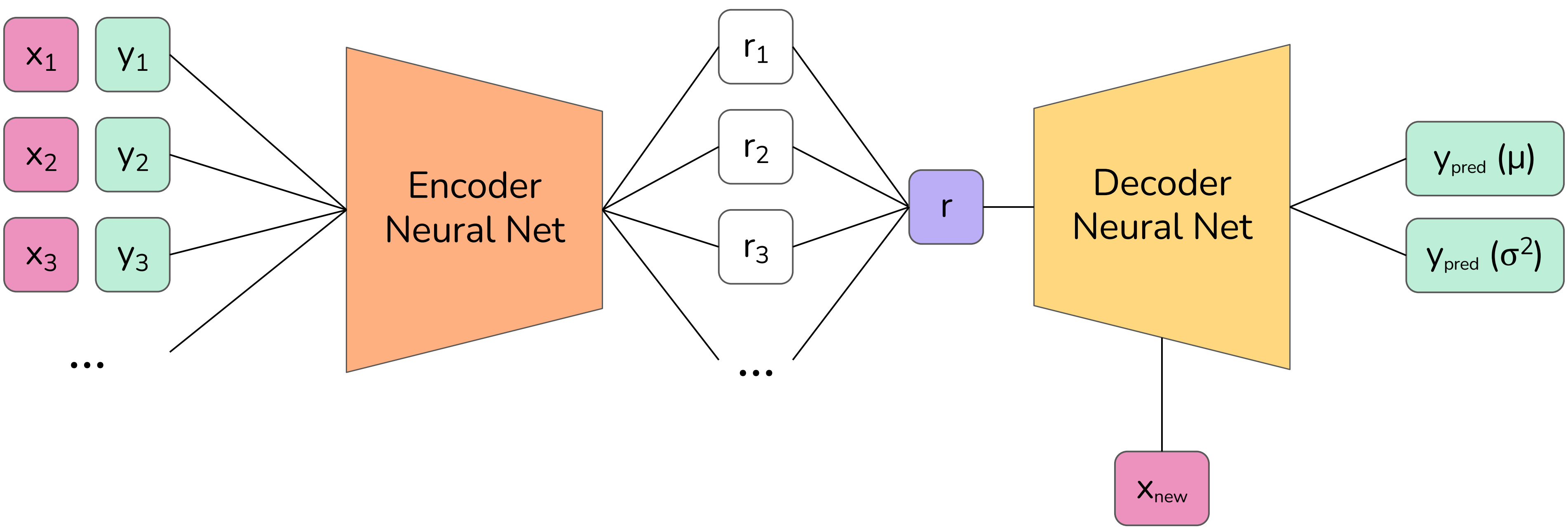
Use NPs as a better preprocessing step for **all** LSST light curves

- Focusing on transients here, but methodology is data-driven ... should work for any light curve!

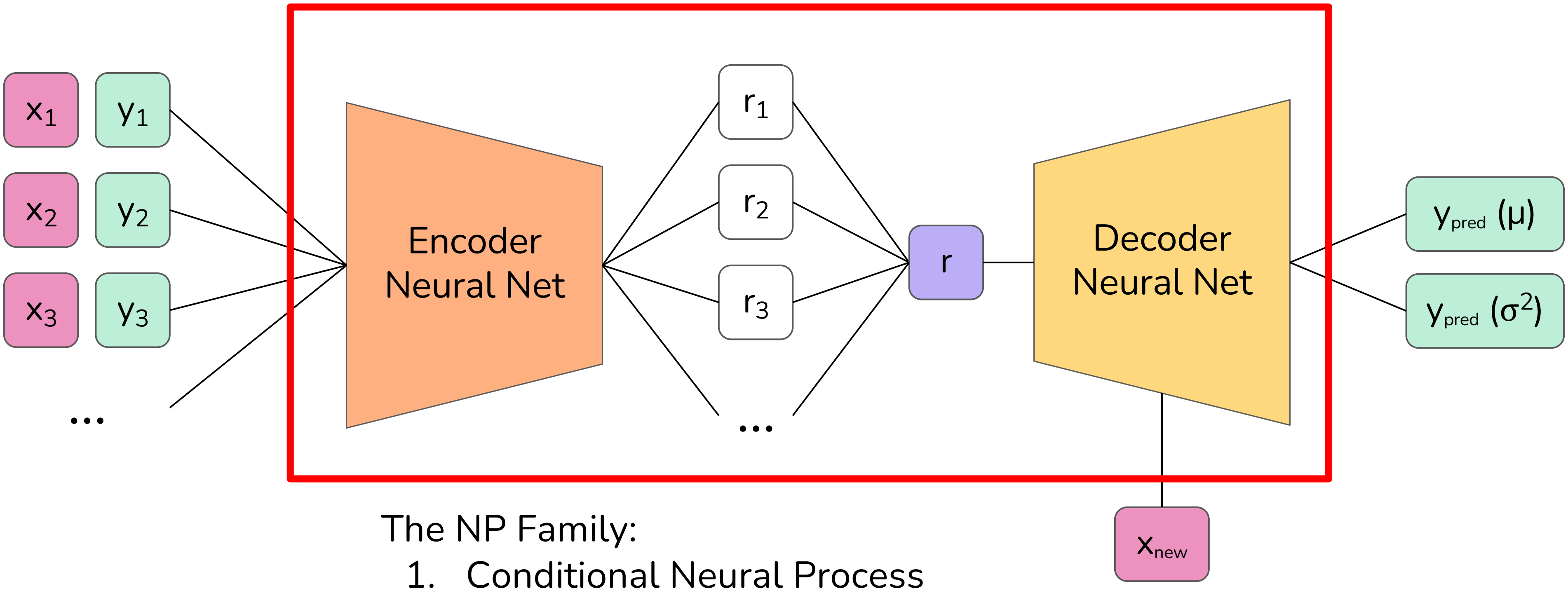
Neural Processes



Neural Processes



Neural Processes



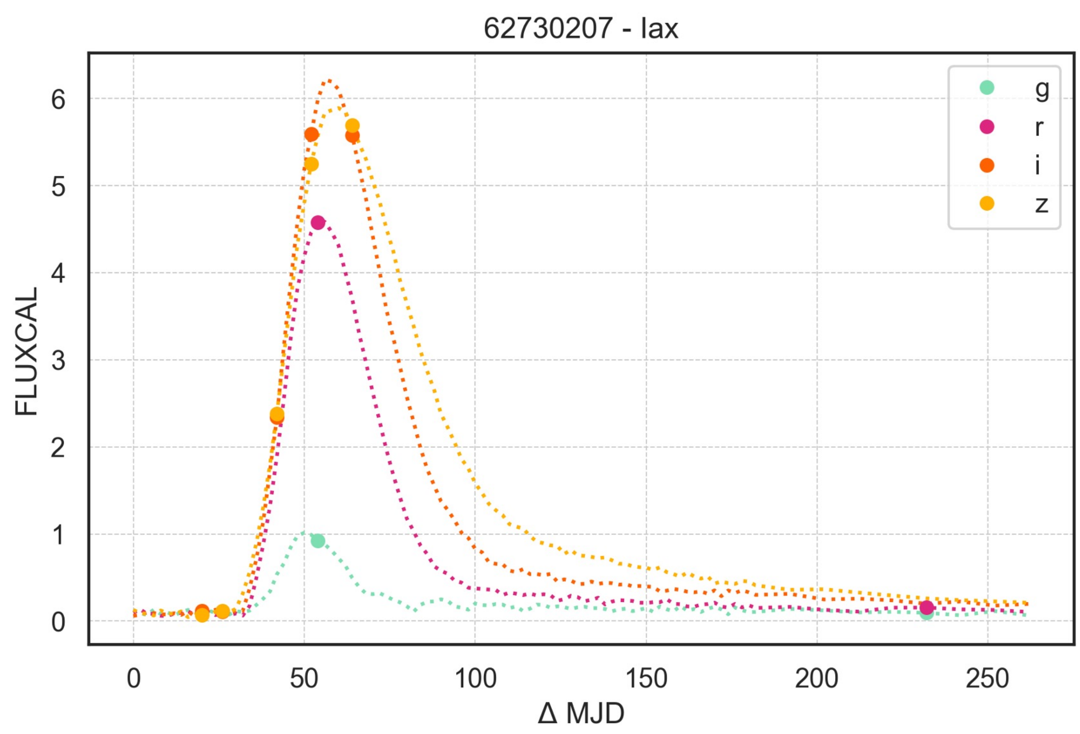
The NP Family:

1. Conditional Neural Process
2. (Latent) Neural Process
3. **Attentive Neural Process**

1. SNIax example

How well does the ANP do on an entirely unseen light curve?

- On an LSST-like cadence light curve
- And let's keep track of the true value



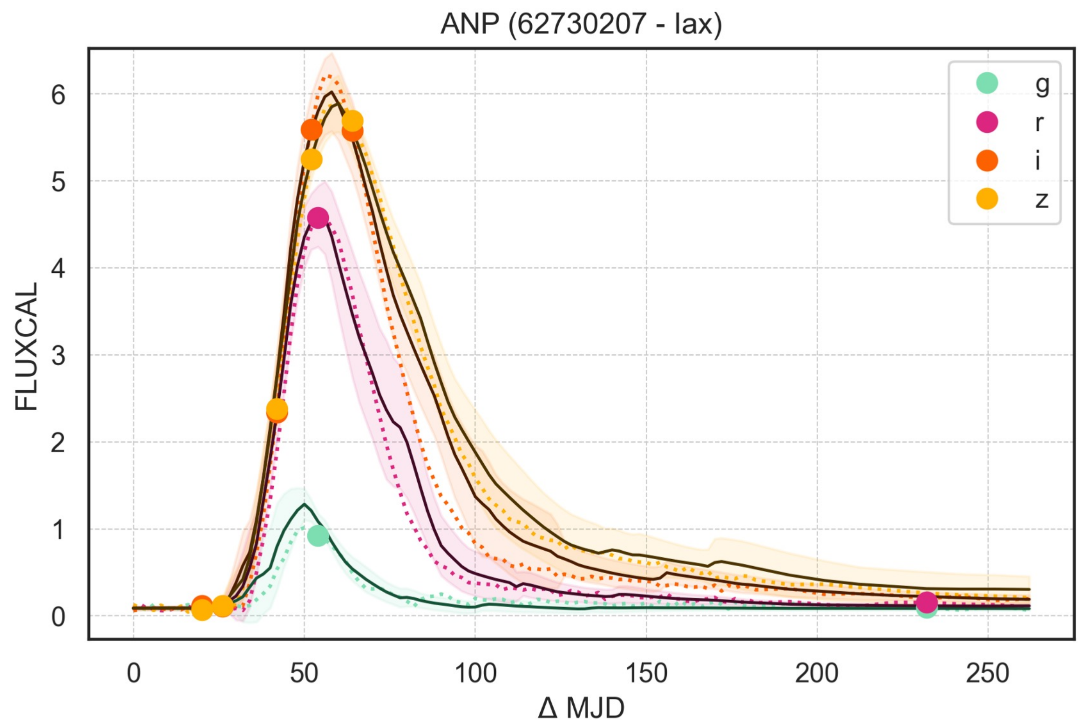
Based on LSST OpSims Baseline v5.0.0

1. ANP over time: e.g. SNIax

How well does the ANP do on an entirely unseen light curve?

- On an LSST-like cadence light curve
- And let's keep track of the true value

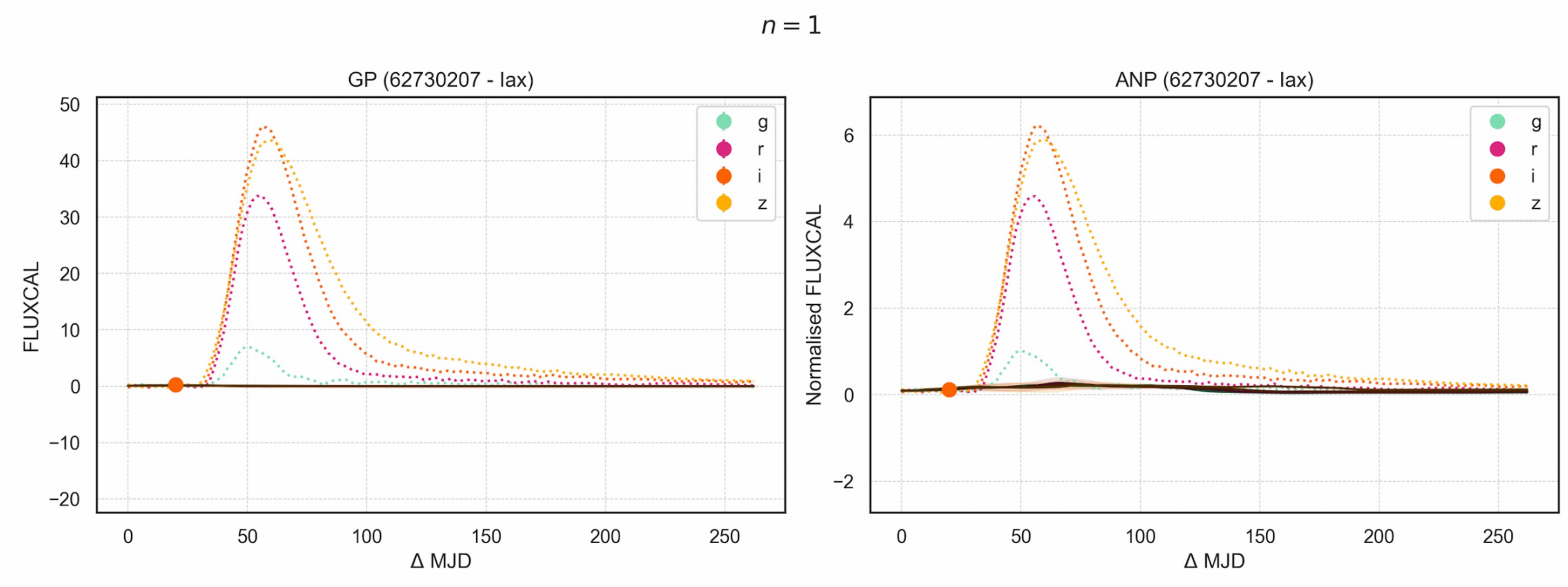
$n = 14$



Based on LSST OpSims Baseline v5.0.0

(Chaini+26, in-prep)

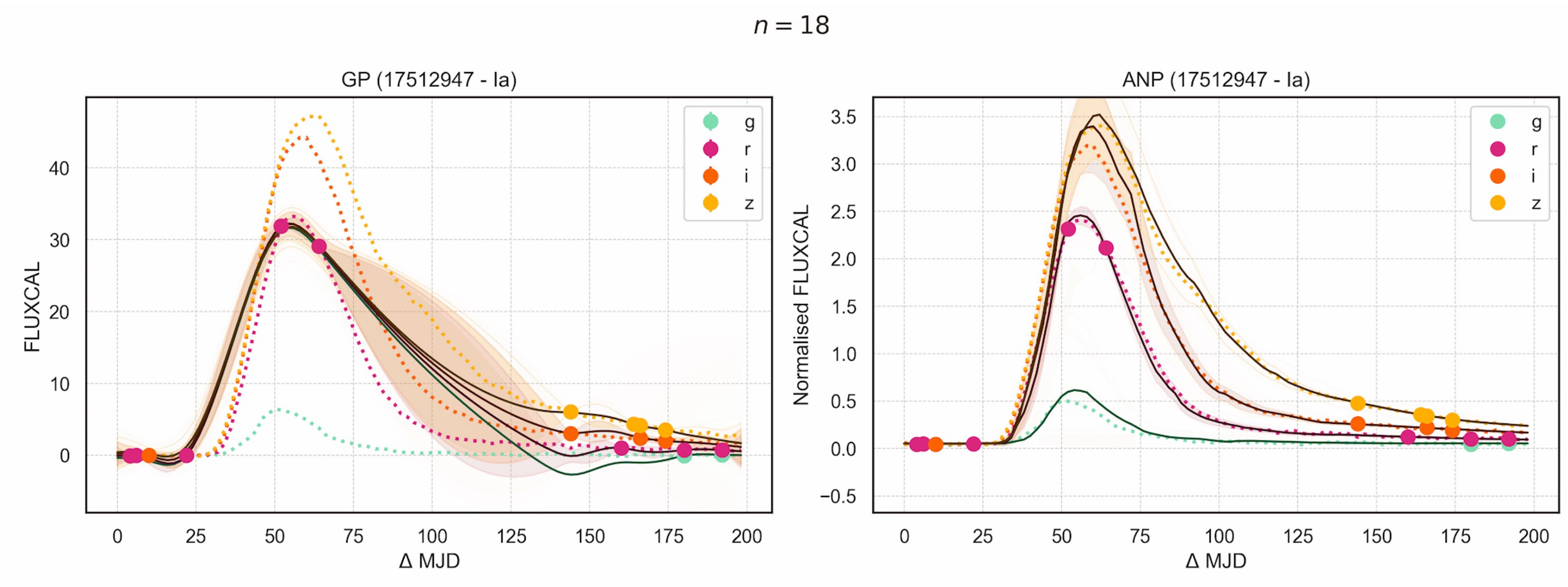
1. ANP v/s GP over time: e.g. SNIax



Based on LSST OpSims Baseline v5.0.0

(Chaini+26, in-prep)

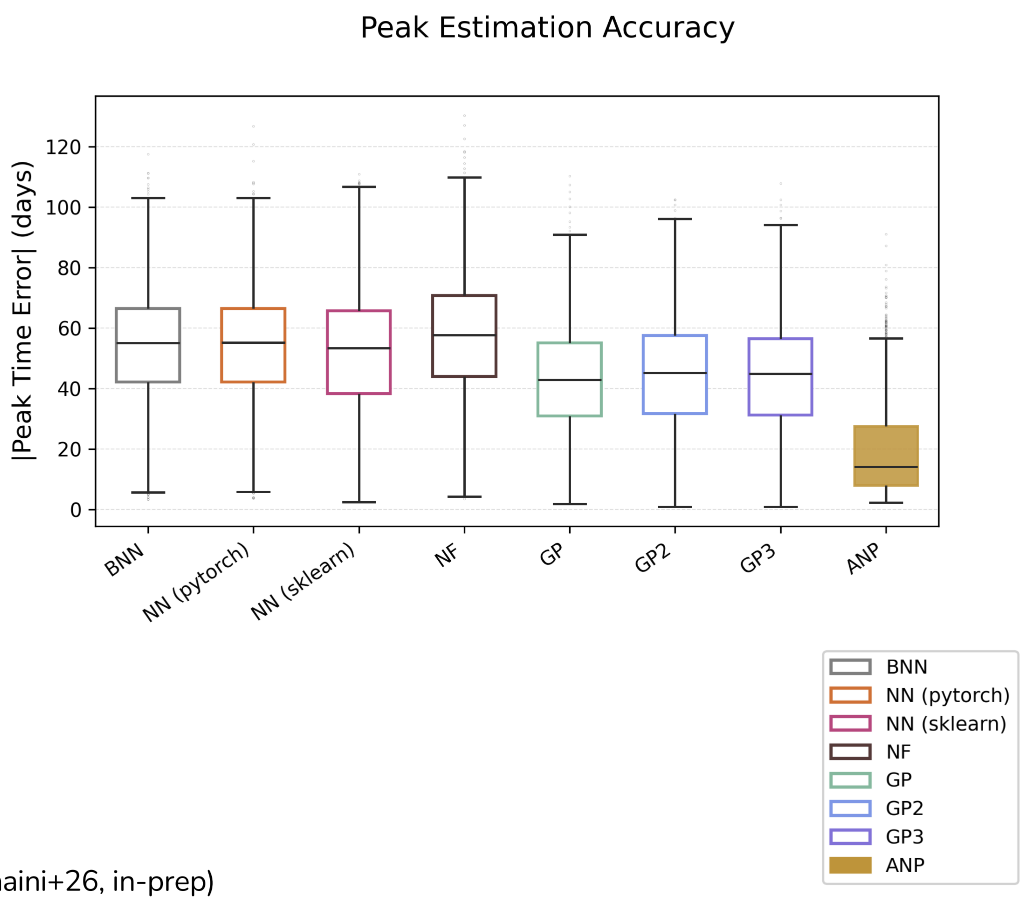
2. ANP v/s GP over time: e.g. SNIa

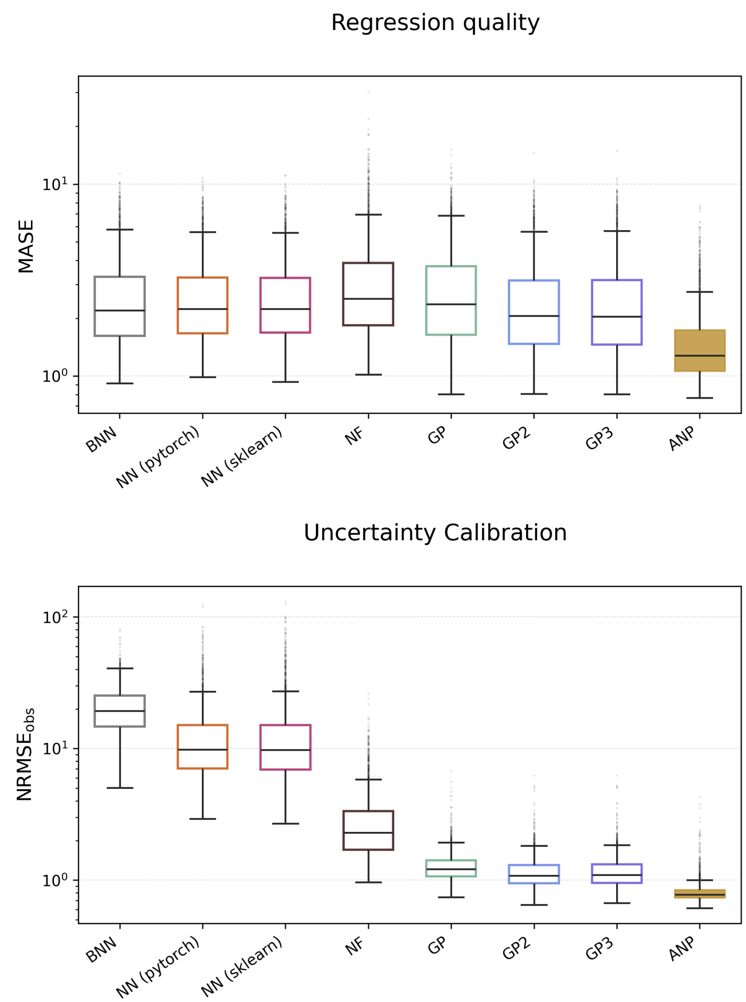


Based on LSST OpSims Baseline v5.0.0

(Chaini+26, in-prep)

3. ANP compared to SOTA

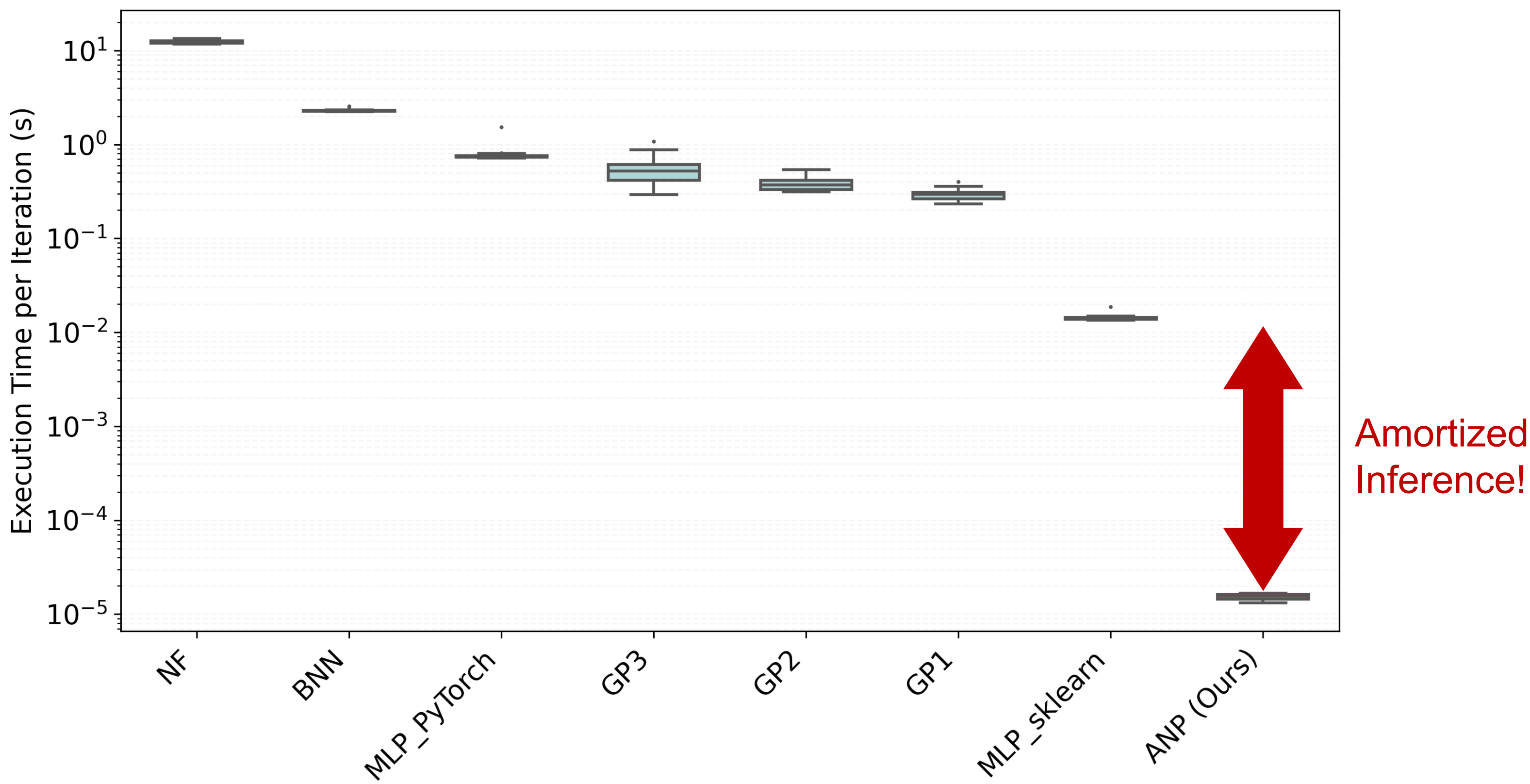




(Chaini+26, in-prep)

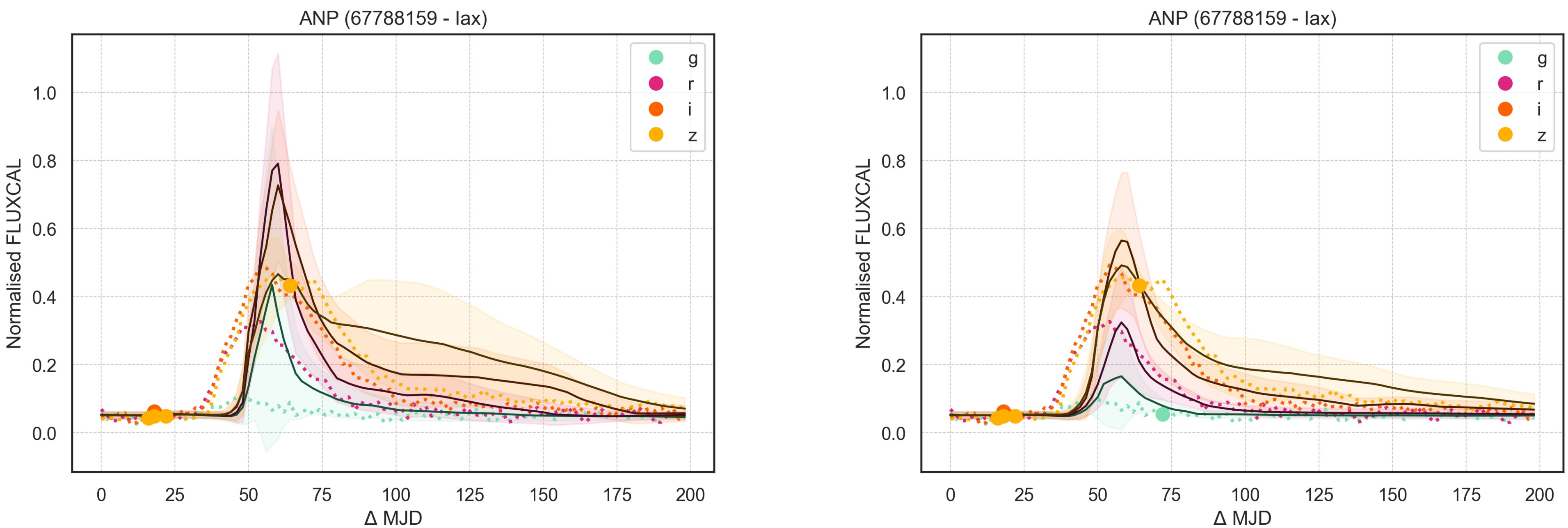
4. ANP : amortized inference

*excludes training time



(Chaini+26, in-prep)

5. Multiwavelength hope: ANP



Just observing one extra point in a new filter improves the prediction dramatically!

So a small no. of follow-up / synergistic points in different filters may aid a lot!!!

(Chaini+26, in-prep)

Summary

sidchaini / **keras-neural-processes**

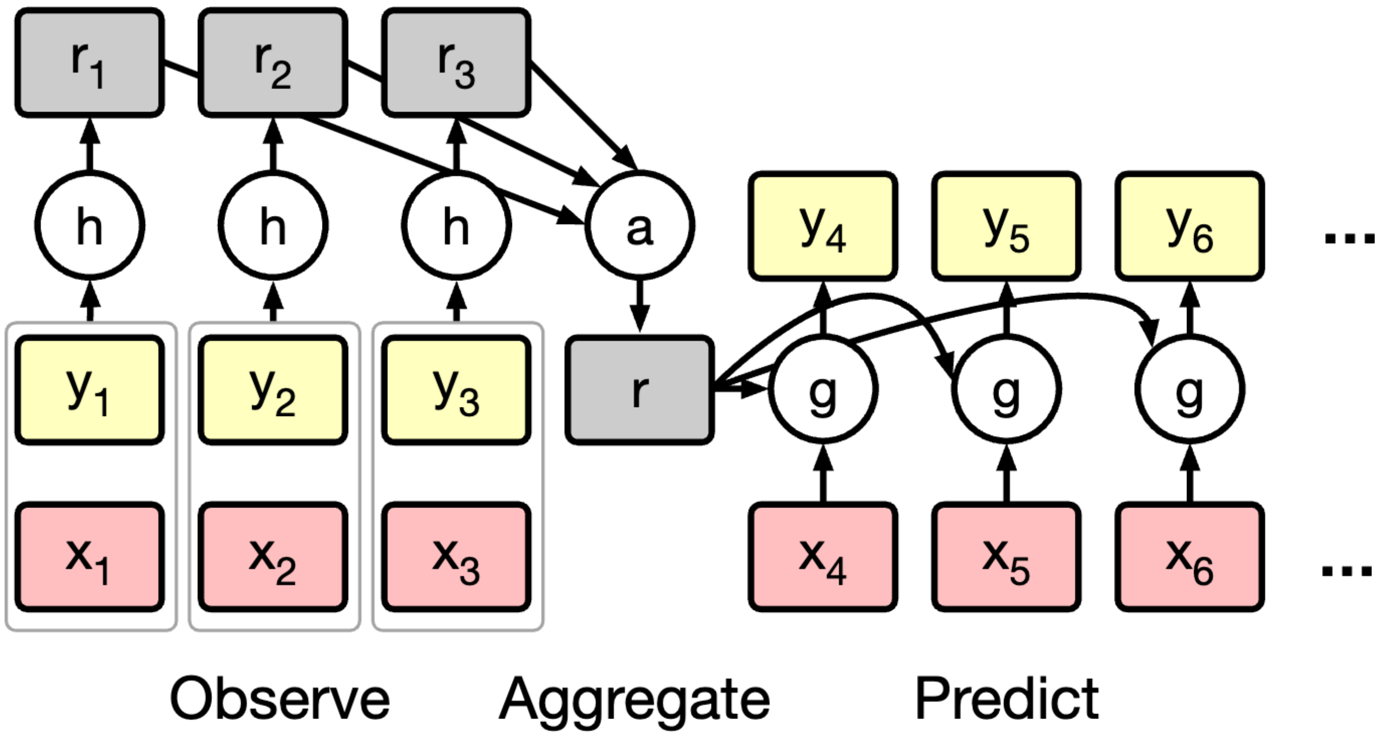
1. Neural Processes may be the answer to an unsupervised interpolator across multiband (and multi-survey?) light curve data
2. Amortized inference makes it quick!
3. Next step: moving from simulations to real alert streams
4. If this is interesting, let me know! (chaini@udel.edu)

AST#2511639

Thank you!

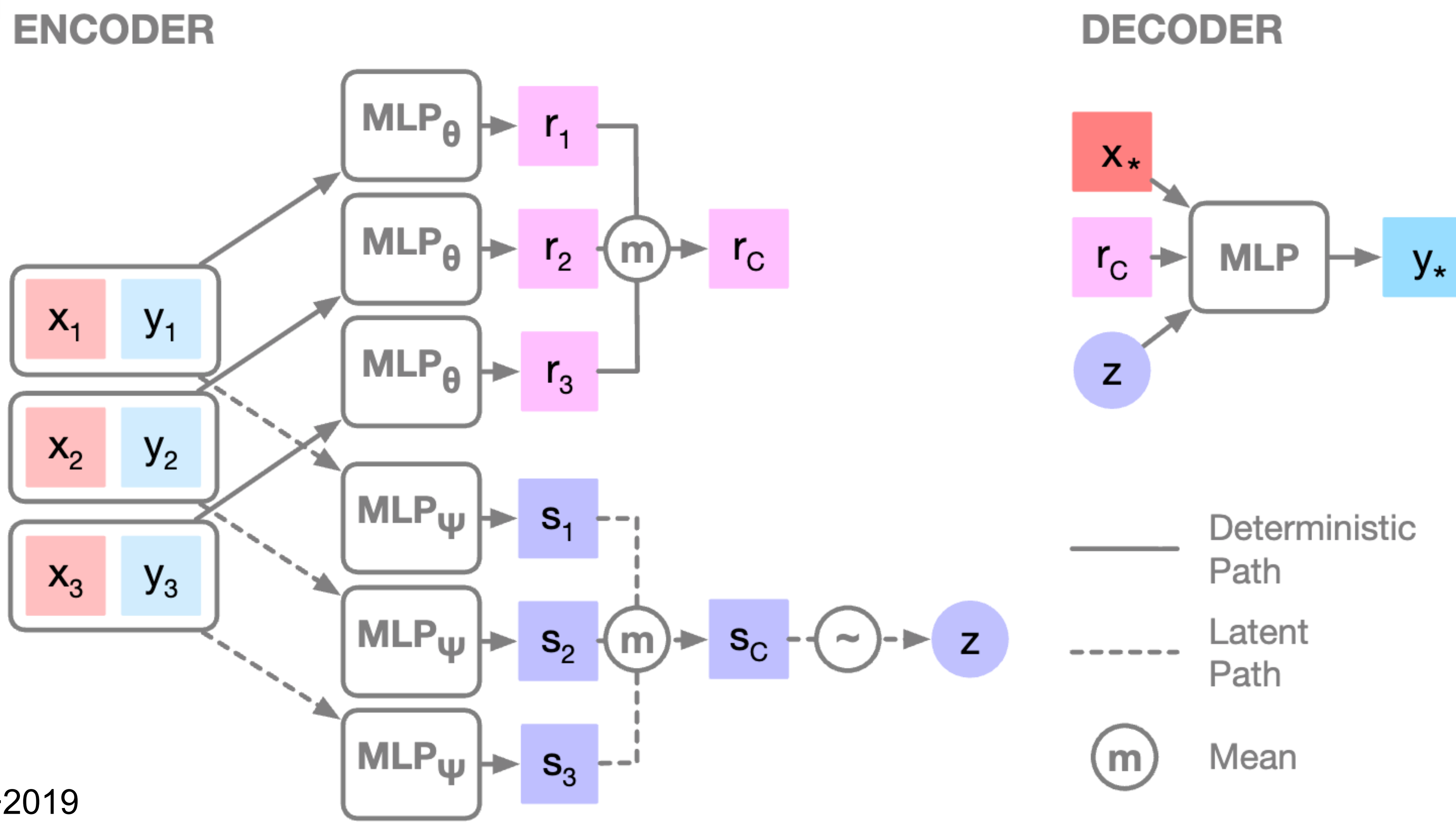
Siddharth Chaini
sidchaini.github.io

Conditional Neural Process



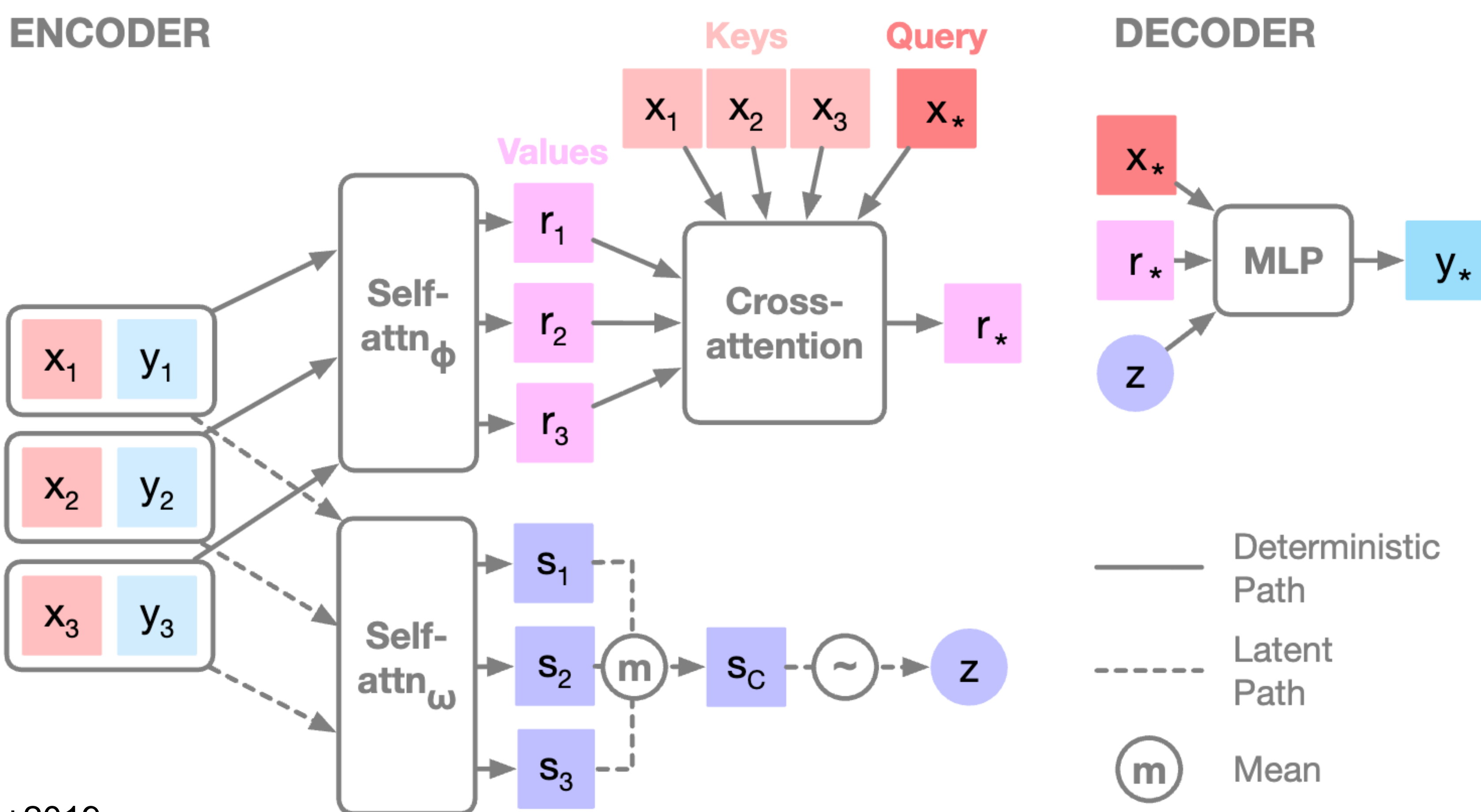
Garnelo+2018

NEURAL PROCESS



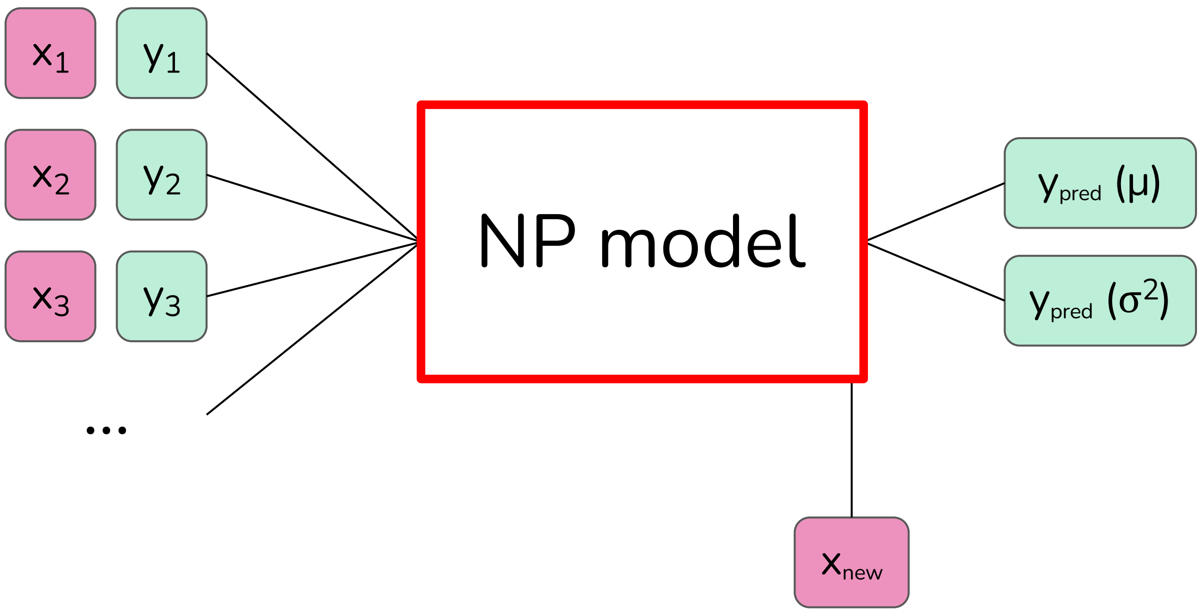
Kim+2019

ATTENTIVE NEURAL PROCESS

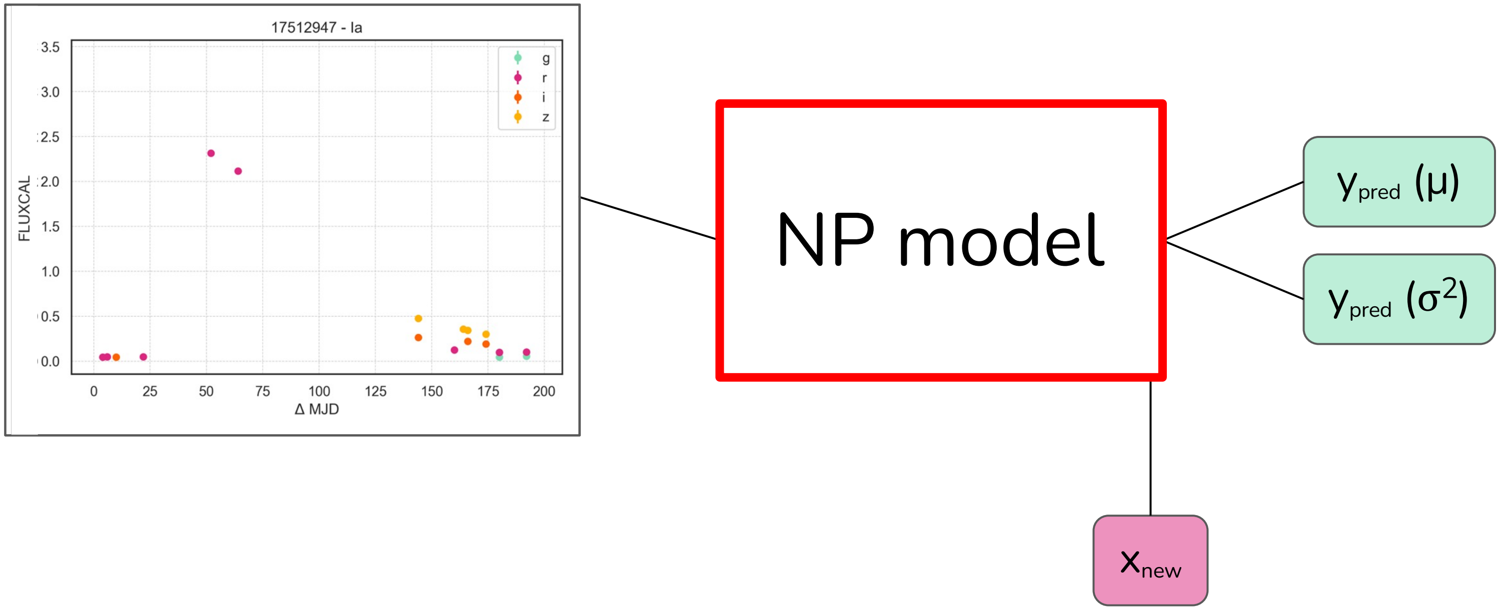


Kim+2019

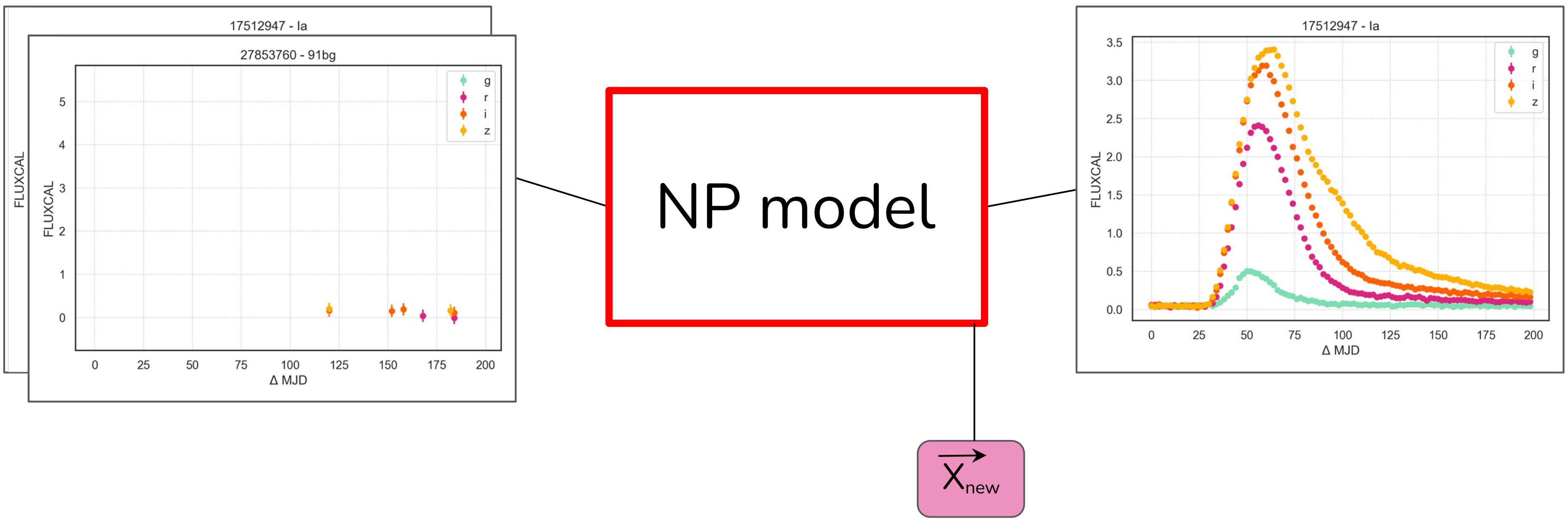
How to train your NP



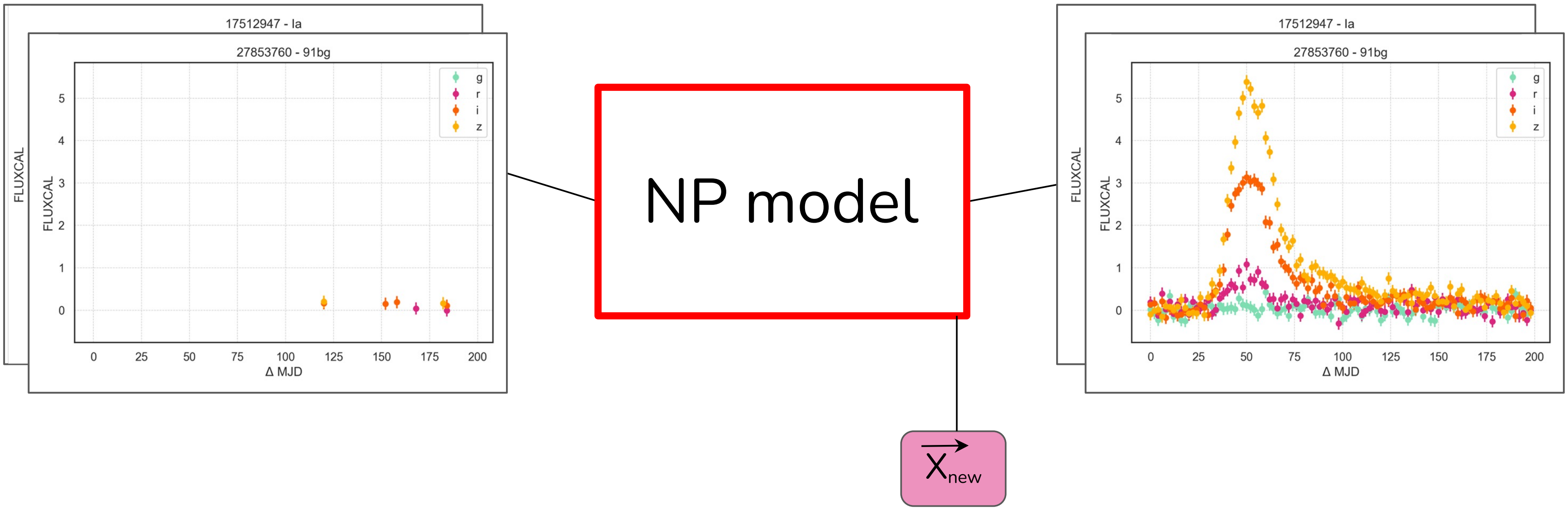
How to train your NP



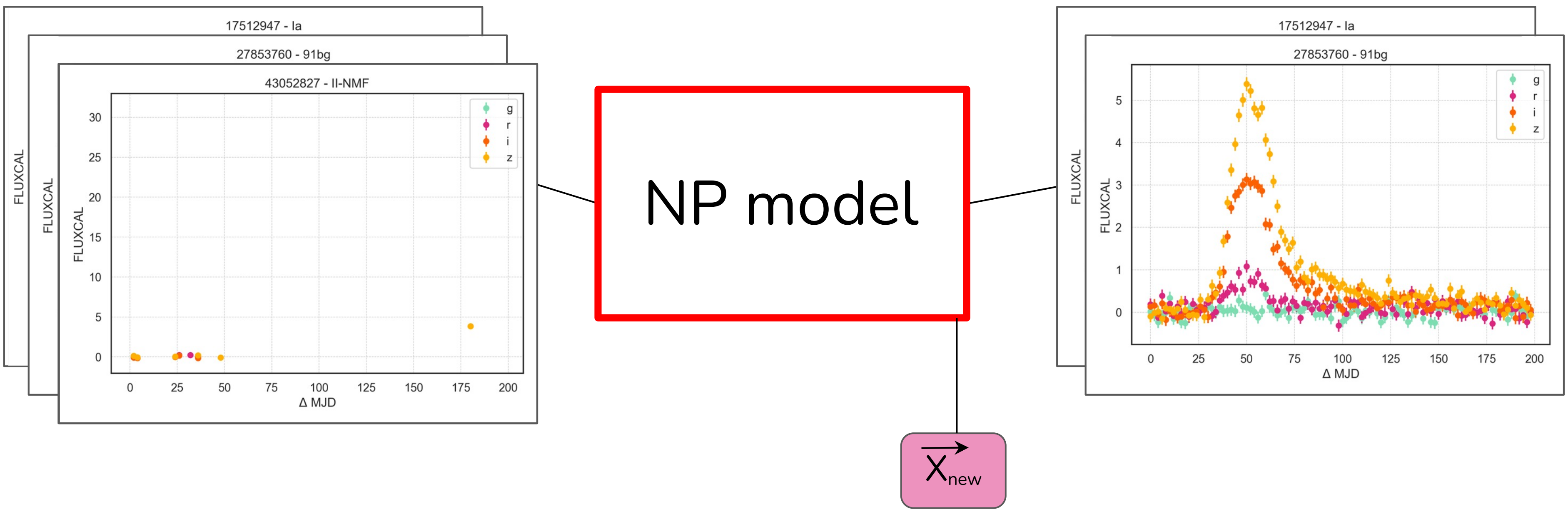
How to train your NP



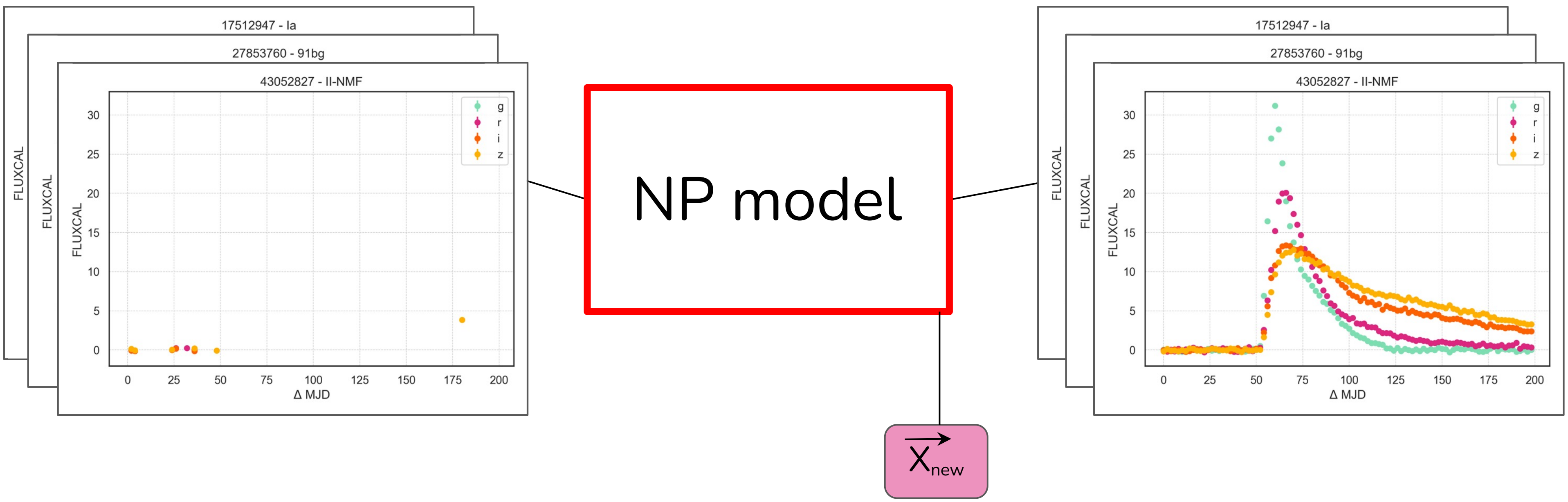
How to train your NP



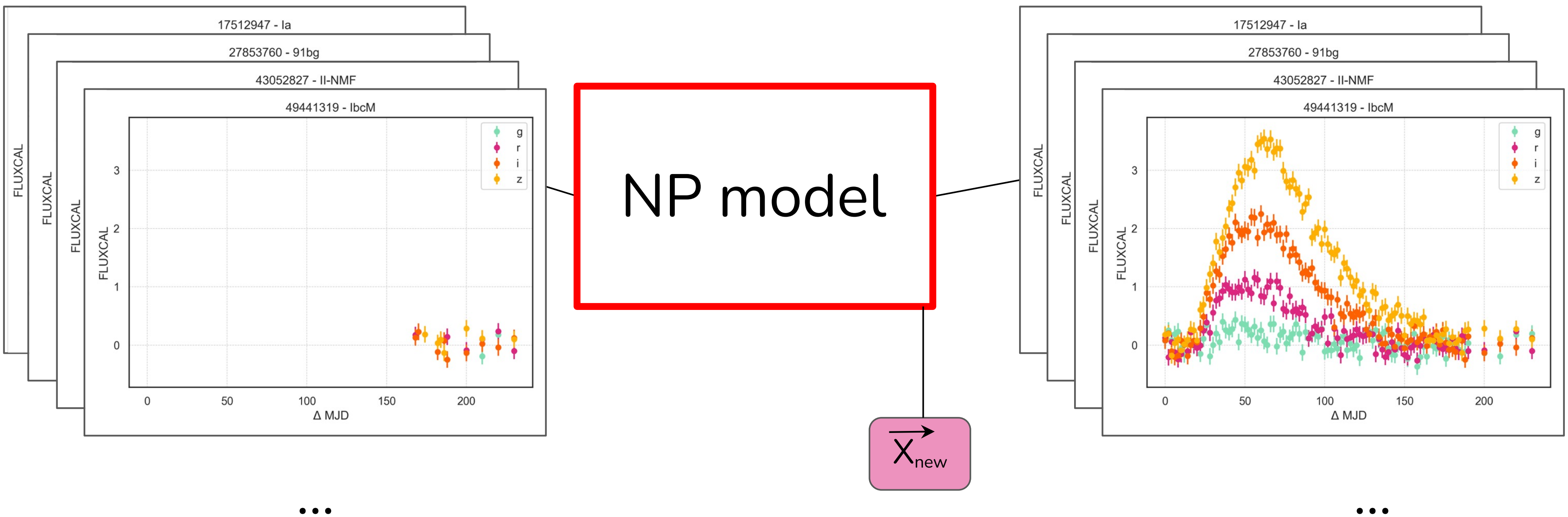
How to train your NP



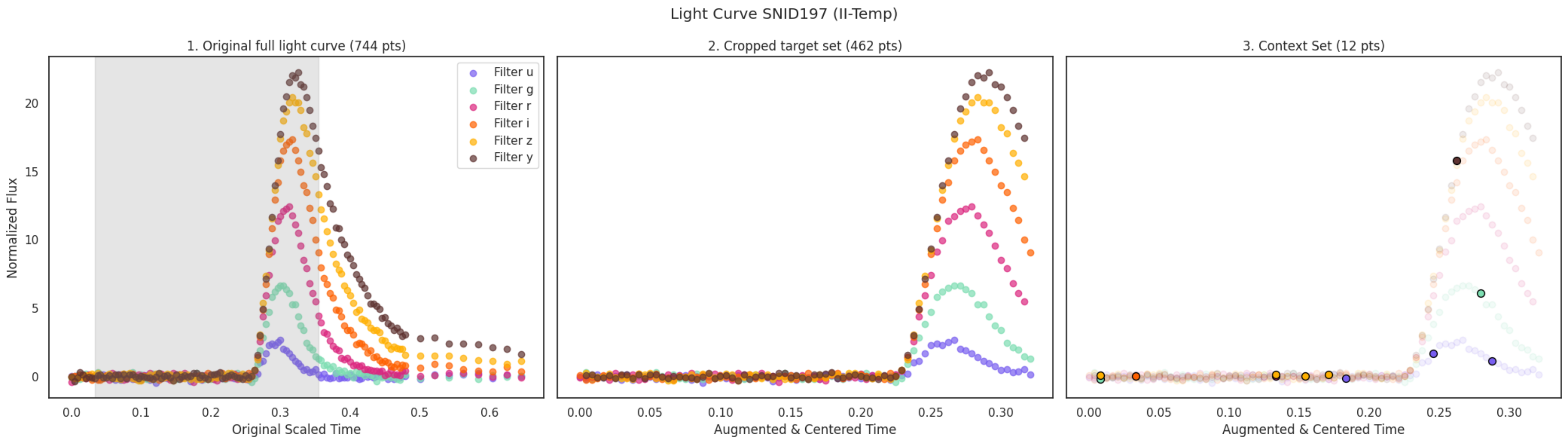
How to train your NP



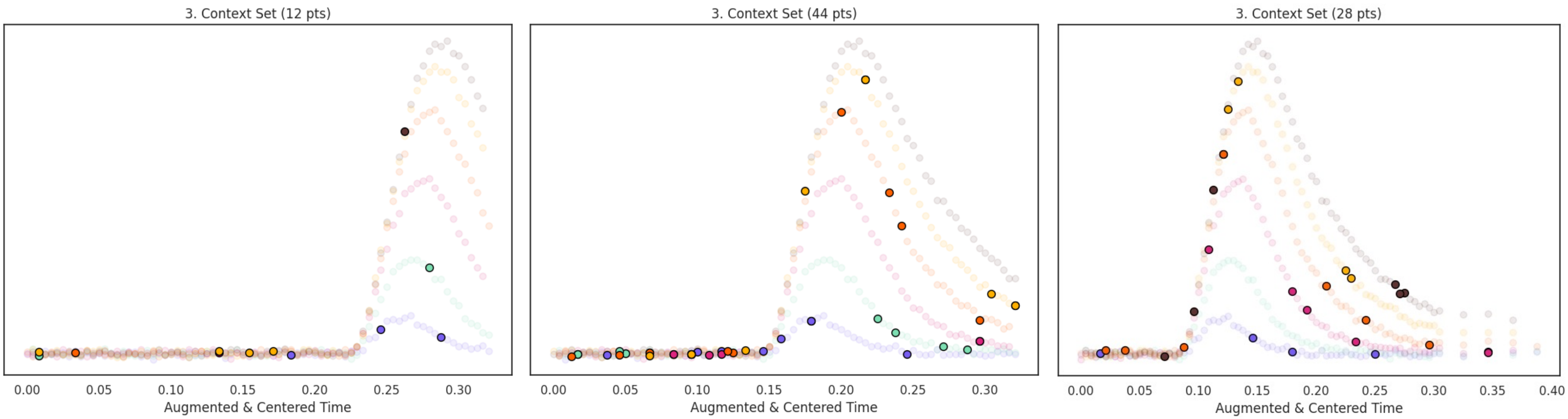
How to train your NP



How to train your NP (more details)



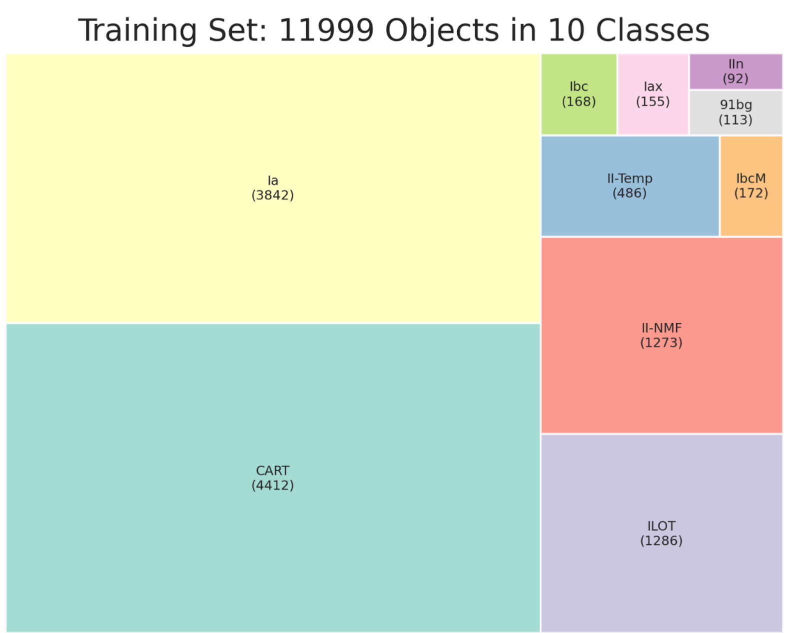
How to train your NP (more details)

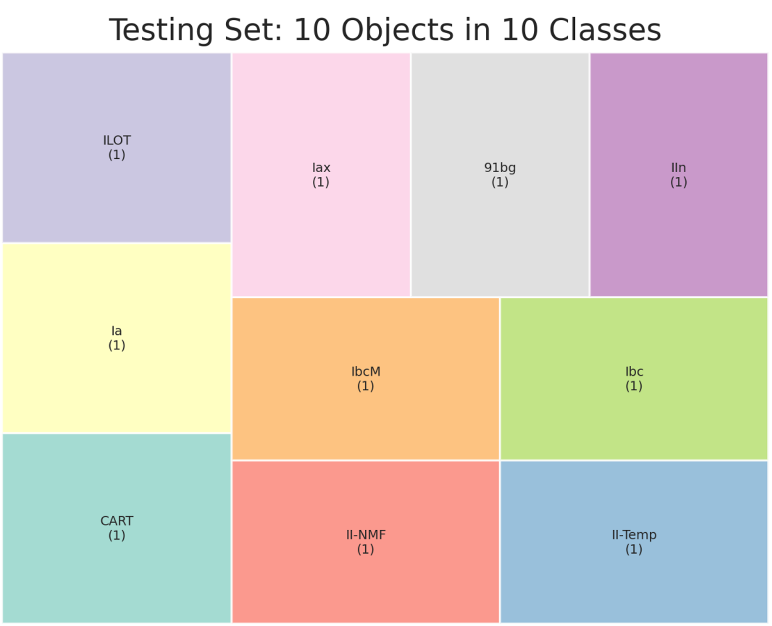


Meta-learn a distribution over functions

How to train your NP

In this example here, my dataset consists of SNANA simulated light curves from PLAsTiCC (Thanks to the RESSPECT team and Emille Ishida!!)





POC

(preliminary results, paper in-prep)