

Documentation of MSAP3-31 for PLATO

Consistency checks

Earl Patrick Bellinger

April 3, 2023

Table 1: Author information

| Prepared by | Date |
|-------------------|---------------|
| Earl P. Bellinger | March 4, 2023 |
| Checked by | |
| Approved by | |
| Authorized by | |

Table 2: Version history

| Issue | Date | Nº change description | Page(s) | Paragraph(s) |
|-------|---------------|-----------------------|---------|--------------|
| 1.0 | March 4, 2023 | Initial release | All | All |

Contents

| | | |
|----------|--|----------|
| 1 | Introduction | 3 |
| 1.1 | Scope of the document | 3 |
| 1.2 | Nomenclature | 3 |
| 1.3 | Referenced documents | 3 |
| 1.4 | Abbreviations | 3 |
| 2 | General overview | 4 |
| 2.1 | Name of the algorithm and status | 4 |
| 2.2 | Synopsis | 4 |
| 2.3 | Model | 4 |
| 3 | Lists of inputs and outputs | 5 |
| 3.1 | Complete list of inputs | 5 |
| 3.2 | Complete list of outputs | 5 |
| 4 | Processing description | 6 |
| 4.1 | Type of delivery | 6 |
| 4.2 | Algorithm maturity | 6 |
| 4.3 | Algorithm source | 6 |
| 4.4 | Pseudo-code | 6 |
| 4.5 | Flow diagram | 6 |
| 5 | Test case(s) | 7 |
| 5.1 | Implementation test case(s) | 7 |
| 5.2 | Scientific test case(s) | 8 |
| | Bibliography | 8 |

1 Introduction

1.1 Scope of the document

This document aims to provide a description of the consistency checking algorithm for the selection and validation module of the MSAP5. It provides technical details (inputs, outputs, data types) as well as the functional description (implementation). The position of this algorithm within the data processing pipeline is described in [RD1].

1.2 Nomenclature

See 3 and 4.

Table 3: Nomenclature

| Term | Description |
|------|---|
| M | mass of the star in units of the solar mass M_{\odot} |
| R | radius of the star in units of the solar radius R_{\odot} |
| A | age of the star in units of Gyr |

Table 4: Standard data types

| Type | Size | Values |
|-------|-----------|--------|
| array | arbitrary | floats |

1.3 Referenced documents

The following documents are referenced:

RD1 PLATO-LESIA-PSPM-DD-0021, Work and data flows of the stellar L1/L2 processing pipeline

1.4 Abbreviations

HSD Tukey’s Honestly Significantly Different statistical test

2 General overview

2.1 Name of the algorithm and status

The algorithm is MSAP5-31, *Consistency checks*. The baseline algorithm has been implemented and the prototype is complete with only minor revisions expected.

2.2 Synopsis

The objective of MSAP5-31 is to determine whether different measurements of mass (M), radius (R), and age (A) from different modules of the PLATO pipeline are consistent with one another on a star-by-star basis.

We test for consistency using the Honestly Significantly Different (HSD) algorithm proposed by [Tukey \[1949\]](#). The documentation from `scipy` [[Virtanen et al., 2020](#)] summarizes the HSD algorithm as:

“a pairwise comparison of means for a set of samples... The null hypothesis is that the distributions underlying the samples all have the same mean. The test statistic, which is computed for every possible pairing of samples, is simply the difference between the sample means. For each pair, the p-value is the probability under the null hypothesis of observing such an extreme value of the statistic, considering that many pairwise comparisons are being performed... In case of unequal sample sizes, the test uses the Tukey-Kramer method [[Kramer, 1956](#)].”

2.3 Model

The algorithm runs on M, R, and A independently. For each of these, the algorithm accepts a variable number of inputs corresponding to the measurements of that quantity from different methods used in the PLATO pipeline. Each input is an array containing samples from the posterior distribution according to that method. The algorithm then computes the HSD on the inputs, and reports whether any of the measurements are inconsistent at a significance level of 0.05. The implementation of HSD used here is from `scipy 1.10.1`.

3 Lists of inputs and outputs

3.1 Complete list of inputs

The inputs are tabulated in Table 5.

Table 5: Input parameters

| Name | Source | Status | Data type | Dimension | Unit |
|-----------------------------|---------|----------|-----------|-----------|-------------|
| IDP_MASS_SEISMIC | IDP_124 | optional | array | arbitrary | M_{\odot} |
| IDP_MASS_GRANULATION | IDP_125 | optional | array | arbitrary | M_{\odot} |
| IDP_MASS_GRANULATION_CGBM | IDP_125 | optional | array | arbitrary | M_{\odot} |
| IDP_MASS_RHO_TRANSIT_CGBM | IDP_125 | optional | array | arbitrary | M_{\odot} |
| IDP_RADIUS_SEISMIC | IDP_124 | optional | array | arbitrary | R_{\odot} |
| IDP_RADIUS_GRANULATION_CGBM | IDP_125 | optional | array | arbitrary | R_{\odot} |
| IDP_RADIUS_RHO_TRANSIT_CGBM | IDP_125 | optional | array | arbitrary | R_{\odot} |
| IDP_AGE_SEISMIC | IDP_124 | optional | array | arbitrary | Gyr |
| IDP_AGE_GYRO | IDP_125 | optional | array | arbitrary | Gyr |
| IDP_AGE_ACTIVITY | IDP_125 | optional | array | arbitrary | Gyr |
| IDP_AGE_GRANULATION_CGBM | IDP_125 | optional | array | arbitrary | Gyr |
| IDP_AGE_RHO_TRANSIT_CGBM | IDP_125 | optional | array | arbitrary | Gyr |

3.2 Complete list of outputs

Table 6: Output parameters

| Name | Status | Data type | Dimension | Unit |
|--------------------|-----------|-----------|-----------|------------|
| M consistency flag | mandatory | boolean | bit | true/false |
| R consistency flag | mandatory | boolean | bit | true/false |
| A consistency flag | mandatory | boolean | bit | true/false |

4 Processing description

4.1 Type of delivery

Prototype

4.2 Algorithm maturity

Algorithm concept defined, but interfaces (inputs/outputs) unstable. Has been tested with randomly-generated pseudo inputs, but needs to be tested with actual inputs from all of the PLATO modules. The complete list of seismic inputs needs to be added.

4.3 Algorithm source

The implemented algorithm and test cases are shipped directly to WP12 office alongside this document as a compressed archive.

4.4 Pseudo-code

N/A

4.5 Flow diagram

N/A

5 Test case(s)

5.1 Implementation test case(s)

We test four cases using simulated data. The default inputs we adopt are 100 random samples from a truncated normal distribution for each measurement method. We use default mean values corresponding to those of the present Sun: $1 M_{\odot}$, $1 R_{\odot}$, 4.57 Gyr. We adopt default standard deviations of $0.04 M_{\odot}$, $0.01 R_{\odot}$, and 0.5 Gyr. We truncate at $[0, 3] M_{\odot}$, $[0, 3] R_{\odot}$, and $[0, 15]$ Gyr. These tests are run in `MSAP5-31-consistency.ipynb`.

Case 1

- All consistent measurements.
- Inputs: Defaults
- Outputs: [True, True, True]

Case 2

- One inconsistent mass measurement. Additionally, in this case, the seismic radius measurement is missing.
- Inputs: Defaults, except 1 is added to all the samples from the first mass method
- Outputs: [False, True, True]

Case 3

- Two inconsistent radius measurements. Additionally, in this case, the seismic and granulation mass measurements are missing.
- Inputs: Defaults, except 0.5 is added to all the samples from the first radius method, and 1 is added to all the samples from the second radius method
- Outputs: [True, False, True]

Case 4

- Three inconsistent age measurements.
- Inputs: Defaults, except 2, 4, 6 are added to the first, second, third age methods
- Outputs: [True, True, False]

Case 5

- Consistent but invalid measurements.
- Inputs: Defaults, except the radii are 10 solar masses larger
- Outputs: [True, True, True]

5.2 Scientific test case(s)

Simulated data would be highly valuable in testing the algorithm.

Bibliography

- John W. Tukey. Comparing individual means in the analysis of variance. *Biometrics*, 5(2):99–114, 1949. ISSN 0006341X, 15410420. URL <http://www.jstor.org/stable/3001913>.
- Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.
- Clyde Young Kramer. Extension of multiple range tests to group means with unequal numbers of replications. *Biometrics*, 12(3):307–310, 1956. ISSN 0006341X, 15410420. URL <http://www.jstor.org/stable/3001469>.