Joint Light-Curve Analysis (JLA)

The `jla` module provides access to supernova data compiled by the Joint Light-Curve Analysis project. It includes recalibrated light-curves of type Ia supernova (SN Ia) from the SDSS-II and SNLS collaborations.

Data Release	Class name	Data Type	Publication
Betoule et al. 2014	Betoule14	Photometric	Betoule et al. 2014

Betoule et al. 2014

class **sndata.jla.Betoule14** [source]

The `Betoule14` module provides access to light-curves used in a joint analysis of type Ia supernova (SN Ia) observations obtained by the SDSS-II and SNLS collaborations. The data set includes several low-redshift samples (z<0.1), all 3 seasons from the SDSS-II (0.05 < z < 0.4), and 3 years from SNLS (0.2 <z < 1) and totals 740 spectroscopically confirmed type Ia supernovae with high quality light curves. (Source: Betoule et al. 2014)

This data set includes observations taken in the pre 2015 MegaCam filter set used by the Canada-France-Hawaii Telescope (CFHT). These filters were measured at multiple positions by both the observing team and manufacturer. Transmission functions registered by this module represent the average transmission across the filter as reported by the manufacturer.

> **Deviations from the standard UI:**
>
> - None

> **Cuts on returned data:**
>
> - None

> **delete_module_data**() → None
>
> Delete any data for the current survey / data release

download_module_data(*force: bool = False*, *timeout: float = 15*)

Download data for the current survey / data release

Parameters:
- **force** – Re-Download locally available data
- **timeout** – Seconds before timeout for individual files/archives

get_available_ids() → List[str]

Return a list of target object IDs for the current survey

Returns: A list of object IDs as strings

get_available_tables() → List[Union[str, int]]

Get Ids for available vizier tables published by this data release

get_data_for_id(*obj_id: str*, *format_table: bool = True*) → astropy.table.table.Table

Returns data for a given object ID

See `get_available_ids()` for a list of available ID values.

Parameters:
- **obj_id** – The ID of the desired object
- **format_table** – Format data into the `sndata` standard format

Returns: An astropy table of data for the given ID

classmethod **get_zp_for_band**(*band: str*) → str

Get the zeropoint for a given band name

Parameters: **band** – The name of the bandpass

iter_data(*verbose: bool = False*, *format_table: bool = True*, *filter_func: bool = None*) → astropy.table.table.Table

Iterate through all available targets and yield data tables

An optional progress bar can be formatted by passing a dictionary of `tqdm` arguments. Outputs can be optionally filtered by passing a function `filter_func` that accepts a data table and returns a boolean.

Parameters:
- **verbose** – Optionally display progress bar while iterating
- **format_table** – Format data for `SNCosmo` (Default: True)
- **filter_func** – An optional function to filter outputs by

Yields: Astropy tables

`load_table`(*table_id: Union[int, str]*) → astropy.table.table.Table

Return a Vizier table published by this data release

Parameters: **table_id** – The published table number or table name

`register_filters`(*force: bool = False*)

Register filters for this survey / data release with SNCosmo

Parameters: **force** – Re-register a band if already registered