

SPLASH

v0.1 under development

Updated Date: **June 08, 2020**

Maintainer: Lionel Garia (lionel.garcia@student.uliege.be)

Prajwal Niraula (pniraula@mit.edu)

With: Artem Burdanov, Michael Gillon, Julien de Wit

Class Target:

Person working on: **Prajwal**

Located in: splash.py

LightCurve to be modeled after Cambridge Pipeline and Prose
(Flexibility of N components)

Some of the methods to be modeled after lightcurve class in prose)

Reading the fits file.

First Column: Time

Second Column: Flux (Not normalized)

Input Parameters:

Location: path to the lightcurve

TargetName: FileName which is expected to be named after

Optional

BatchRun: To see if this is a batch run for different targets

Methods:

Parses and stores:

ParameterName,ParamValue

AccessData using the following instances:

Time: Time Series of the data

Flux: Flux Series of the data

NightwiseData ---> Divide the data into list parameters values night by night

CreateOutputDirectory---> Creates output directory tree.

StellarVariation ---> Quick sinusoidal fitting if power higher than certain threshold

PreClean ---> Running running STD. Mask data with high scatter.

MaskFlares --->

If Processed (Flag) is True:

Products for Portal

Plotting:

Best transit component Data every night

Class GLS:

Person working on: **Lionel**

Base class

Initialization: requires

Location: path to the tuning parameters (tuning option for speed and completeness)

T0 constraint to can be passed

TargetName: FileName which is expected to be named after

Class LinearSearch(GLS):

SPLASH polynomial

Nuance GP

Should yield light curve for Ben to work with

Suggested to have multicore capability. How many data points to fit at once

Class PeriodicSearch:

Global likelihood used (as in FM2015) (Lionel)

Folding Likelihood (Prajwal)

Periodogram producing phase coverage and Periodogram

BLS in likelihood Space (Suggested by Julien)

Have a framework to more different algorithms.

Should yields TO and period with SDE/SNR (metric)

What metric to optimize (SNR/SDE) --- Different metric expected to understand the robustness

Yields: TO, Period, and metric value.

Class TransitFit:

Fast MCMC → Robust characterization of detection threshold and False probability

Robust MCMC → Production of graph that can be used in a paper.

Class TestCase:

Flip Test:

Change the sign of light curve for all of the targets. No significant transits are expected.

BenchmarkTest

“Small Star Tests”

TOI 736

TRAPPIST Data from TRAPPIST South

EPIC 249631677 data for marginal detection cases.

Injection recovery test → Should also lead to completion Test

1. What to name the module?
2. Where the final code will be posted?
3. Where will it run? ---> Need for human intervention in early phase of implementation.
4. API to get the detrended lightcurve. Help from Peter.
5. API to access TESS lightcurve.