PHYS305 Observing Project

Project Summary

This end-of-term project for PHYS305 is designed to familiarize students with the processes involved in astronomical observing. It will involve downloading archival NASA data from the TESS Mission (https://exoplanets.nasa.gov/tess/), conditioning those data, and then applying a numerical model to them. To complete the project, students will need to submit a short write-up of their analysis. Details on all these aspects appear below. The final project write-up is due by Monday, Apr 29 at 12p MT and should be submitted on Canvas.

Data Analysis Process

1. Download and install the Lighkurve python package on your computer - https://docs.lightkurve.org/. You can use the MP301 lab computers, too, if you can't or don't want to use your own computer.

2. Find a hot Jupiter the transit for which you will analyze. Choose from one of the following targets: Kepler-1, TIC 431701493 (aka WASP-10), TIC 292152376 (HAT-P-32), TIC 233948455 (TrES-5), TIC 116264089 (TrES-3), HAT-P-19, TrES-1, Qatar-1, GJ436 (Gliese 436), XO-1, WASP-48, or WASP-3. Look up the planets' parameters using the NASA Exoplanets Archive - https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=PS.

3. To conduct your analysis, use the Jupyter Notebook Observing Project Notebook.ipynb posted on Canvas (and available here - https://github.com/decaelus/PHYS305_Spring2024/blob/main/Observing_Project/Observing%20Project%20Notebook.ipynb). You'll need to change the name of the planet and make a few tweaks throughout the notebook (see notes therein).

4. Be sure to save the figures created by the notebook into your final document.

<u>Questions to Answer for the Final Report</u>
Provide detailed answers to the following questions.

1. The photometric data points you analyzed have uncertainties σ associated with them. How could you estimate the per-point uncertainty? Using your method, what is that uncertainty? How would you expect anomalously outlying data points to affect your estimate? What are some techniques you could use to mitigate the effects of outliers on your estimate?
    a. If we first make the assumption that the model is good, the per-point uncertainty can be retrofit to force the reduced-chi-squared, χ^2_ν, to be equivalent to its ideal value of one:

    $$\chi^2_\nu = \frac{\chi^2}{N-M} = 1$$

    where N is the number of data points, and M is the number of model parameters. Consequently, in the idealized case, χ^2 is equal to $N - M$.

    $$\chi^2 = \sum_{i=0}^{N-1} \left(\frac{y_i - y(x_i)}{\sigma_i}\right)^2 = N - M$$

    Where y_i is the measured signal, $y(x_i)$ is the model prediction for the signal at x_i, and σ_i is the uncertainty of the measurement. Rewriting the difference between the measurement and expectation with notation that indicates that it's the residual we can state that

    $$\sum_{i=0}^{N-1} \left(\frac{r_i}{\sigma_i}\right)^2 = \sum_{i=0}^{N-1} \frac{r_i^2}{\sigma_i^2} = N - M$$

    Let's introduce the following definitions: $q_i = r_i^2$ and $z_i = \frac{1}{\sigma_i^2}$; such that χ^2 can be rewritten in a physically familiar way.

    $$\sum_{i=0}^{N-1} z_i q_i = N - M$$

    Examining the left hand side and dividing it by the sum of q indexed over i, we get:

    $$\frac{\sum_{i=0}^{N-1} z_i q_i}{\sum_{i=0}^{N-1} q_i} = \frac{\sum_{i=0}^{N-1} z_i q_i}{Q} = z_{cm}$$

    Where z_{cm} is the weighted centroid. I said that this should be physically familiar because this is the same notation as calculating the center of mass or center of charge for a discrete distribution of objects resting along the z-axis.

    $$z_{cm} = \frac{N-M}{\sum_{i=0}^{N-1} q_i}$$

    But, in this context, this is equivalent to finding the "center" of the distribution of per-point uncertainties.

    $$\left(\sigma_{cm}\right)^{-2} = \frac{N-M}{\sum_{i=0}^{N-1} r_i^2} \quad \Rightarrow \quad \sigma_{cm} = \sqrt{\frac{\sum_{i=0}^{N-1} r_i^2}{N-M}}$$

Note that this is an equivalent argument to saying that there is some constant, characteristic uncertainty of the distribution such that the following is true:

$$\sum_{i=0}^{N-1} \frac{r_i^2}{\sigma_i^2} = \frac{1}{\sigma_{cm}^2} \sum_{i=0}^{N-1} r_i^2$$

In other words, we're asking the question of what uncertainty value could be assigned uniformly to all measurements so that the behavior of the sum is the same as it would have been had we known each individual measurement uncertainty; based on the previously stated assumptions, the answer to that question is σ_{cm} which is approximately $\pm\ 0.004$.

b. $\sigma_{cm} \approx \pm\ 0.004$

```
[64]
# Calculating the per-point uncertainties using the folded, flattenned (detrended) observations and the Carter-model
# with the assumption that the model is a good fit, and forcing the reduced-chi-squared to 1
r_sq = (Carter_model(folded_time, *transit_shape_params)-detrended_flux)**2 # array of the squared residuals
N = len(detrended_flux) # number of observations
M = 2 # two model parameters
r_sq,N,M

(array([5.21017457e-05, 2.45355680e-06, 3.85935560e-07, ...,
        2.92901486e-05, 4.73019156e-05, 6.04216257e-05]),
 17228,
 2)

[69]
unc_cm = np.sqrt(sum(r_sq)/(N-M))
unc_cm

0.003631407992146301
```

Figure 1. Code to calculate $\sigma_{cm} \approx \pm\ 0.00363$

c. In principle, this uncertainty estimate, like the measures of center of mass, will be sensitive to the introduction of large outliers. However, because the measurements in question are of flux values and they're assumed to follow Poisson processes, the most anomalous detections shouldn't influence the metric because evaluating for σ_{cm} in this method occurs after cutting $\pm\ 5\sigma$ outliers in the derivative test. Additionally, while σ_{cm} will still be most representative of the larger uncertainty measurements relative to their frequency, that isn't necessarily a problem as it simply means that it is a conservative estimate by nature.

d. If we wanted to further mitigate the influence of outliers we could instead look to the median absolute deviation (MAD) of residuals instead:

$$MAD = median\left(\left|y_i - median\left(y_i\right)\right|\right)$$

$$\sigma_{MAD} = 1.4826 \times MAD$$

Performing this calculation yields a per-point uncertainty σ_{MAD} of $\pm\ 0.004$. Since both methods return a similar estimate, we can have increased confidence that the reported uncertainty is reasonable.

```
[71]
# Calculating the median per-point uncertainty using
# median deviation
r = Carter_model(folded_time, *transit_shape_params)-detrended_flux
r_median = np.median(r)
r, r_median
```

```
(array([-0.00721815,  0.00156638, -0.00062124, ..., -0.00541204,
        -0.00687764, -0.00777313]),
 3.805804925257299e-05)
```

```
MAD = np.median(abs(r-r_median)) #median absolute deviation (MAD)
unc_med = 1.4826*MAD # MAD based per-point uncertainty
unc_med
```

```
0.003607422032911811
```

Figure 2. Code to calculate $\sigma_{MAD} \approx \pm\ 0.00361$

2. Imagine you switched planets to observe a star that was 2.5 magnitudes brighter than your first target. Assuming Poisson uncertainties, how would you expect σ to change and by how much?

Qualitatively: signal-to-noise ratio should improve (increase) as we have more photons, but σ will also increase as it is proportional to the square root of the photon count for Poisson processes.

$\sigma_{Poisson} = \sigma(N) = \sqrt{N}$

$N \propto F$; the photon count is proportional to the flux

$N_2/N_1 = F_2/F_1$

$\sigma_2/\sigma_1 = \sqrt{N_2/N_1} = \sqrt{F_2/F_1}$

$\Delta m = m_2 - m_1 = -2.5$; brighter = smaller magnitude

$m = -2.5 log_{10}(F/F_0)$

$m_2 - m_1 = -2.5 log_{10}(F_2/F_0) + 2.5 log_{10}(F_1/F_0)$

$m_2 - m_1 = 2.5 log(F_1/F_2)$

$F_1/F_2 = 10^{(m_2 - m_1)/2.5}$

$F_2/F_1 = 10^{-(m_2 - m_1)/2.5} = 10^1$

$\sigma_2/\sigma_1 = \sqrt{10} \approx 3$

Therefore, we would expect a star with a magnitude brighter by 2.5 would yield an uncertainty larger by a factor of 3.

3. Of course, the photometric uncertainty σ will impact your results, specifically the results you get for the estimate of each transit time, t_c. The uncertainty on t_c depends on the system parameters according to the following equation:

$$\sigma_{t_c} = \sqrt{\frac{\tau}{2\Gamma}}\frac{\sigma}{\delta}$$

where τ is related to the ingress or egress duration, Γ is the sampling rate for your data (probably once every 2 minutes), σ is the per-point photometric uncertainty, and δ is the transit depth (how big the planet is compared to the star).

How would your uncertainty on the transit time change if you doubled the photometric uncertainty? How would it change if you doubled the transit depth (made the planet bigger compared to the star)? You can, of course, use the equation to make these estimates, but also explain qualitatively *why* you would expect that behavior? In words, why does the transit timing uncertainty go up or down as you change the photometric uncertainty and the transit depth?

1) $\sigma'_{t_c} = \sqrt{\frac{\tau}{2\Gamma}}\frac{2\sigma}{\delta} = 2\sigma_{t_c}$

   Doubling the photometric uncertainty, σ, doubles the transit uncertainty, σ_{t_c}. Physically, this would look like increased scatter, making it more difficult to distinguish the transit.

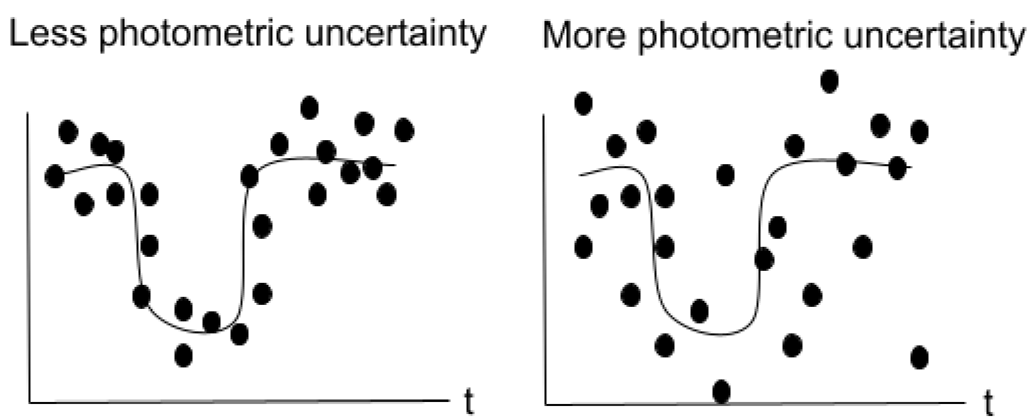
   

   Figure 3. Illustration comparing the same light curve with different amounts of photometric scatter.

2) $\sigma'_{t_c} = \sqrt{\frac{\tau}{2\Gamma}}\frac{\sigma}{2\delta} = 0.5\sigma_{t_c}$

   Doubling the transit depth, δ, halves the transit uncertainty. This is a reasonable expectation as the transit depth is the signal that we are observing, a decrease in measured flux from the host star. Therefore, a larger drop in flux caused by a larger planet would result in a signal that is easier to pick out from the background noise, improving our certainty on the transit timing.

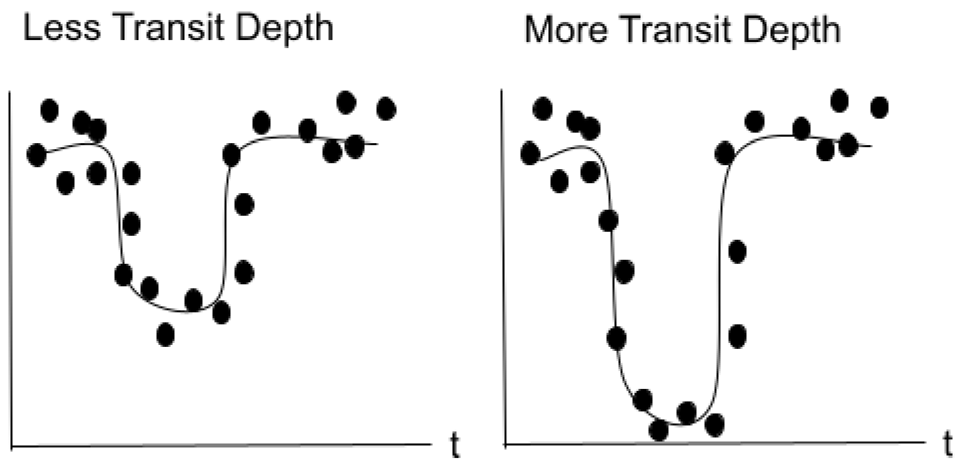


Figure 4. Illustration comparing transit light curves with approximately fixed scatter but different transit depths.

4. Now look at the period value reported on the Exoplanet Archive. We want to know whether your result is consistent (to within uncertainties) with their value. Look at your period value P_{yours} and their period value P_{theirs}, along with the corresponding uncertainties (σ_{yours} and σ_{theirs}, respectively). We want to consider the function $f = P_{\text{yours}} - P_{\text{theirs}}$, calculate the corresponding uncertainty for that function, and figure out whether the function might be equal to zero to within uncertainties. Consult Chapter 2 in Chromey to refresh your memory about how to propagate uncertainties.

   If your result does not agree with the Archive's, what are some possible reasons? Look at your transit model and compare it to the data. Does it look like a good fit to all the transits?

   Table 1. Shows the calculated period of Tres-3b with the observing project notebook alongside its corresponding uncertainty and literature references for comparison. The delta between "my" period (1.30618 ± 4E-5 days) and "their" (the reference) periods were calculated along with their propagated uncertainties. Finally, an average of the deltas and the uncertainty of that mean value was calculated (-0.6E-5 ± 1E-5 days). Literature observations that did not include uncertainties in the orbital period were excluded from consideration.

My Period [days]	My unc. [days]	Planetary Parameter Reference	Orbital Period [days]	uncertainty [days]	f [days]	sigma(f) [days]	sigma(f)^2 [days]^2	f_ave [days]	sigma(f_ave) [days]
1.30618	4.28939E-05	Patel & Espinoza 2022	1.3061858	0.0000003	-5.8E-06	4.28949E-05	1.83998E-09	-6.77149E-06	1.24107E-05
		ExoFOP-TESS TOI	1.306188017	3.30098E-07	-8.0169E-06	4.28952E-05	1.84E-09		
		Jiang et al. 2013	1.30618619	0.00000015	-6.19E-06	4.28942E-05	1.83991E-09		
		Kokori et al. 2022	1.30618639	0.00000003	-6.39E-06	4.28939E-05	1.83989E-09		
		Christiansen et al. 2011	1.30618608	0.00000038	-6.08E-06	4.28956E-05	1.84003E-09		
		Kokori et al. 2023	1.306186348	0.000000035	-6.348E-06	4.28939E-05	1.83989E-09		
		Southworth 2010	1.3061864	0.0000005	-6.4E-06	4.28968E-05	1.84014E-09		
		O'Donovan et al. 2007	1.30619	0.00001	-1E-05	4.40441E-05	1.93989E-09		
		Southworth 2011	1.306187	0.00000072	-7E-06	4.28999E-05	1.84041E-09		
		Bonomo et al. 2017	1.306186483	0.00000007	-6.483E-06	4.2894E-05	1.83989E-09		
		Mannaday et al. 2022	1.30618628	0.00000002	-6.28E-06	4.28939E-05	1.83989E-09		
		Ivshina & Winn 2022	1.30618627	0.00000021	-6.27E-06	4.28944E-05	1.83993E-09		

(4) $f = P_{my} - P_{theirs}$

$$\sigma_f^2 = \sum_i^N \left(\frac{\partial f}{\partial P_i}\right)^2 \sigma_i^2$$

$$= (1)^2(\sigma_{my})^2 + (-1)^2 \sigma_{theirs}$$

$$= \sigma_{my}^2 + \sigma_{theirs}^2$$

$$\sigma_f = \sqrt{\sigma_{my}^2 + \sigma_{theirs}^2}$$

$$f_{ave} = \frac{1}{N}\sum_i^N f_i$$

$$\sigma_{f_{ave}}^2 = \sum_i^N \left(\frac{\partial f_{av}}{\partial f_i}\right)^2 (\sigma_{f_i})^2$$

$$= \left(\frac{\partial}{\partial f_i}\left(\frac{f_i + f_j + f_k + \dots}{N}\right)\right)^2 \sigma_{f_i}^2 + \dots$$

$$= \left(\frac{1}{N}(1)\right)^2 \sigma_{f_i}^2 + \left(\frac{1}{N}(1)\right)^2 \sigma_{f_j}^2 + \dots$$

$$= \left(\frac{1}{N}\right)^2 \left[\sum_i^N \sigma_{f_i}^2\right]$$

$$\sigma_{f_{ave}} = \frac{1}{N}\left[\sum_i^N \sigma_{f_i}^2\right]^{\frac{1}{2}}$$

Figure 5. The work for deriving the propagated errors.

The calculated orbital period is in agreement with the reference values found in the literature (NASA Exoplanet Archive) to within the respective uncertainties as the difference between the "my" period and "their" periods for Tres-3b is about an order of magnitude less than the propagated uncertainty of their differences. However, the fact that "my" period was smaller in magnitude than all 12 references may indicate the presence of some systematic error that caused a slight underestimate of the period.

5. Using your period P and T_0 value (called "ephemeris_fit_params[1]" in your python notebook), you will estimate the next time that your planet could be observed in transit.

   First, you'll need to figure when your planet will next be visible. One way to check this is to use Stellarium (https://stellarium-web.org/). Most of your targets are in the web version, but a few (WASP-10, HAT-P-19, and Qatar-1) seem not to be. For those, you'll have to download and install Stellarium.

   In Stellarium, run time forward from today and check when your object will next be visible at night. Record that date and convert it to Julian date using this online calculator - https://www.aavso.org/jd-calculator. Don't worry about getting the exact instant the planet is visible at night; just get close.

   Next, you'll need to calculate the times in the future when your object will transit. You can calculate the transit time t_c for the Eth orbit using this equation: $t_c = T_0 + P\,E$. The first thing you'll need to do is to convert your T_0 value from your fit into Julian date. The fit value you get is in Julian date - 2457000, so start by adding 2457000 to your T_0. Then determine the number of orbits you'll have to wait until t_c is greater than the date you estimated from Stellarium. That should give you the minimum orbit number E for when your object is both visible and transiting. Record the next date when your object could be observed transiting and include all your arithmetic (neatly written) as part of your answer to this question.

Based on Stellarium (web version), TrES-3 is observable tonight, April 21st, 2024. Again according to Stellarium, the boundary between twilight and moonlight is around 22:05 in local time, which corresponds to a Universal Time (UT) of 05:05, April 22nd, 2024, as mountain standard time (MST) is seven hours behind. Similarly, the boundary between moonlight and dawn is around 05:10 MST which corresponds to 12:10 UT. So the nearest observing window approximately spans 05:05-12:10 UT, April 22nd, 2024. This can be converted to a Julian date (JD) window of 2460422.71181 to 2460423.00694.

```python
T0_fit = ephemeris_fit_params[0] + 2457000 # fit in JD
P_fit = ephemeris_fit_params[1]
tc_min = 2460422.71181 # in JD, calculated from Stelarium https://stellarium-web.org/
T0_fit, P_fit, tc_min
```

```
(2457000.372422372, 1.306184761192191, 2460422.71181)
```

```python
E = (tc_min-T0_fit)/P_fit # calculates the number of orbits
import math
E_min = math.ceil(E) # minimum number Eth orbits for visible and transiting
E, E_min
```

```
(2620.103594306306, 2621)
```

```python
tc_estimate = E_min*P_fit+T0_fit
print(f"The Julian Date Estimate estimate for the transit time, tc, is {tc_estimate}")
```

```
The Julian Date Estimate estimate for the transit time, tc, is 2460423.882681457
```

Figure 6. The code used to calculate the tc estimate; 2460423.882 in JD.

(5) From Stellarium, TrES-3 is next observable tonight; 05:05 UT

using provided JD calculator

$\rightarrow$ 05:05 UT = 2460422.71181 JD

given $t_c = T_0 + PE$

$$t_c - T_0 = PE$$

$$E = \frac{t_c - T_0}{P}$$

$T_0 \approx 245700.37242$ JD from fit

$P \approx 1.30618$ days/orbit from fit

$$E \approx \frac{[2460422.71181 \text{ JD} - 245700.37242 \text{ JD}]}{1.30618 \text{ days/orbit}}$$

$$\approx 2620.1 \text{ orbits}$$

$$E_{min} = \text{ceiling}(E) = 2621 \text{ orbits}$$

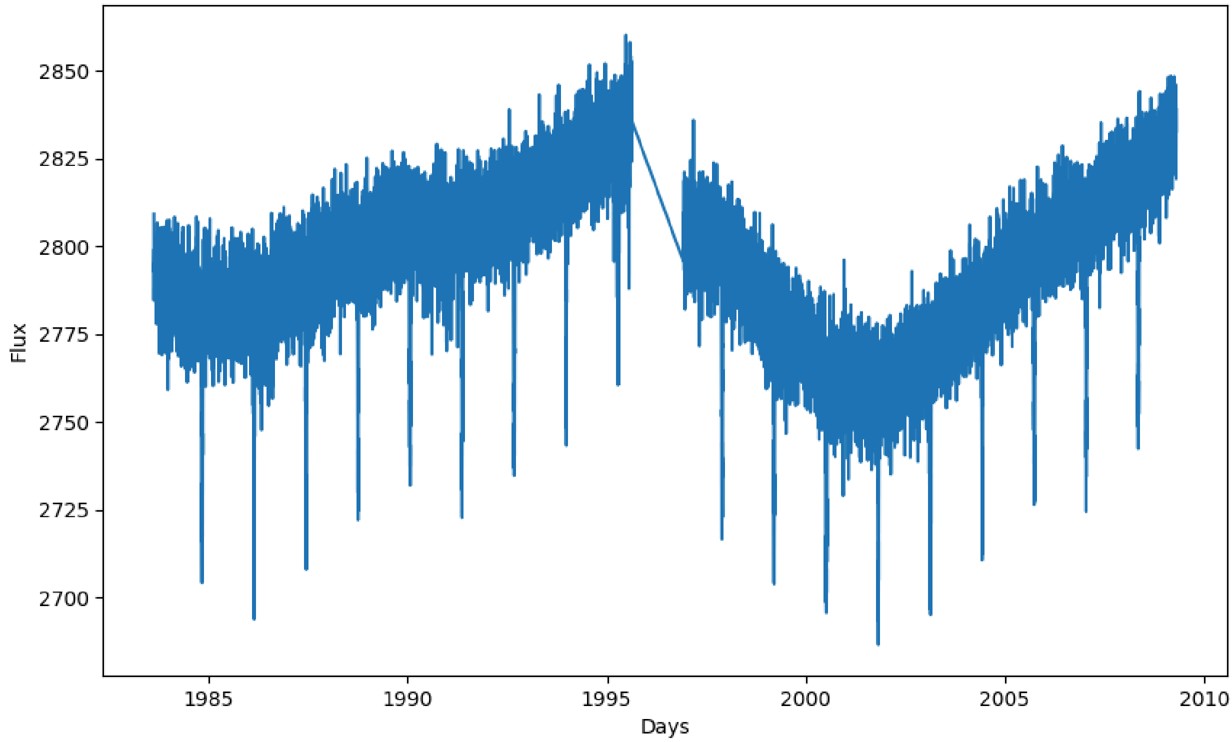
t_c estimate $= T_0 + PE_{min}$

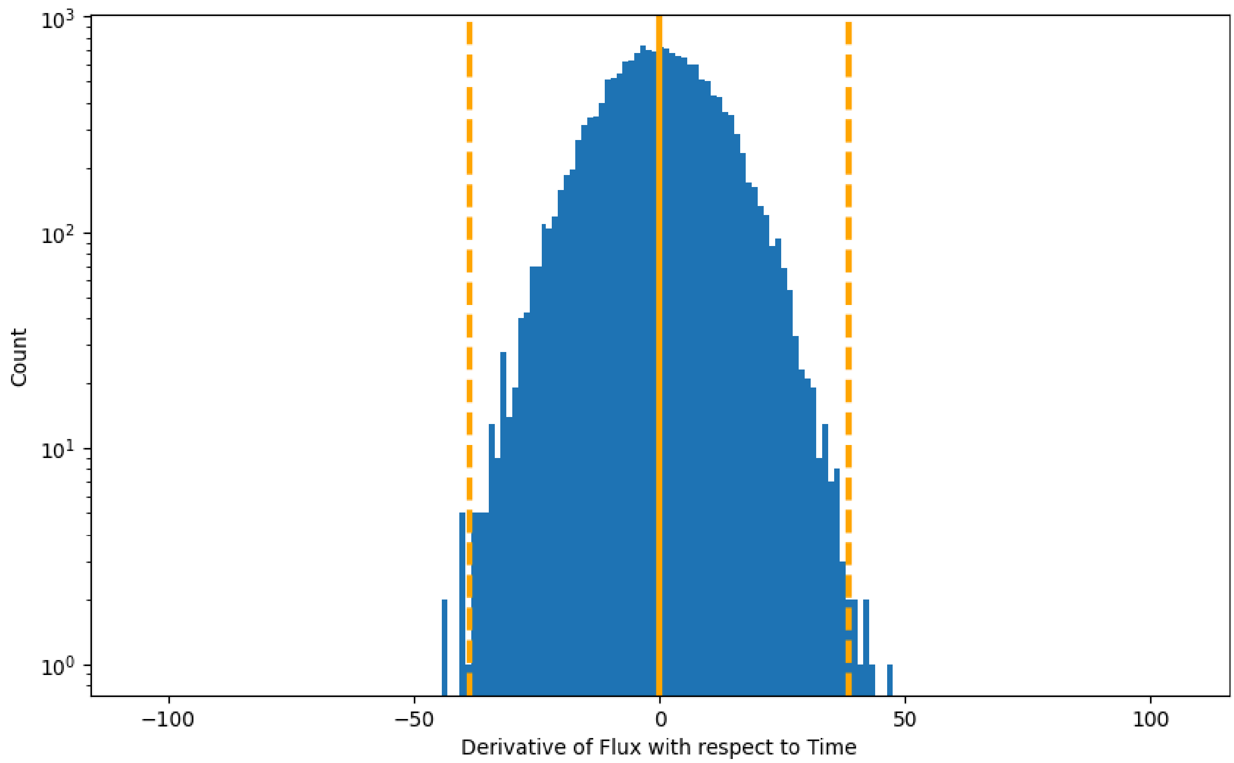
$$\approx (245700.37242 \text{ JD}) + \left(1.30618 \frac{\text{days}}{\text{orbit}}\right)(2621 \text{ orbs})$$

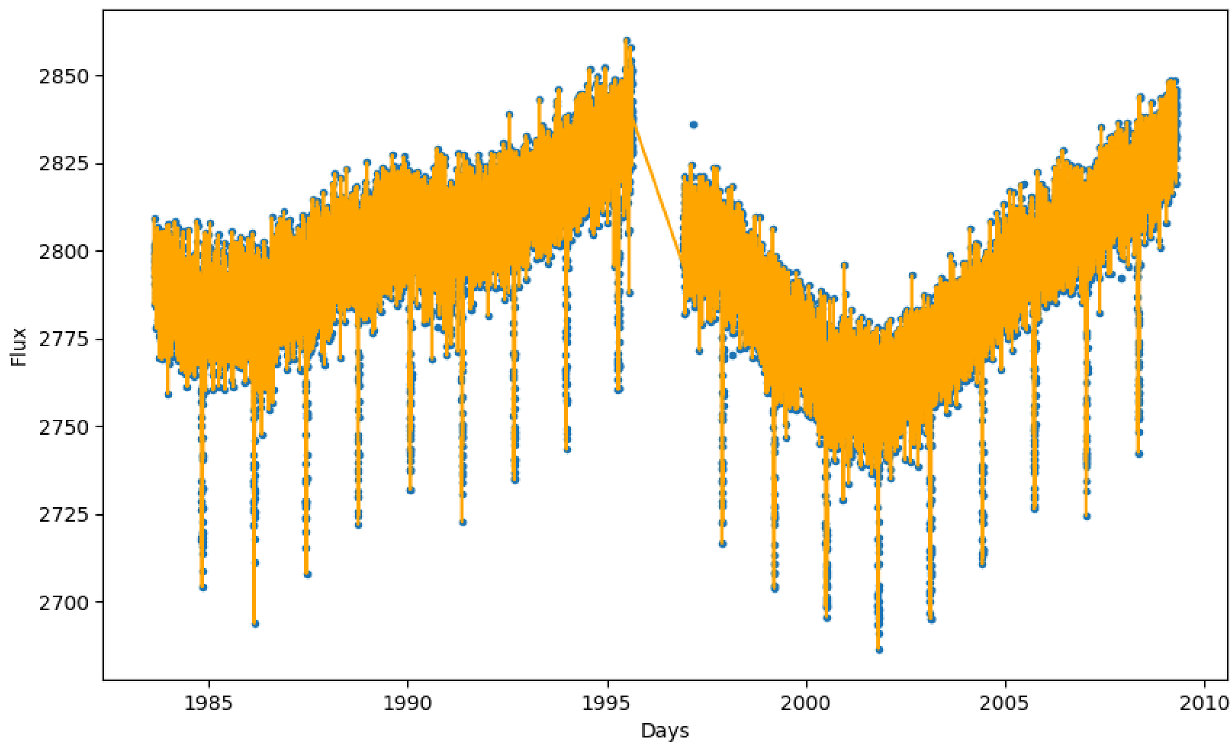
$$\approx 2460423.882 \text{ JD}$$

Figure 7. Hand-written solution.

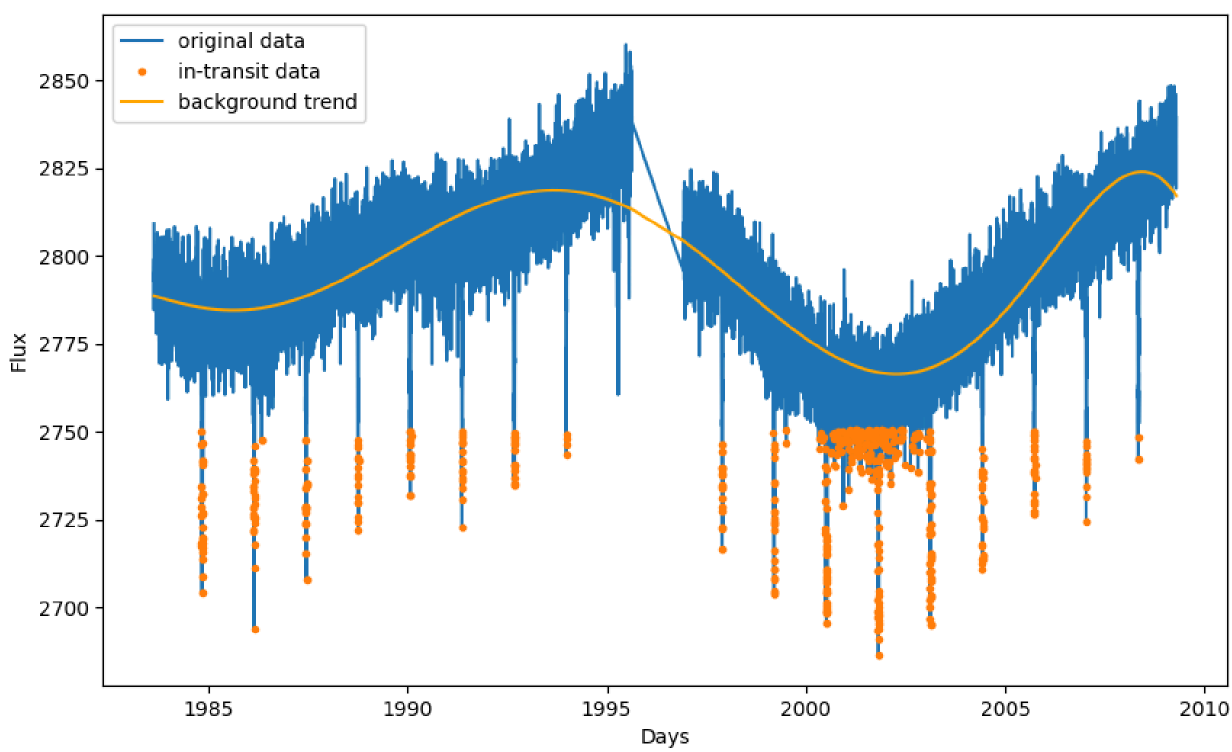
Figures from python notebook:



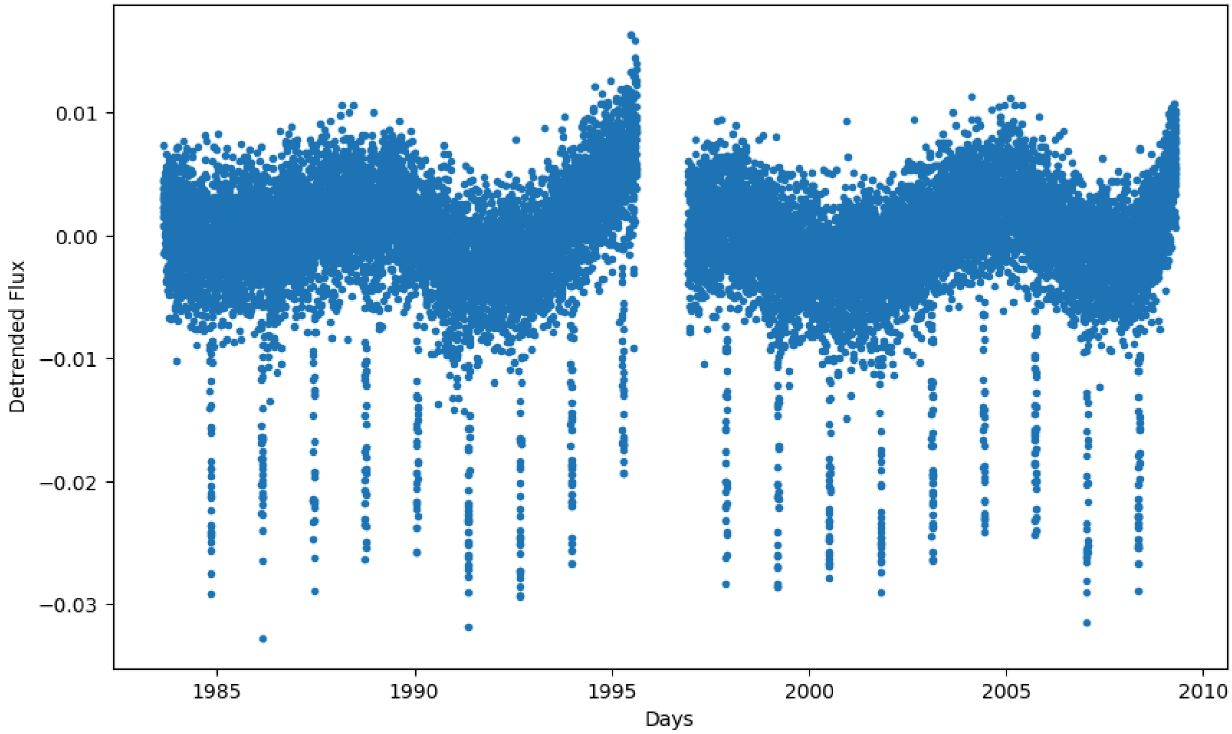


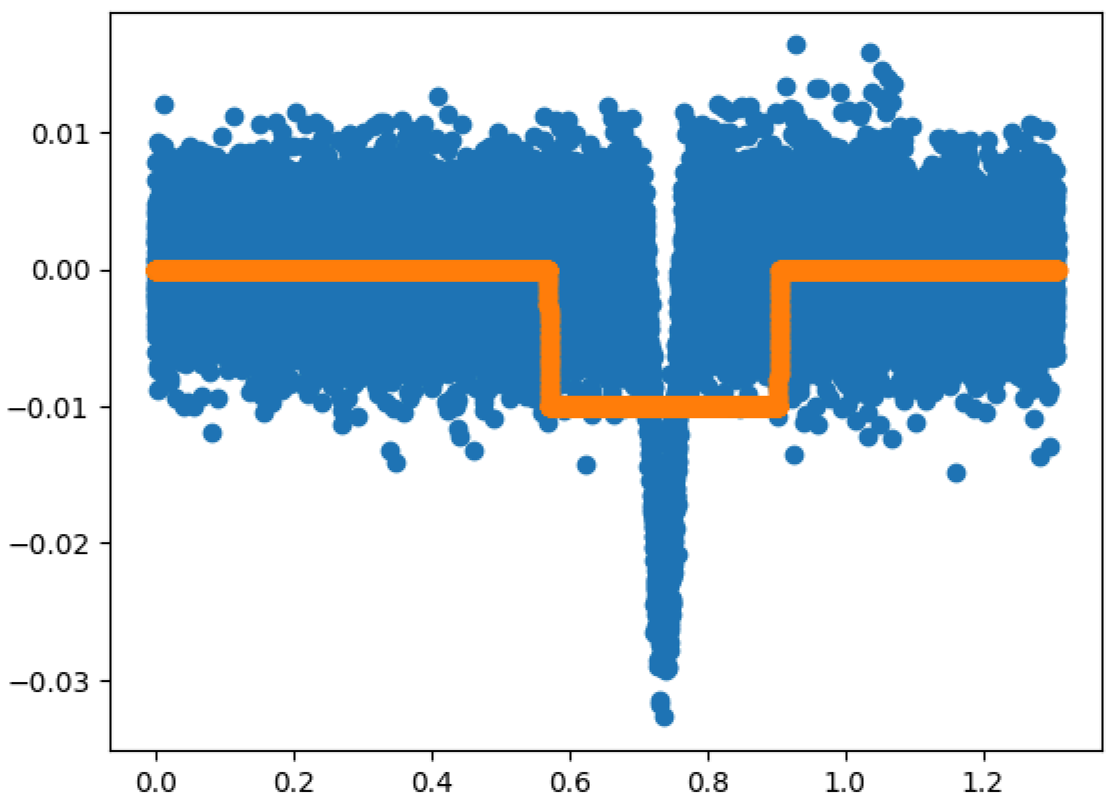


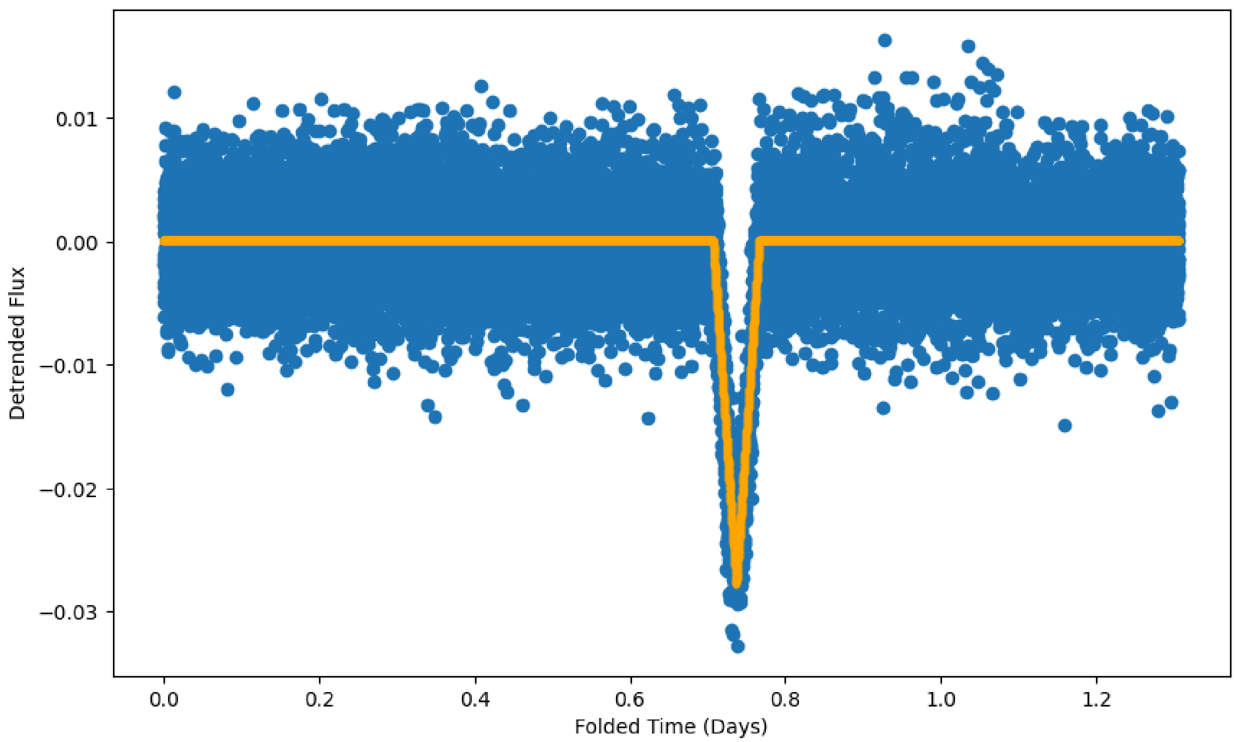
Flux
Days
2850
2825
2800
2775
2750
2725
2700
1985
1990
1995
2000
2005
2010



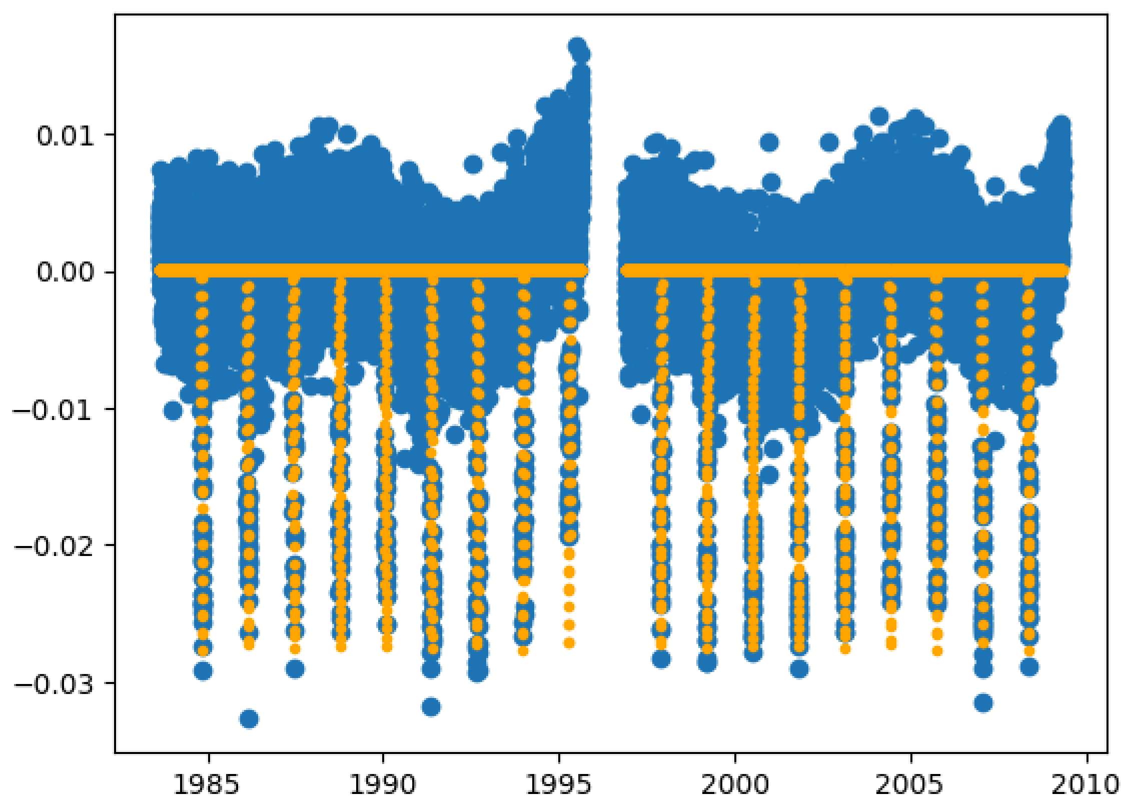
original data
in-transit data
background trend
Flux
Days
2850
2825
2800
2775
2750
2725
2700
1985
1990
1995
2000
2005
2010


0.01
0.00
−0.01
−0.02
−0.03
Detrended Flux
1985
1990
1995
2000
2005
2010
Days

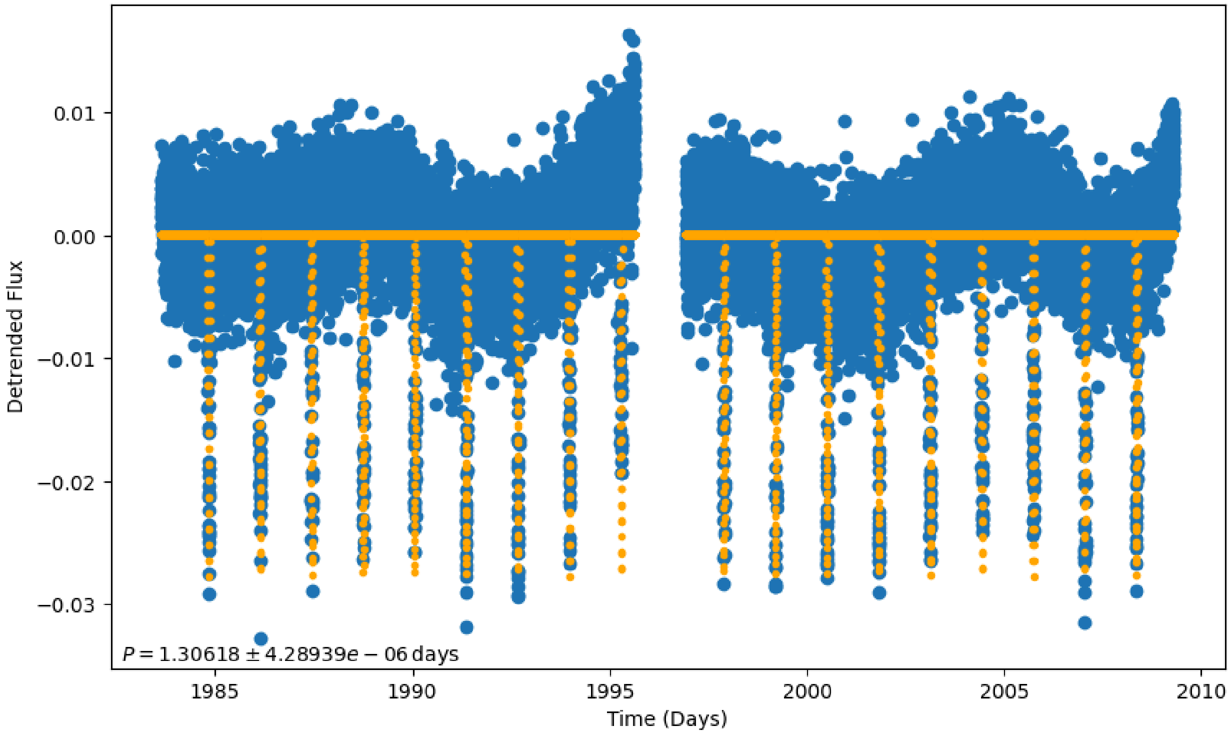

0.01
0.00
−0.01
−0.02
−0.03
0.0
0.2
0.4
0.6
0.8
1.0
1.2



Detrended Flux
Folded Time (Days)
0.01
0.00
−0.01
−0.02
−0.03
0.0
0.2
0.4
0.6
0.8
1.0
1.2



0.01
0.00
−0.01
−0.02
−0.03
1985
1990
1995
2000
2005
2010



0.01
0.00
−0.01
−0.02
−0.03
Detrended Flux
$P = 1.30618 \pm 4.28939e-06\,\mathrm{days}$
1985
1990
1995
2000
2005
2010
Time (Days)