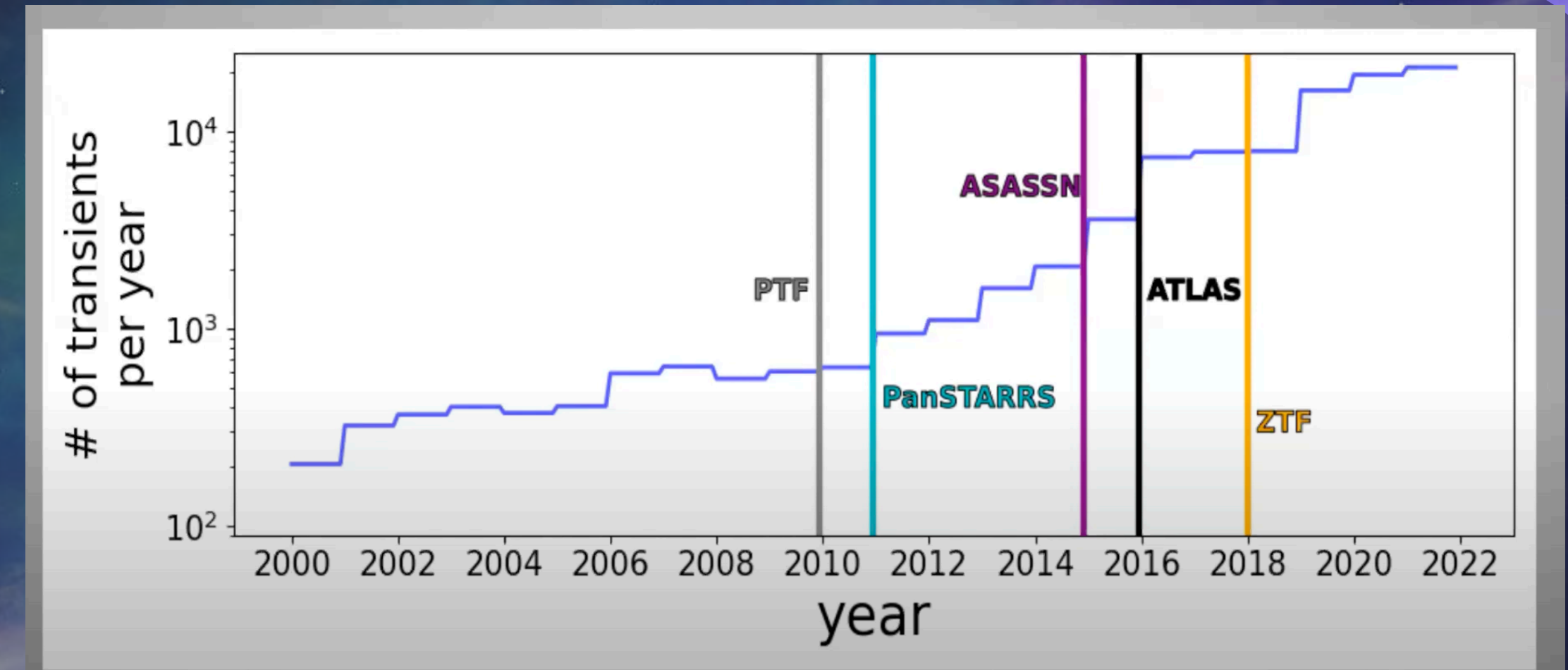


REDBACK – A BAYESIAN INFERENCE SOFTWARE PACKAGE FOR TRANSIENTS

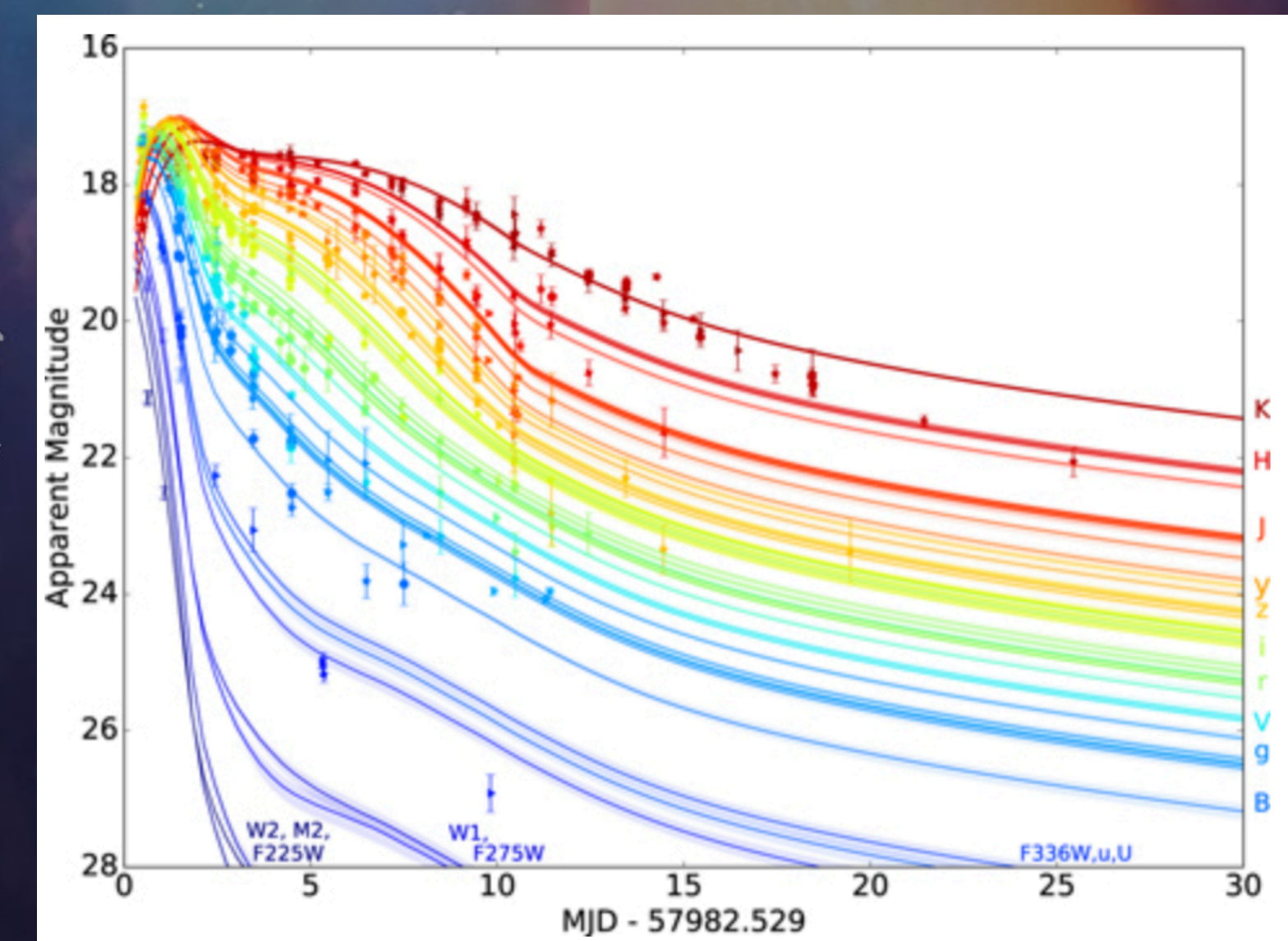
NIKHIL SARIN ++



- ▶ The number of transient observations is exploding.
- ▶ Accumulating large samples, and rate is not slowing down with e.g., Einstein Probe, Roman, Rubin, Ultrasat, etc
- ▶ But what can we actually learn from the observations?
 - ▶ Let the gospel of Reverend Bayes guide us to the answer...

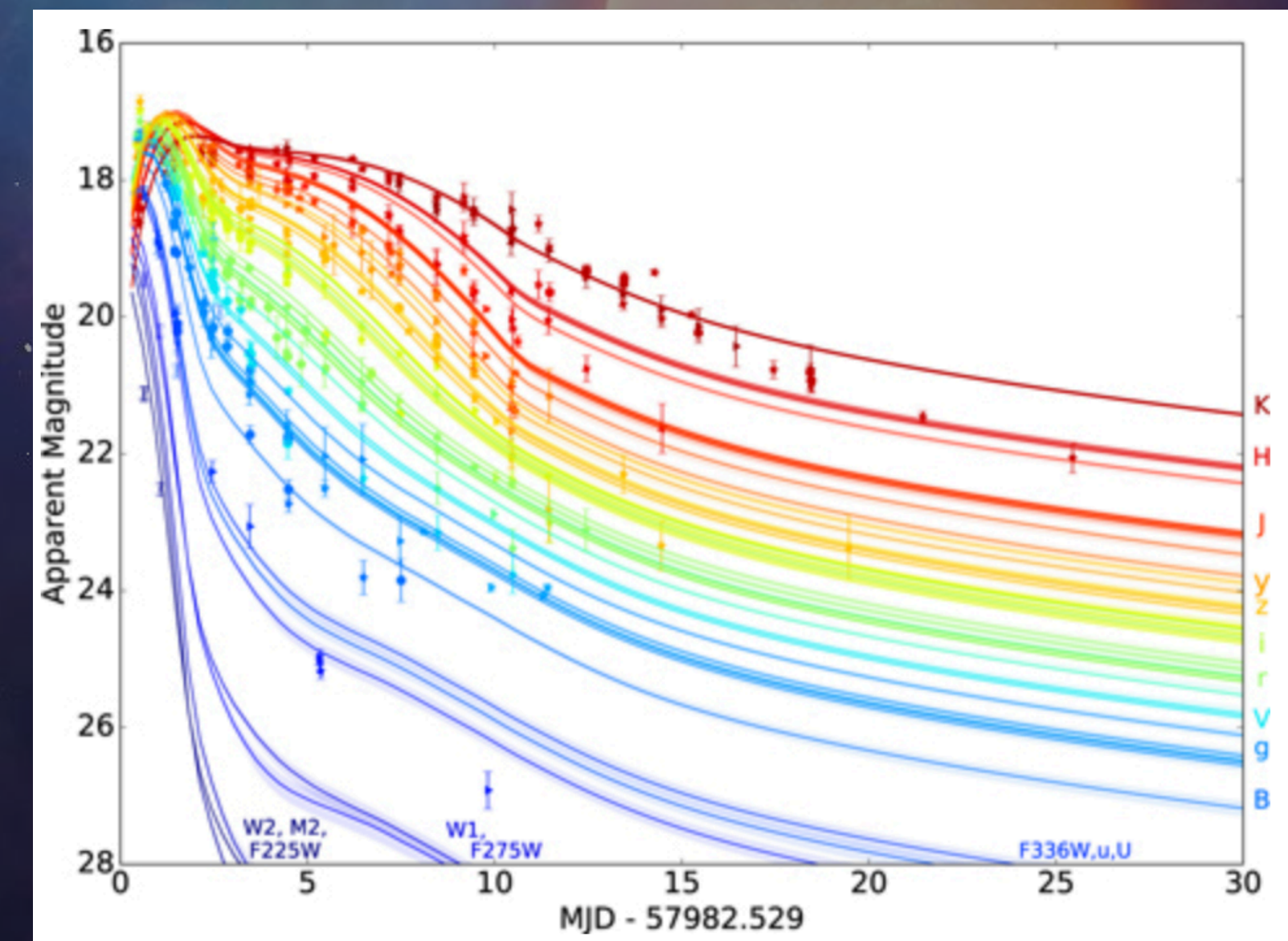
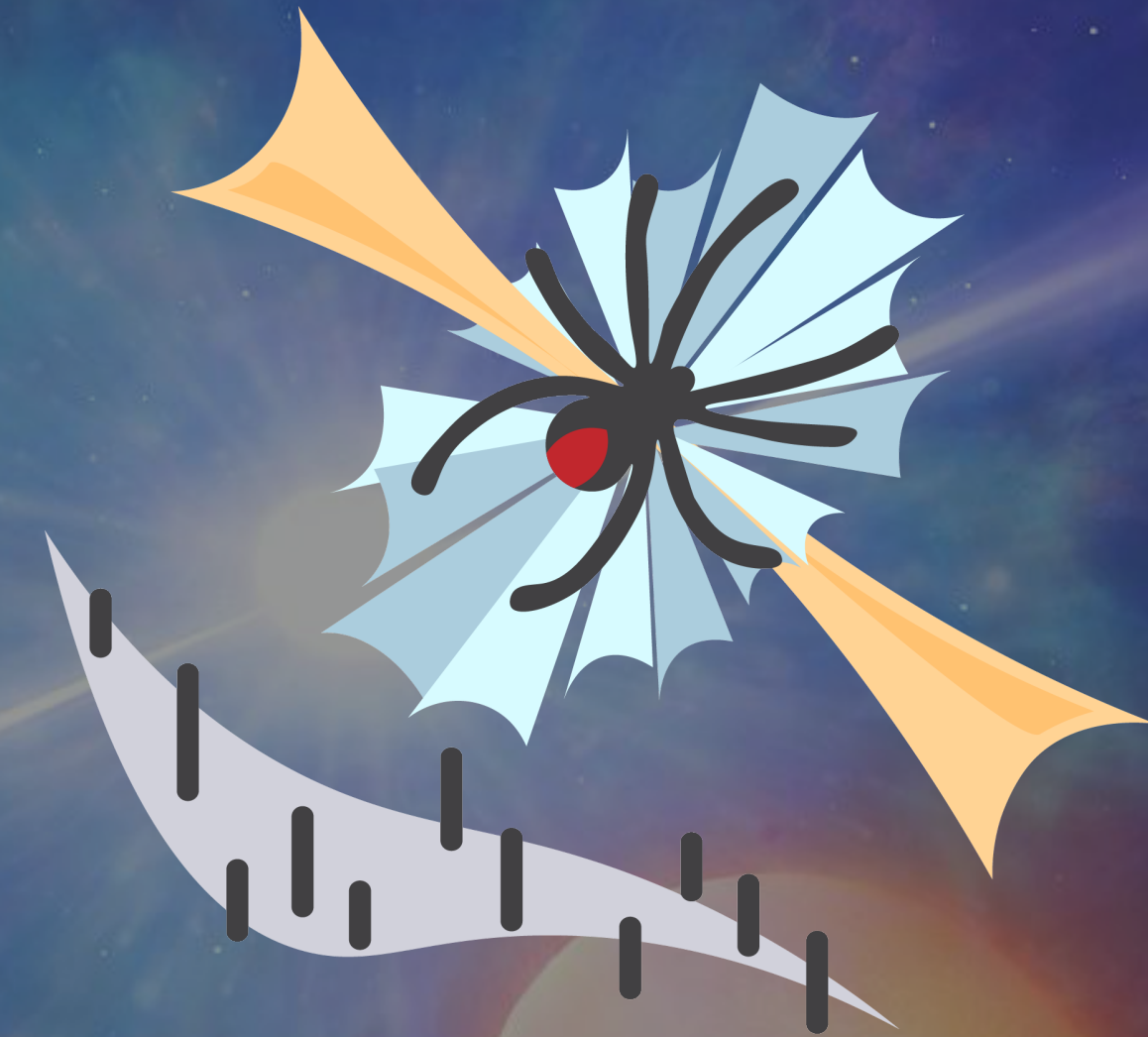


Credit: Michael Tucker

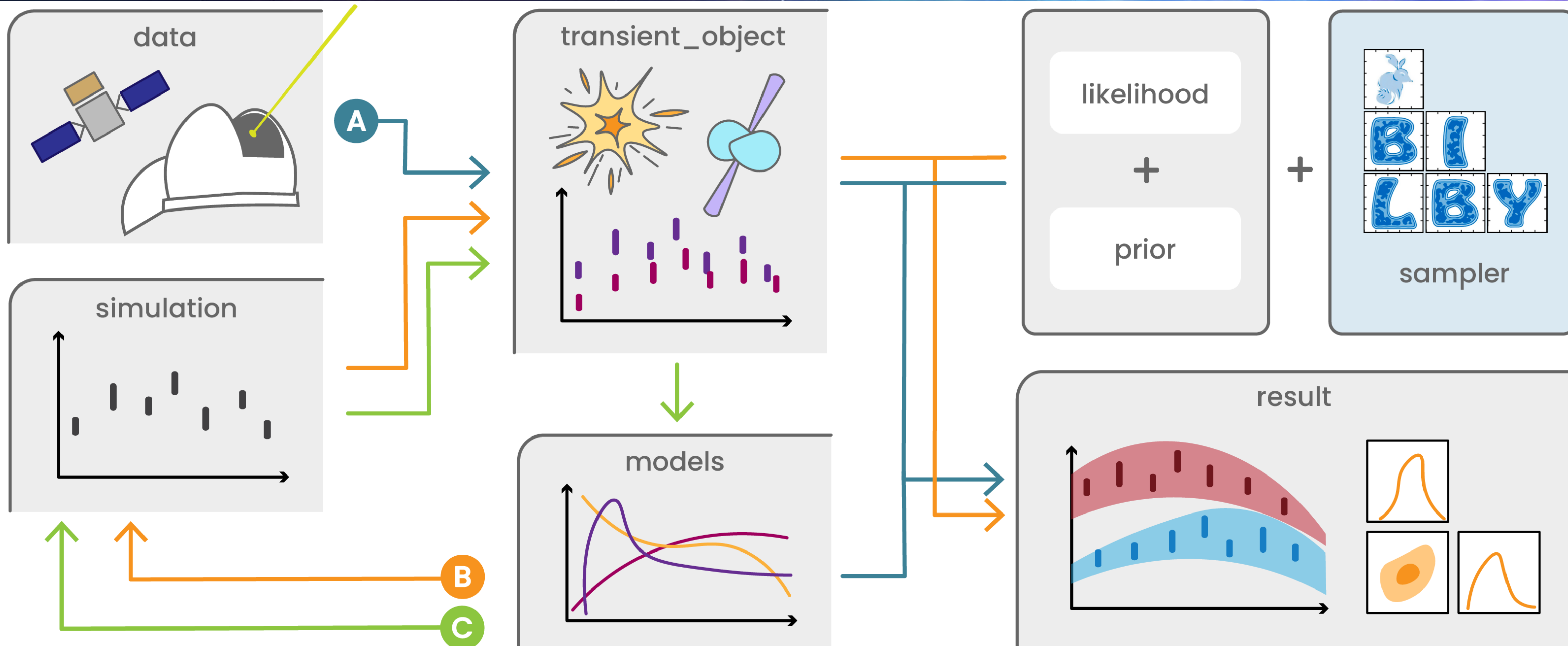


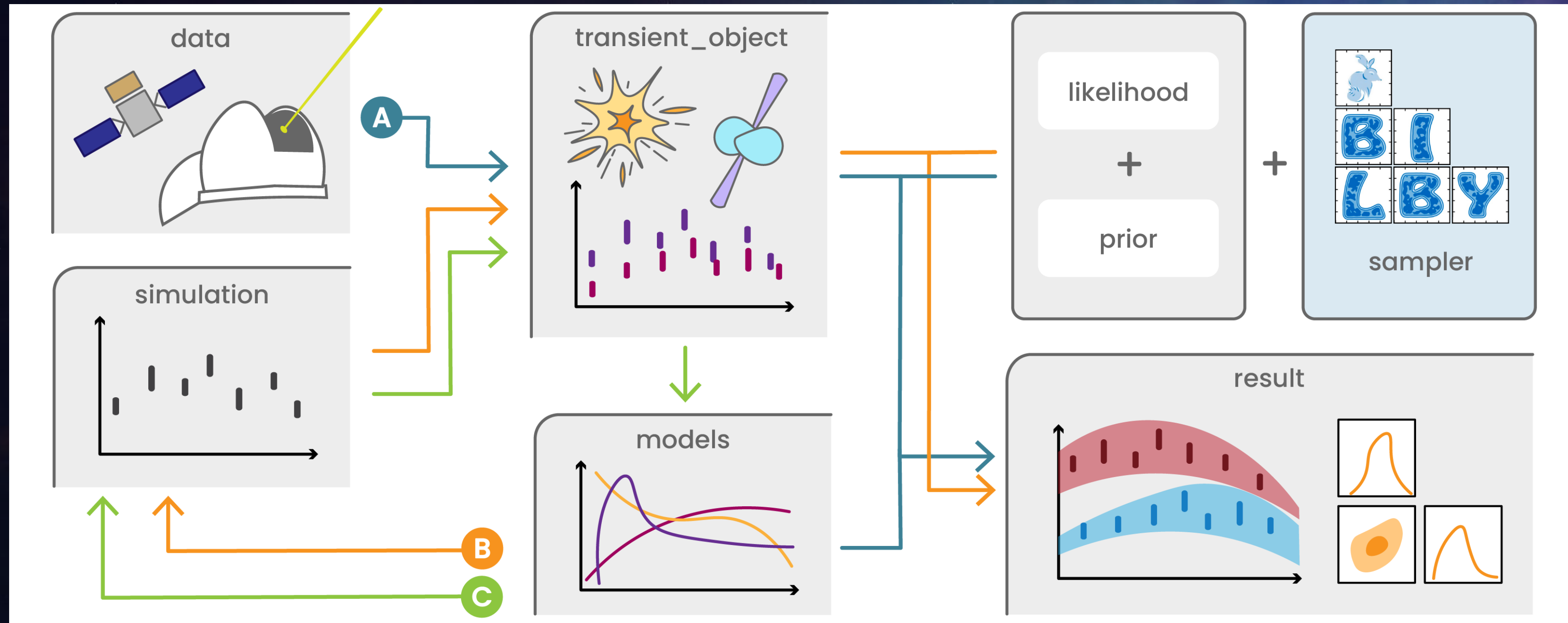
Villar+2017

- ▶ Need modular, open-source software.
 - ▶ Keep pace with rapidly accumulating observations and improving models.
- ▶ Adaptable to expert workflows, e.g., treatment of data, noise assumptions, and type of data.
 - ▶ But friendly for beginners.
- ▶ End to end.
 - ▶ Simulate, investigate/develop models, and do inference with one package.
- ▶ Capable of joint modelling with gravitational-wave data, or spectrum + photometry, or of a GRB afterglow + supernova/kilonova etc



Villar+2017





- ▶ A modular design. 3 primary workflows.
 - ▶ Get data/load private data -> create transient object -> specify a model -> set a prior + likelihood -> Specify a sampler implemented in Bilby -> Fit
 - ▶ Choose Model/provide your own -> Simulate observations -> Fit.
 - ▶ Model -> simulate single transient/population/survey -> fit, or check survey strategies.

```
import redback
import pandas as pd

# First, let's show how to get data from FINK.
name = 'ZTF22abdjqlm'
data = redback.get_data.get_fink_data(transient=name, transient_type='supernova')

# Now LASAIR
transient = 'ZTF20aamdsjv'
data = redback.get_data.get_lasair_data(transient=transient, transient_type='supernova')

# OAC
tde = "PS18kh"
data = redback.get_data.get_tidal_disruption_event_data_from_open_transient_catalog_data(tde)

# BATSE
name = '910505'
data = redback.get_data.get_prompt_data_from_batse(grb=name)

# Swift
GRB = '070809'
data = redback.get_data.get_bat_xrt_afterglow_data_from_swift(grb=GRB, data_mode="flux")
```

- ▶ APIs for Open Access Catalog, LASAIR, Fink, Swift, BATSE.
 - ▶ Can download different types of data.
 - ▶ Redback does some additional processing internally. Looks for metadata e.g., T0, redshift, RA/DEC.
- ▶ Data saved as a text file; returned in pandas data frame.

- ▶ Where the magic happens
 - ▶ Transient processing.
 - ▶ Plotting functionality.
 - ▶ Links all the modules together.
- ▶ 2 parent classes:
 - ▶ Generic transient
 - ▶ Optical Transient
- ▶ 6 child classes for specific transients
 - ▶ Kilonovae, SGRB/LGRB afterglow, Supernovae, TDEs, Prompt GRB.

```
name = 'ZTF22abdjqm'  
supernova = redback.supernova.Supernova.from_open_access_catalogue(name=name,  
                                                                    data_mode='flux')
```

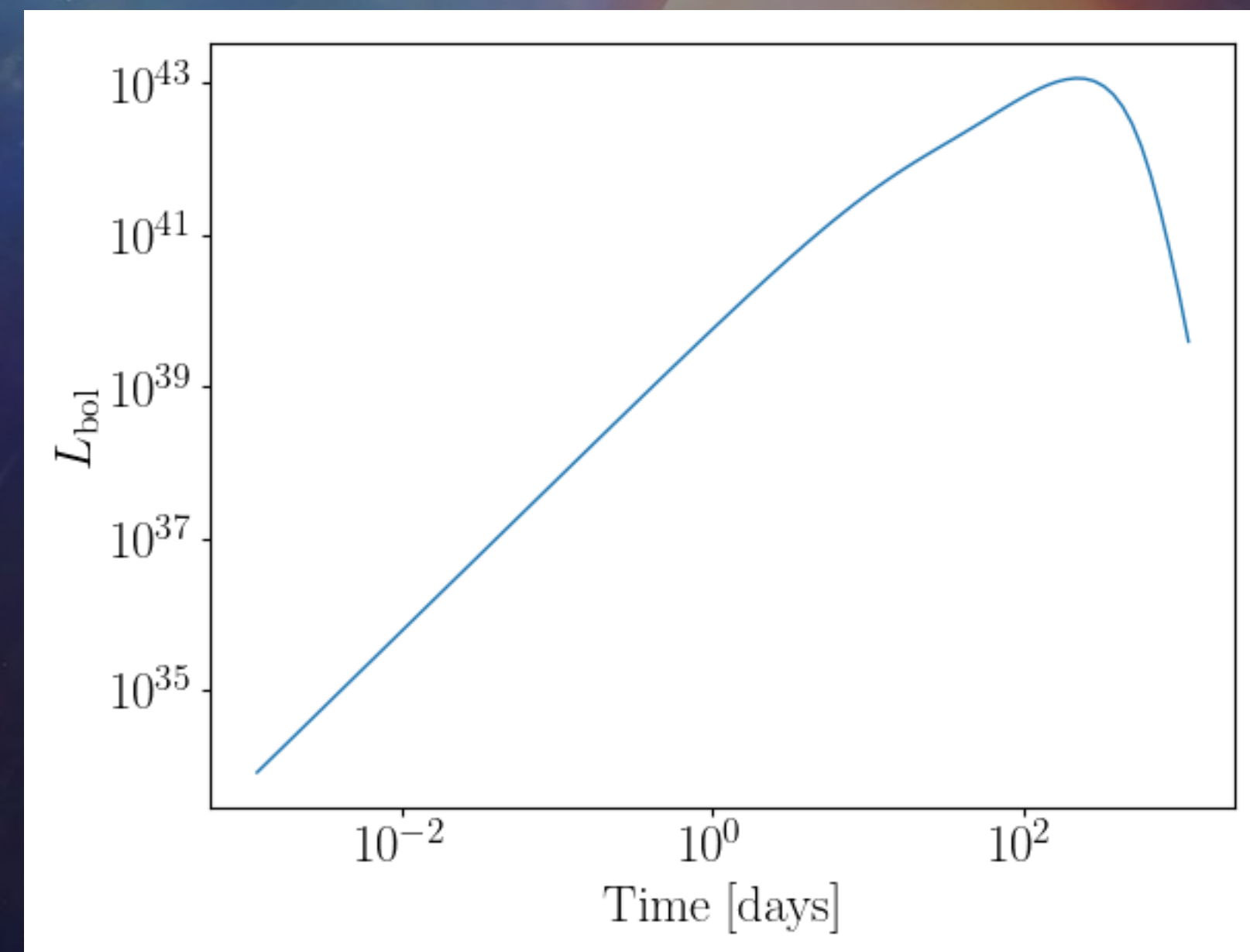
```
# We can also create a transient object from values in a csv file.  
data = pd.read_csv('example_data/grb_afterglow.csv')  
time_d = data['time'].values  
flux_density = data['flux'].values  
frequency = data['frequency'].values  
flux_density_err = data['flux_err'].values  
# set some other useful things as variables  
name = '170817A'  
  
afterglow = redback.transient.Afterglow(  
    name=name, data_mode='flux_density', time=time_d,  
    flux_density=flux_density, flux_density_err=flux_density_err, frequency=frequency)
```

- ▶ >150 models are currently implemented in Redback.
 - ▶ Phenomenological models → surrogates of numerical simulations.
 - ▶ Simple interfaces to many other model packages e.g.,
 - ▶ All models in SNCosmo
 - ▶ Afterglowpy
 - ▶ Mosfit ports
- ▶ Inbuilt models for
 - ▶ GRB afterglows/kilonova afterglows/kilonovae/supernovae/TDEs/FBOTs, magnetar-driven explosions, millisecond magnetars + more.
- ▶ Almost all models can output in multiple formats
 - ▶ Flux density, bandpass magnitude, flux, and bolometric luminosity.

```
from redback.constants import day_to_s
from redback.model_library import all_models_dict

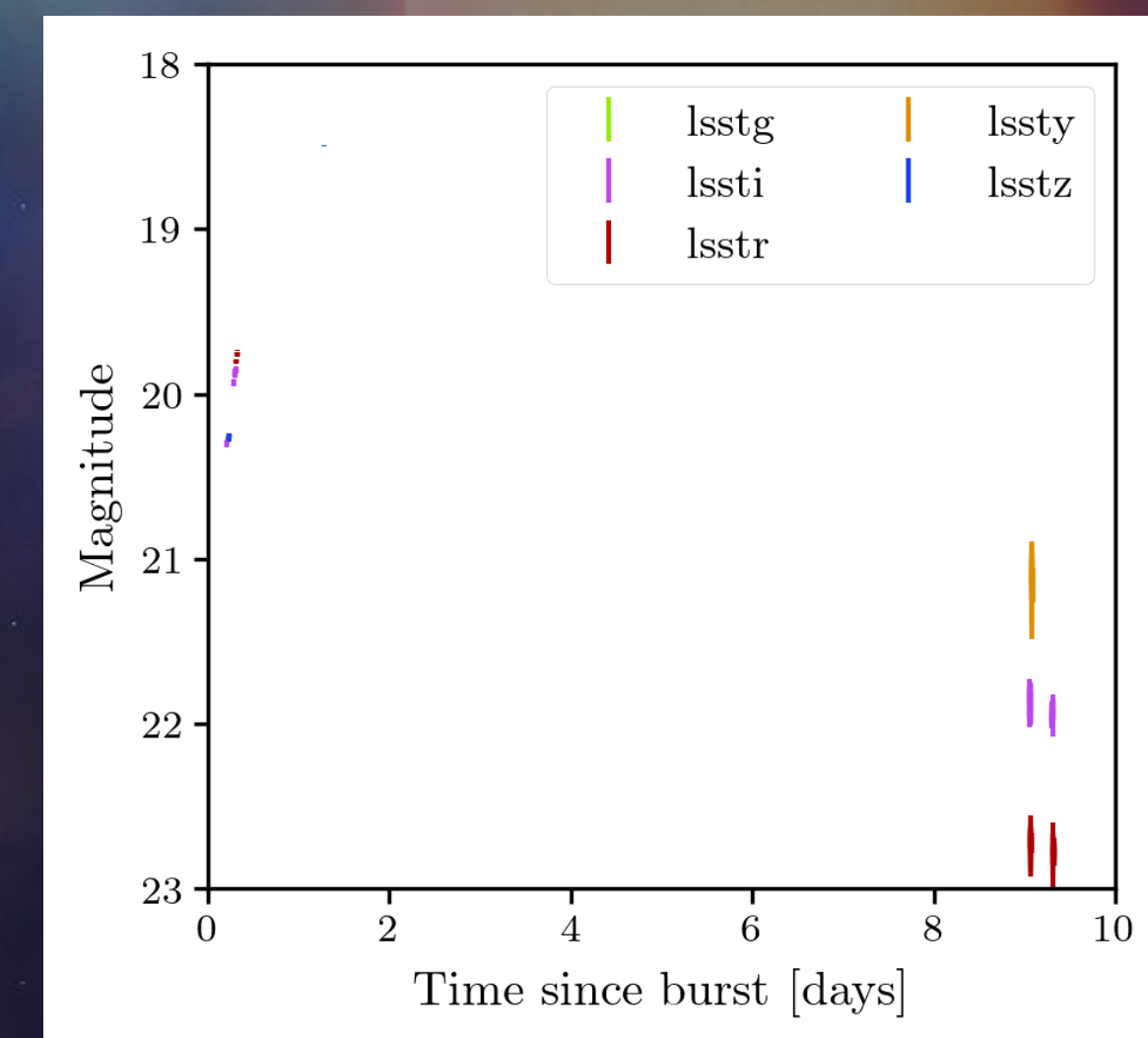
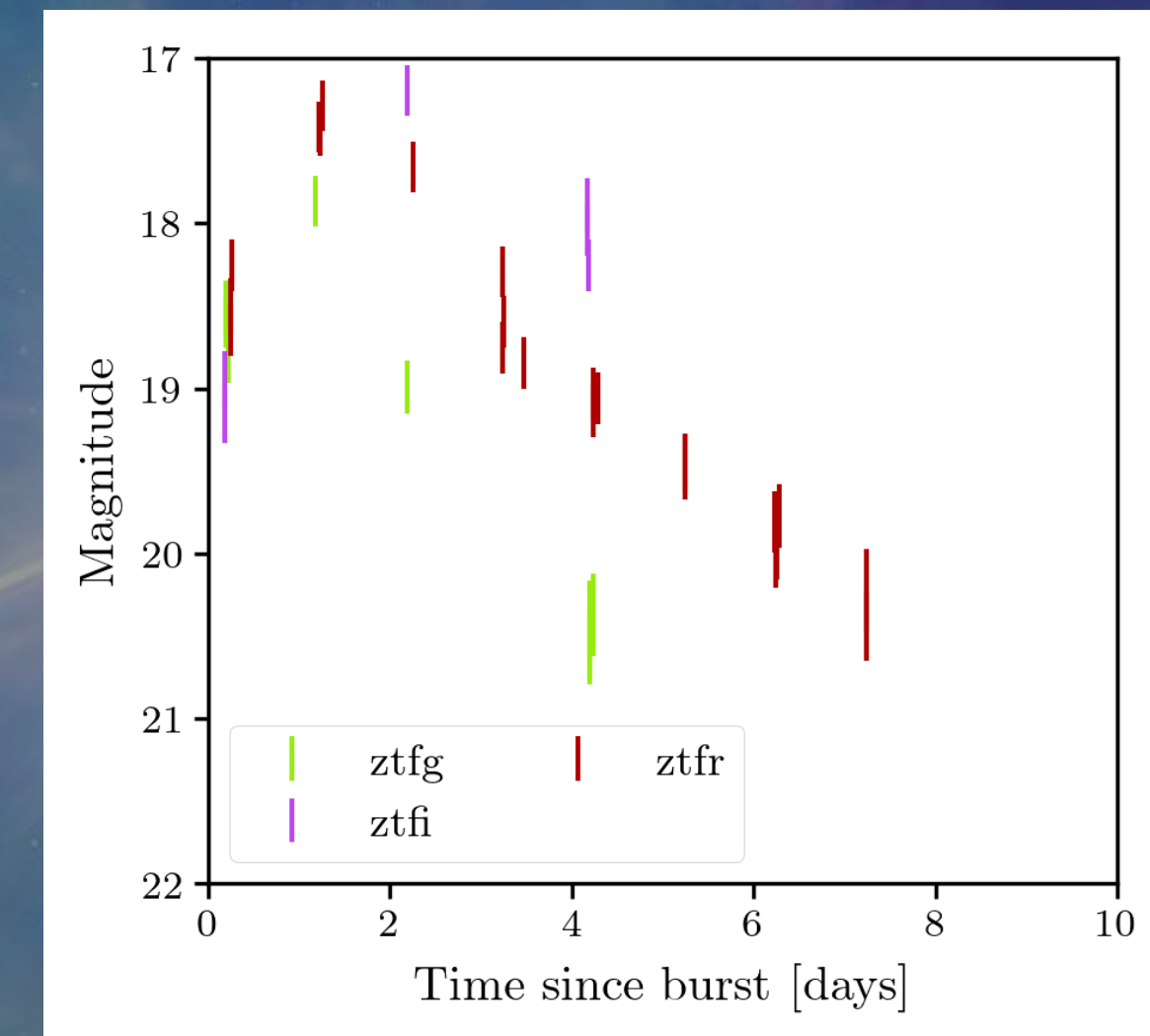
model = 'arnett_bolometric'

function = all_models_dict[model]
time = np.logspace(2, 8, 100)/day_to_s
bolometric_luminosity = function(time, f_nickel=0.2,
                                 mej=30, vej=10000, kappa=2, kappa_gamma=1e2)
plt.loglog(time, bolometric_luminosity)
plt.xlabel('Time [days]')
plt.ylabel(r'$L_{\rm bol}$')
```



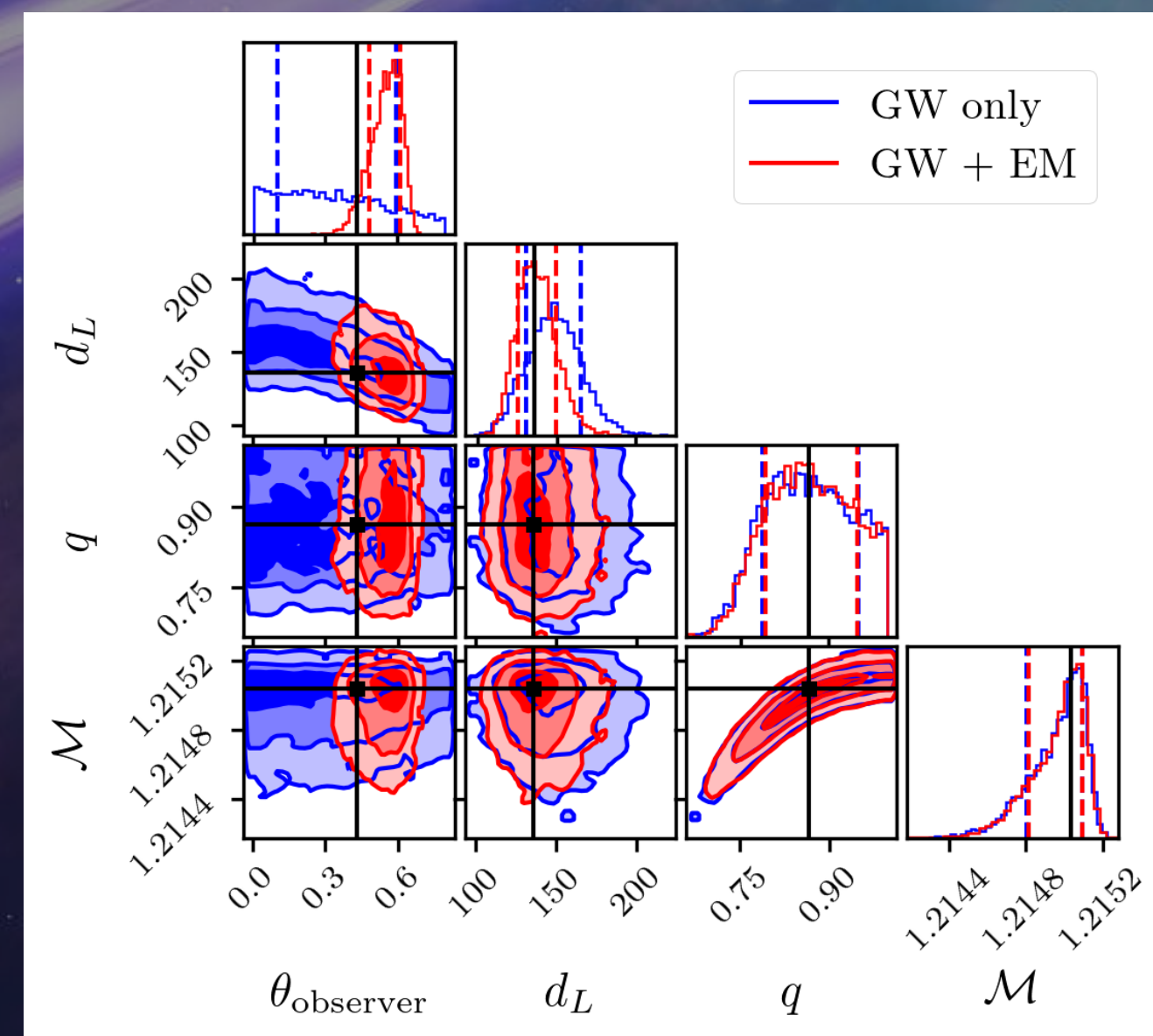
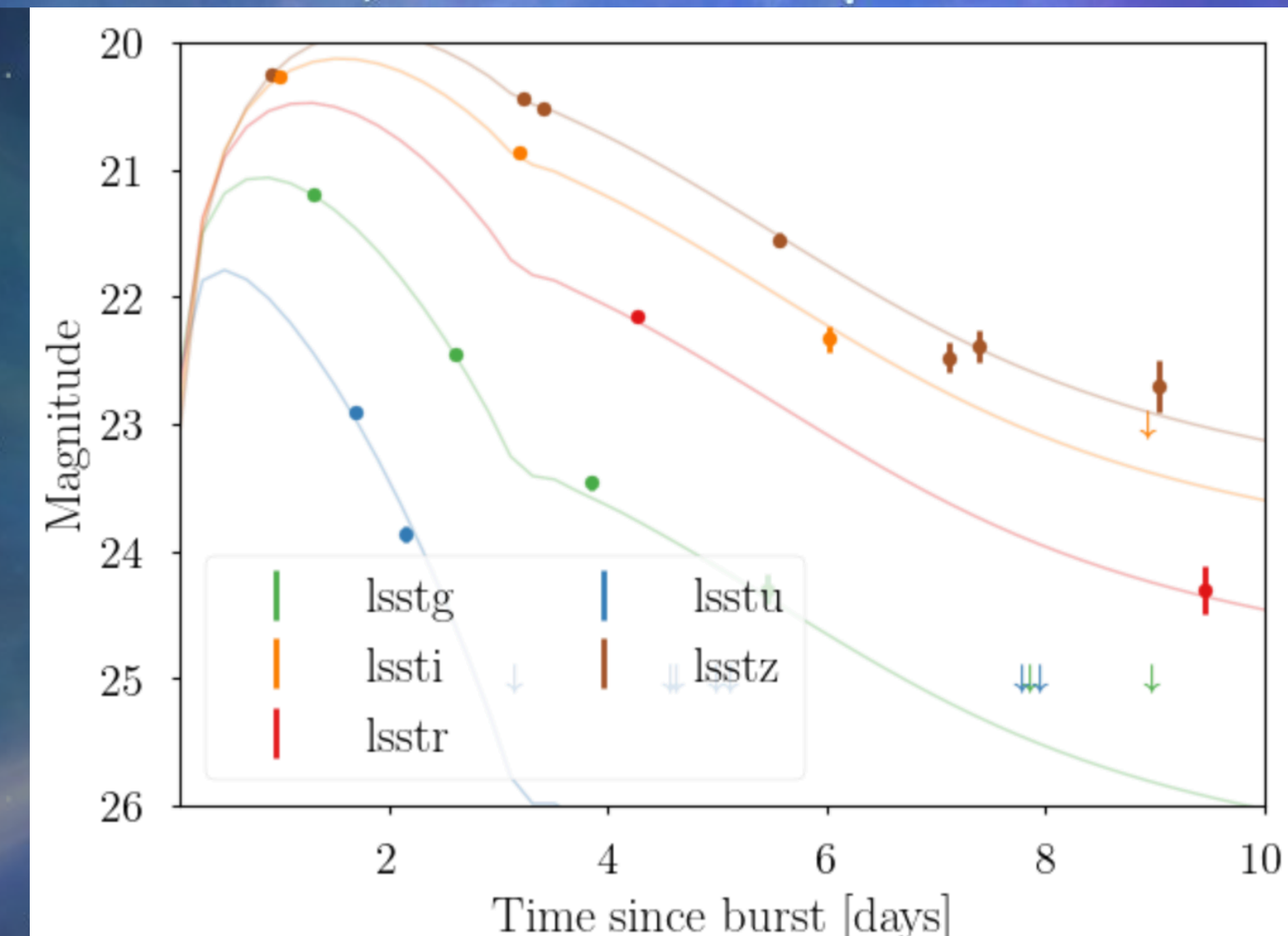
- ▶ Any redback model (or user provided model) can be used by the simulation module.
- ▶ Generic transient simulation, any type of transient/telescope.
- ▶ OpticalTransient (appropriate for a transient found in a survey/ToO observations).
- ▶ Full optical survey.
- ▶ ZTF and LSST Rubin have already been implemented (real/"official" survey pointing tables).
- ▶ Users can also create their own survey and simulate transients for that.

Simulated Kilonova in ZTF

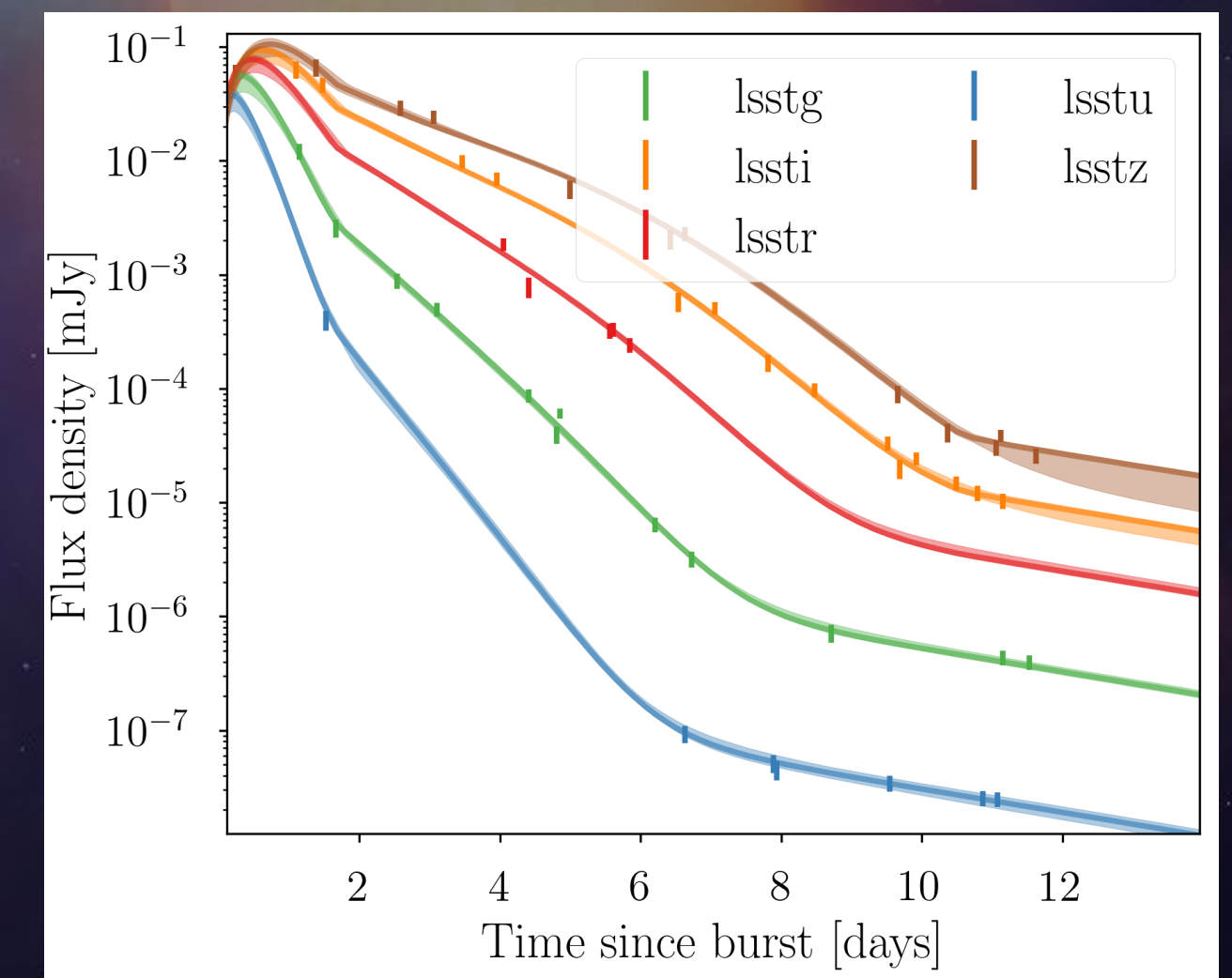


Simulated Kilonova in LSST with Rubin baseline v3.0 survey.

- ▶ Save the mock observations/
create a 'transient' object.
- ▶ Interface now identical to if
the data was private/catalog.
- ▶ Validation for entire
inference procedure.
- ▶ Studies of projected
constraints from future
observations/surveys.
- ▶ All these plots are one/
two lines of code!



Sarin+2023c



Sarin and Rosswog 2024

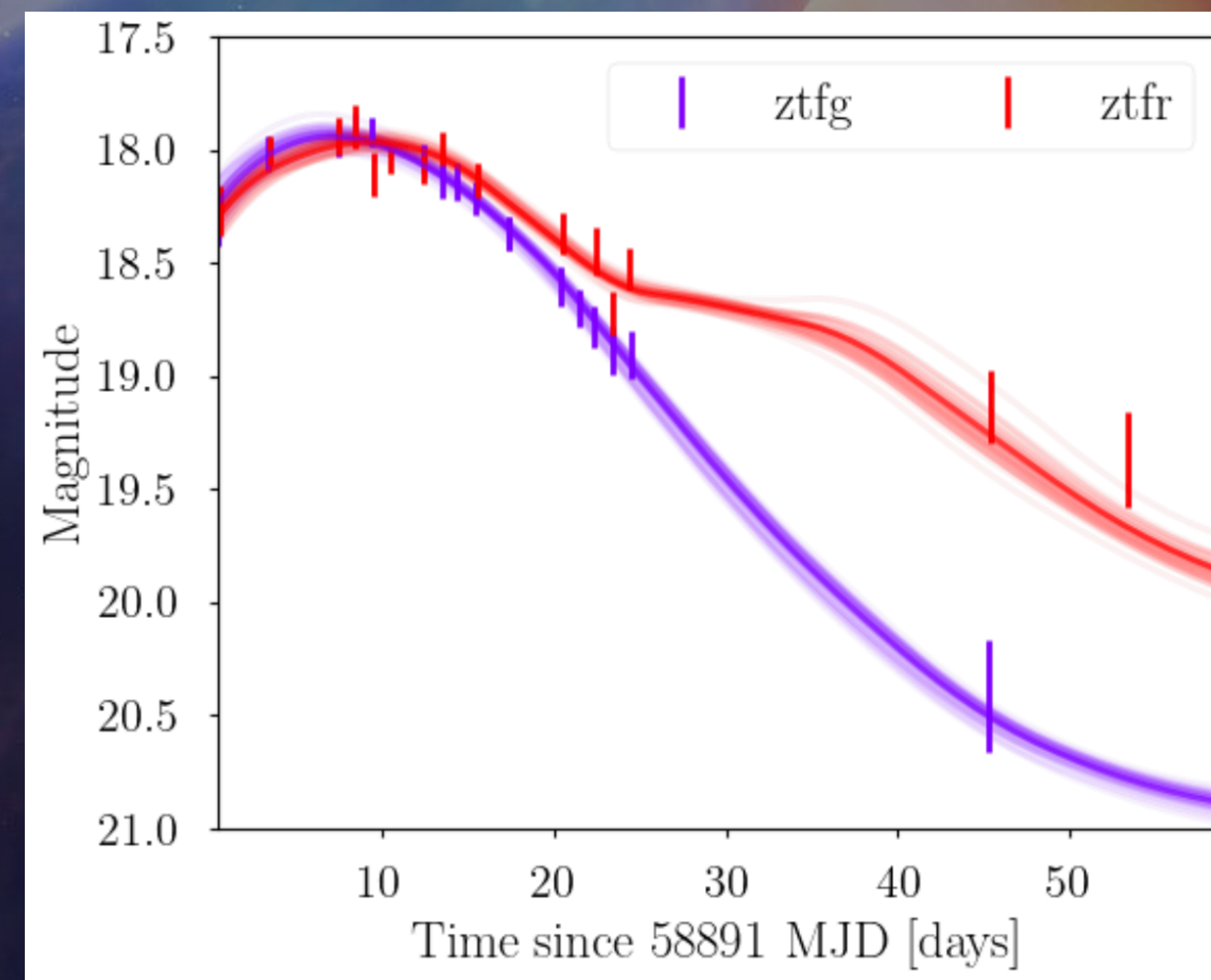
- ▶ Download data or use private data or simulate it.
- ▶ Create a transient object
- ▶ Pick a model (or use your own).
- ▶ Create a prior (or use the default)
- ▶ Choose your likelihood (or use the default). Or write your own.
- ▶ Pick a sampler. ~15 to choose from.

```
data = redback.get_data.get_lasair_data(transient='ZTF20aamdsjv', transient_type='supernova')

# Set up the redback transient object.
sn = redback.transient.Supernova.from_lasair_data(transient, use_phase_model=True,
                                                  data_mode='magnitude')

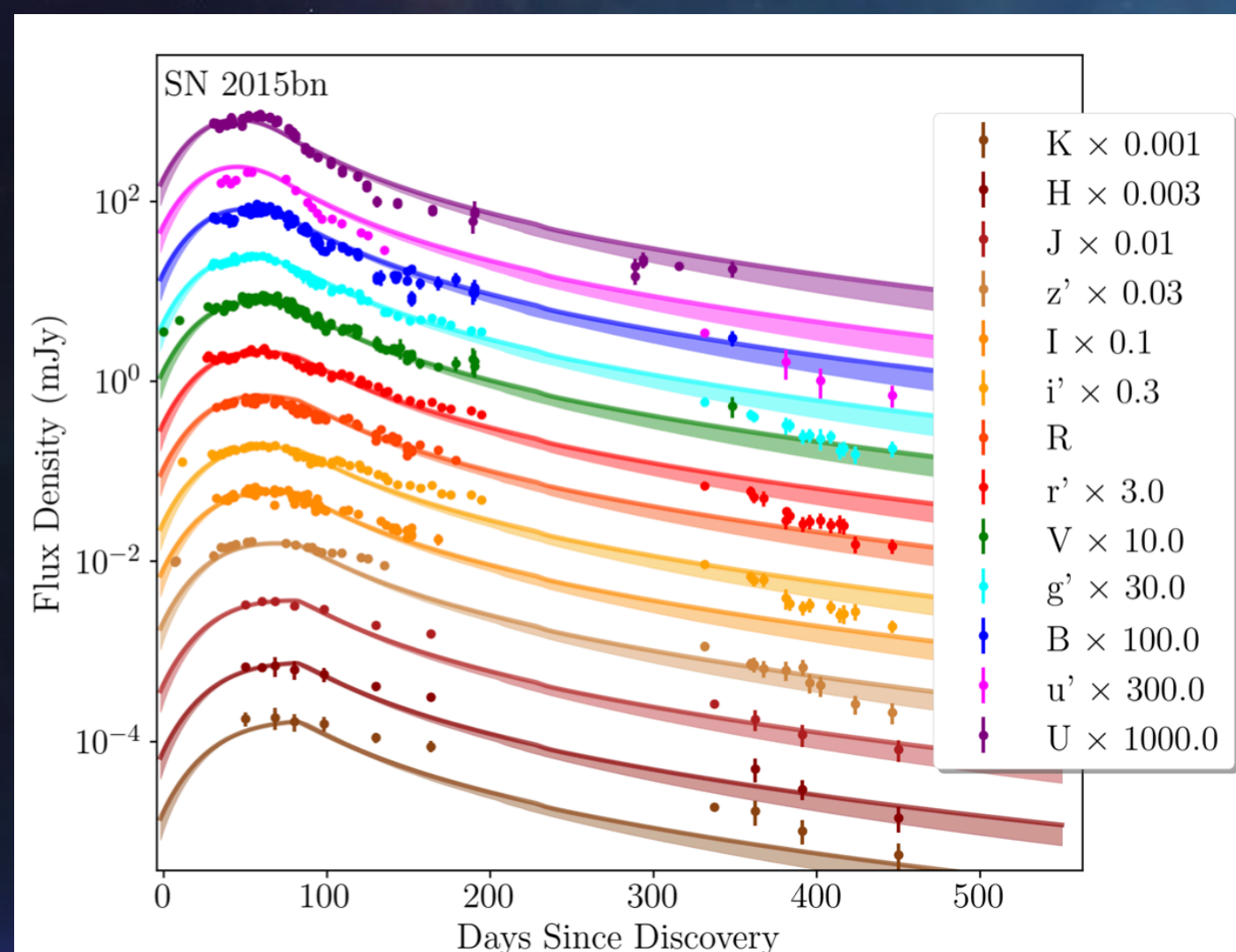
priors = redback.priors.get_priors(model='salt2')
priors['redshift'] = 0.061
kwargs = {'sncosmo_model': sncosmo_model, 'bands': sn.filtered_sncosmo_bands,
          'output_format': 'magnitude', 'model_kwarg_names': ['x0', 'x1', 'c']}

result = redback.fit_model(transient=sn, model='sncosmo_models', prior=priors,
                           model_kwargs=kwargs, sampler='ultranest', nlive=200, plot=False)
ax = result.plot_lightcurve(random_models=50, show=False)
```

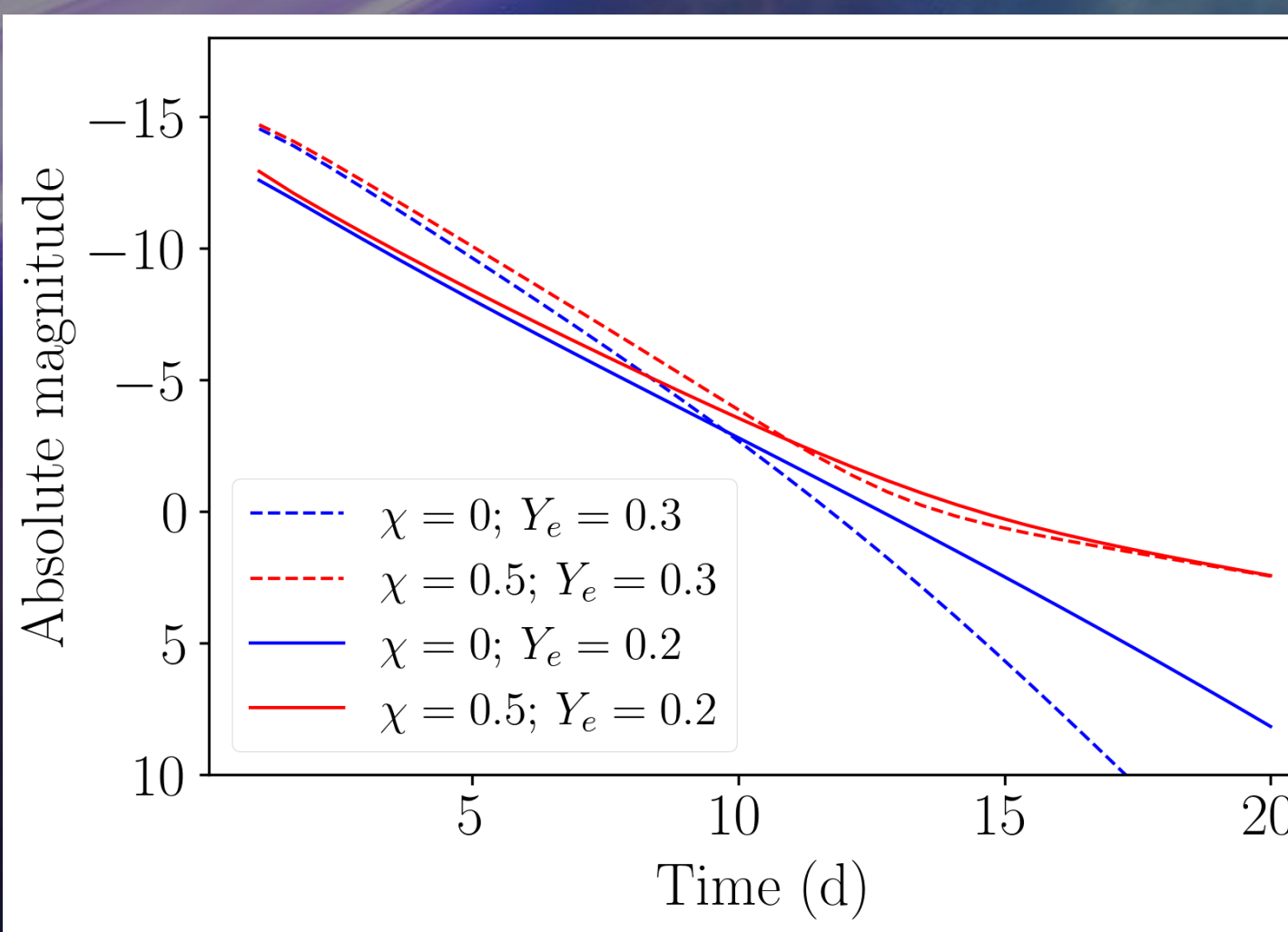
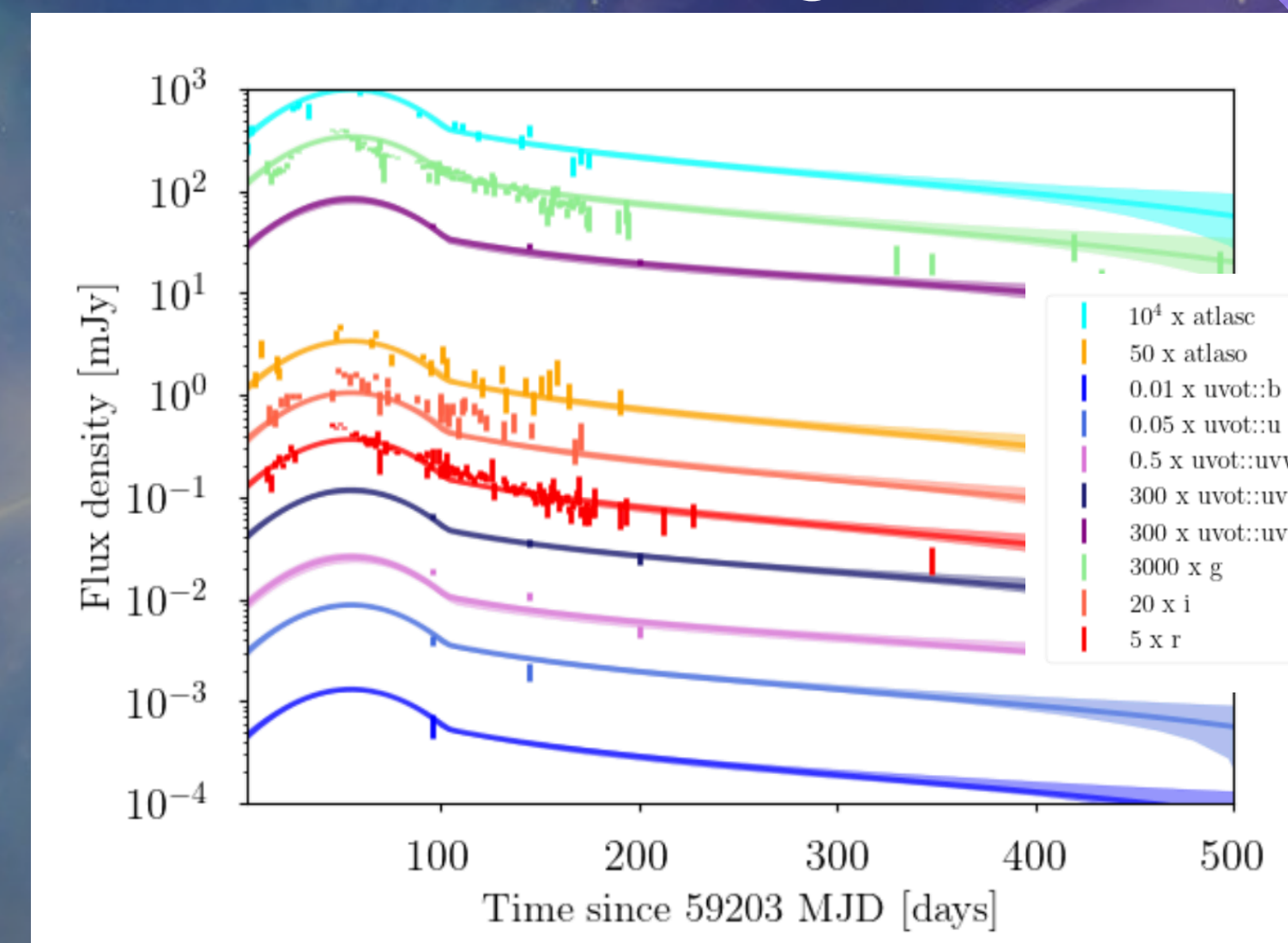


- ▶ Simple interface for a Joint likelihood.
- ▶ Jointly analyse EM data of different types.
 - ▶ Spectrum + photometry
- ▶ GW + EM data
- ▶ Kilonova/Supernova + GRB afterglow
- ▶ All the benefits of a Bayesian framework
 - ▶ Hierarchical modelling, Importance sampling, model selection etc

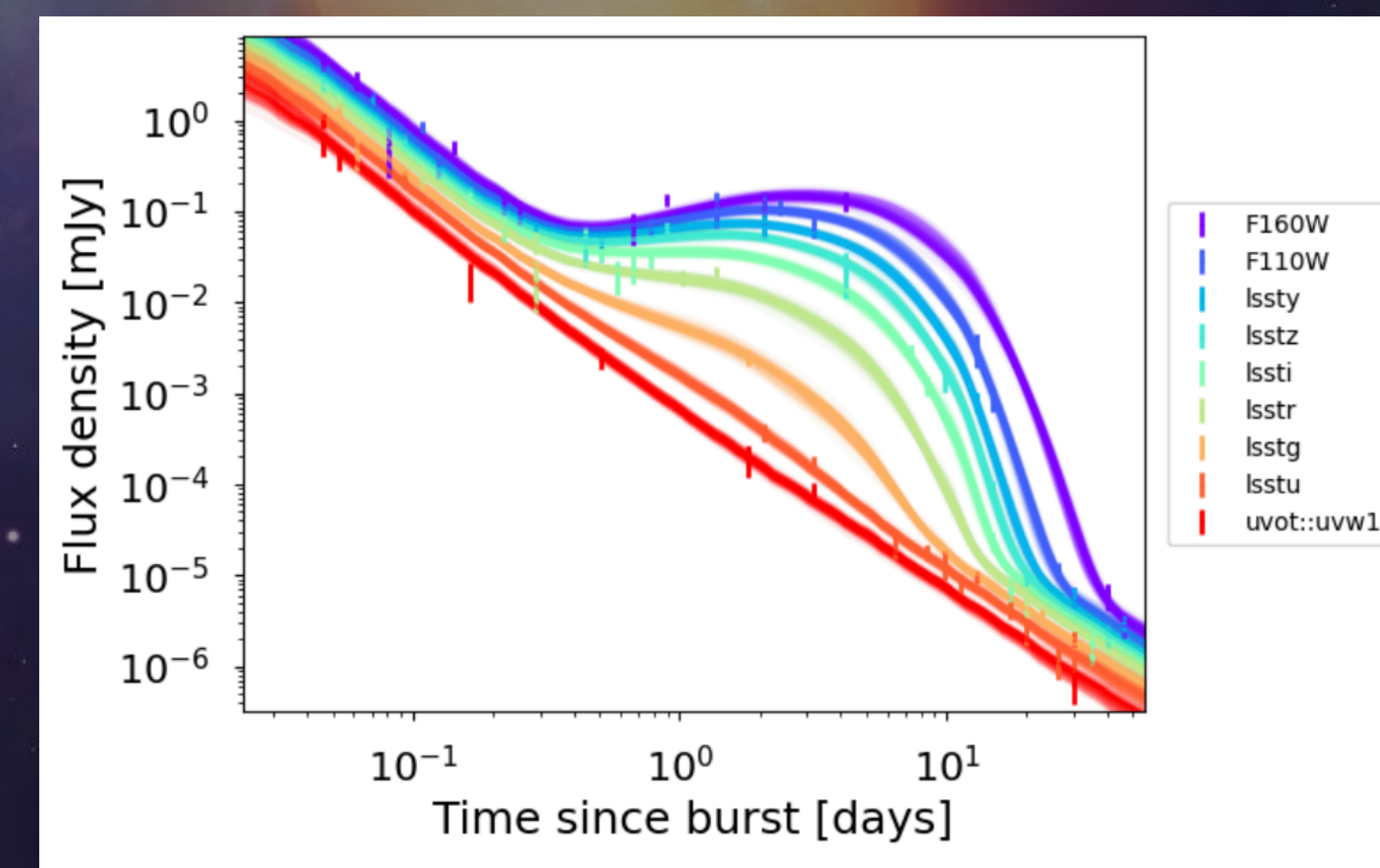
Omand and Sarin 2024



Sarin and Metzger 2024



Rosswog+ (incl Sarin) 2024



Wallace and Sarin 2024

docs **passing** Python application **passing** coverage **83%** pypi **v1.0.1** arXiv **2308.12806**

Redback

Introducing REDBACK, a bayesian inference software package for fitting electromagnetic transients

Online documentation

- [Installation](#)
- [Examples](#)
- [Documentation](#)

- ▶ Built on modern software development practices.
 - ▶ GIT with continuous integration.
 - ▶ 83% of the core software code (~11k lines) is tested on every commit in some way.
 - ▶ Pull request system for integrating changes.
- ▶ Large library of examples, extensive docs (both of API and otherwise).

CONTENTS:

- Installation
- Code motivation
- Basics of Bayesian inference and parameter estimation
- Likelihood
- Priors
- Get data
- Transients
- Models
 - Using redback models as functions
 - Modifying redback models
 - Extra kwargs and output formats in redback models
- Simulation
 - Making changes to models and plotting using dependency injections
- Plotting
- Analysis
- Fitting
- Results
- Joint likelihood
- Examples
- Acknowledgements

Modifying redback models

A lot of the physics in different redback models is set by default. However, several pieces of physics in various models can be changed by either passing your own function/class (see [dependency injections](#)), by switching the default argument with something else already implemented in redback, or changing a keyword argument.

The specific physics that can be changed:

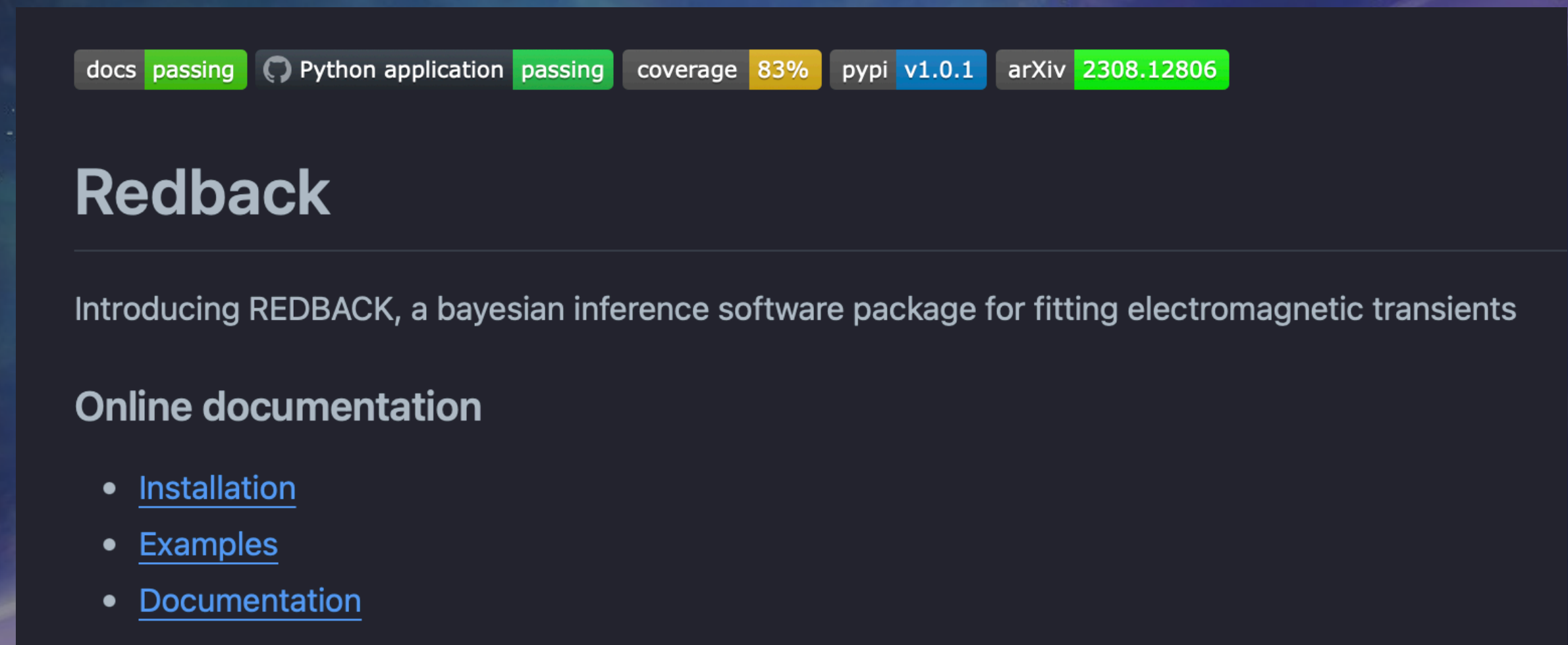
- Jet spreading on/off
- Whether to infer lorentz factor in afterglow models
- Whether to turn on/off pair cascades
- Whether to turn on/off neutron precursor emission
- Different ejecta relations: See relations already implemented [here](#).
- Different equations of states: See eos already implemented [here](#).
- Different interaction process: See processes already implemented [here](#).
- Different photosphere: See photospheres already implemented [here](#).
- Different SED: See SED's already implemented [here](#).
- Gamma-ray leakage
- Engines with different energy injection rates

We encourage users to add more of these physics switches, which is another easy way to contribute to redback.

Extra kwargs and output formats in redback models

All redback models have a set of extra keyword arguments that can be passed to the function of a specific model, users can see the extra keyword arguments by looking at the docstring of the

- ▶ Paper marking v1.0 release published in MNRAS.
- ▶ Development never stops. Plenty of improvements underway
 - ▶ New models, Fermi API. Better integration with Brokers/TNS.
- ▶ Contributions/developers/users welcome!
- ▶ Biweekly dev/user calls for tutorials and Q/A, etc.
 - ▶ Slack



docs passing Python application passing coverage 83% pypi v1.0.1 arXiv 2308.12806

Redback

Introducing REDBACK, a bayesian inference software package for fitting electromagnetic transients

Online documentation

- [Installation](#)
- [Examples](#)
- [Documentation](#)



- ▶ Redback - An open source, pip installable, python package for simulating and inference of any electromagnetic transient.
- ▶ Over 100 models and ~15 samplers implemented.
 - ▶ Swap models and samplers by simply changing a string.
 - ▶ GRB afterglows, TDEs, KNe, SNe, FBOTs, millisecond magnetars in X-ray plateaus, etc
- ▶ Interface to download and process data in a single line for Swift, BATSE, LASAIR, FINK, Open access catalog.
- ▶ Large library of examples, extensive docs (both of API and otherwise).
- ▶ Happy to chat more about specific science done so far with Redback on TDEs, supernovae, GRB afterglows, kilonovae.
 - ▶ Or the science you want to do!

