1 Introduction

We are discussing the Plavchan periodogram, a variation of the "phase dispersion minimization" (PDM) algorithm used for detecting periodic signals in time series data.

2 Why was defined the default number of outliers equal 10?

In the implementation of the Plavchan periodogram, the parameter *num_outliers* is set to 10. This parameter controls the number of worst-fitting data points considered during the computation of the periodogram power. The selection of this value is a trade-off between sensitivity and computational efficiency.

The process involves comparing the data with the dynamical prior, which is generated by box-car smoothing the phased time series. By squaring the difference between the data and the prior, the algorithm quantifies the error or residuals. The sum of squared residuals is then calculated for a subset of the data points, namely the worst-fitting points.

Using a smaller value for *num_outliers* may increase sensitivity in detecting low signal-to-noise periodicities. However, this could also lead to increased computational burden since more data points need to be considered. On the other hand, choosing a larger value for *num_outliers* reduces computational cost but might sacrifice sensitivity.

By setting *num_outliers* to 10, a moderate compromise is made, striking a balance between sensitivity and computational efficiency in the Plavchan periodogram.

3 Why is the phase box size in the Plavchan periodogram typically set to 0.05?

3.1 Explanation

The phase box size, denoted as phase_box_size, determines the width of the phase box used for averaging the time-series data when computing the dynamical prior. In the Plavchan periodogram, a common choice for phase_box_size is 0.05.

3.2 Rationale

Setting phase_box_size to 0.05 allows for sufficient averaging while preserving the essential features of the time series data. A value of 0.05 is often considered suitable for ground-based transit surveys with a few thousand data points, as it balances the trade-off between capturing variability and avoiding oversmoothing.

4 Autoperiodogram: Determining the Number of Bins

The Autoperiodogram algorithm is employed to automatically determine the number of bins for the periodogram. It generates the *trial_periods* variable, which contains a set of candidate periods for exploring periodic signals in the data.

The algorithm takes as input the minimum and maximum periods to test, as well as the desired number of bins (i.e., trial periods) within that range. It evenly divides the specified period range, resulting in the *trial_periods* array.

To implement the Autoperiodogram algorithm in Python using the *lightkurve* library, we use the following code:

```
import lightkurve as lk
import numpy as np

# Search and download the light curve data
lc = lk.search_lightcurvefile(’Pi Mensae c’, mission=’TESS’).download_all()
lc = lc[0].PDCSAP_FLUX.normalize()

# Extract time_data and flux_data from the light curve and convert to 1D NumPy arrays
time_data = lc.time.value.flatten()
flux_data = lc.flux.value.flatten()

# Autoperiodogram algorithm: Generate trial_periods array
min_period = 0.1  # Minimum trial period to test
max_period = 30.0  # Maximum trial period to test
n_periods = 1000  # Number of trial periods to generate
trial_periods = np.linspace(min_period, max_period, n_periods)

# Rest of the code for Plavchan periodogram calculation and plotting
```

The *trial_periods* array is essential for exploring different periods and assessing their significance in the data. By automatically determining the number of bins, the Autoperiodogram algorithm streamlines the process of identifying periodic signals, making the periodogram analysis more efficient and effective.

Does the selection of trial periods in the Plavchan periodogram algorithm involve guessing?

Answer

The trial periods in the Plavchan periodogram algorithm do not involve guessing. Instead, they are systematically chosen to cover a range of possible periodicities for the given time series data. In the algorithm, a set of trial periods is predefined, representing potential periodic signals that may exist in the data. The range of trial periods is often determined based on the nature of the observed phenomenon and the desired sensitivity in the search for periodic signals. These trial periods are then used to test for periodicity in the time series data, allowing the algorithm to identify the most likely period of any periodic signal present in the data. The systematic approach of selecting trial periods ensures that the algorithm comprehensively explores the periodicity landscape, making it less reliant on random guessing and more effective in detecting periodic patterns in the data.