This document compiles a list of issues I know about in my code for calculating mid eclipse times. It is not a comprehensive list, but just a few things I noticed that I think should be taken into consideration for future work. Please feel free to try and implement any ideas you may have to help solve these problems.

1. Choice of pipeline for downloading TESS lightcurves.

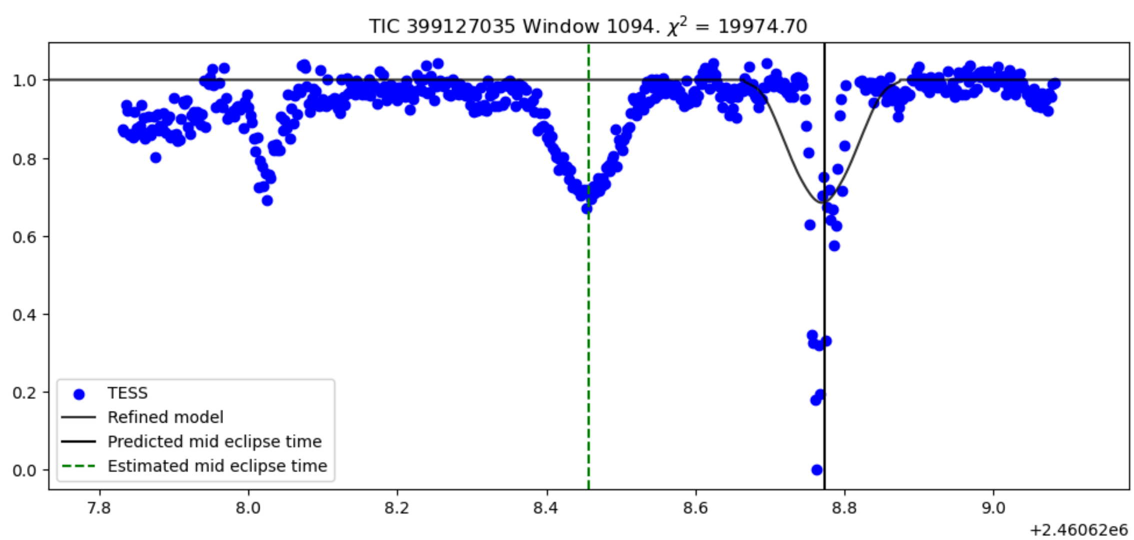
For this project, I chose to use QLP data for the TESS sources. This typically gives high quality lightcurves for a majority of bright sources in TESS, but it is by no means complete, nor is it the highest quality lightcurve that can be obtained from the FFI. Future works may explore either larger data volumes (i.e. Eleanor) or higher quality data (i.e. SPOC), however this isn't really a big issue, more of a consideration.

2. Physically motivated models.

Currently, when I create my models, I simply build them off of the data itself. While this will naturally give us a good fit to the data, it assumes that the signal we see is completely physical. Any artifacts left over in our dataset that do not represent the system itself will still be fit to and may bias our results. In an ideal world, we should be able to physically model these eclipses with tools like Phoebe or batman.

3. Fitting to artifacts

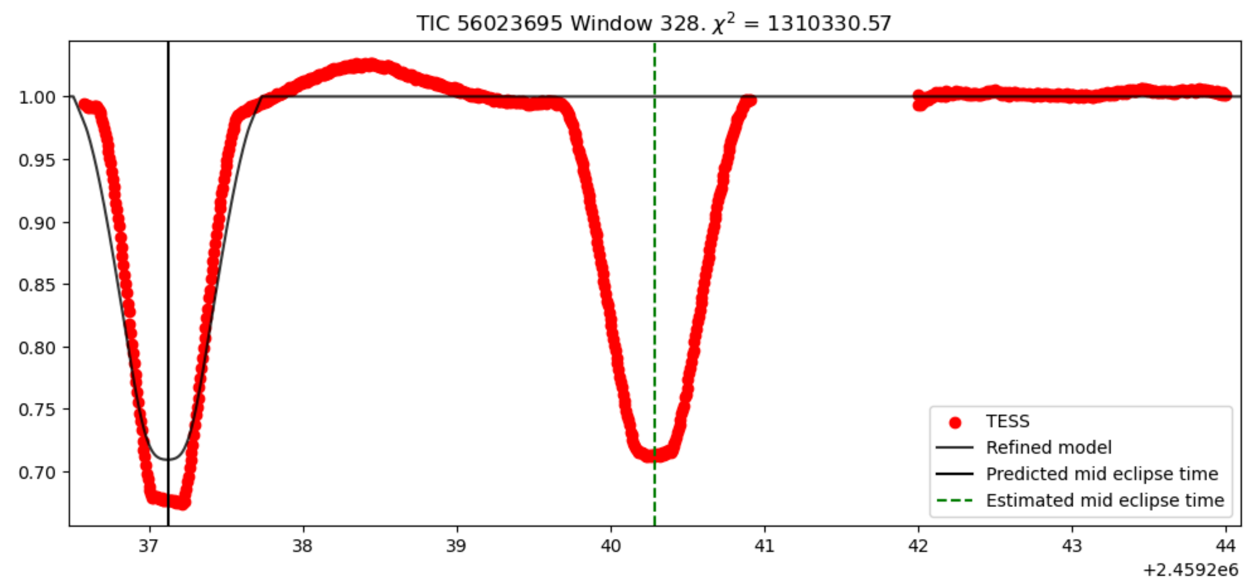
Due to the way the cross correlation works against our models, it preferentially fits to the deepest parts of the lightcurve even if there exists an eclipse that is a better fit.



For example, in the below plot, while it is clear that the eclipse occurs near the green dotted line (as predicted), some weird artifacting/data reduction has led to very low nonphysical decreases in flux at other points along the lightcurve. As such, the cross correlation fits to this deepest point even if it is clearly a worse fit. I don't think there is a super straightforward fix to this. It may be possible to look at the residuals around the eclipse itself, and bias a detection system towards low residuals at the eclipse in the model, ignoring high residuals if there is a good fit elsewhere. Otherwise, once you know the approximate maximum OC of the system, you could tune the window widths to try and only include the bare minimum amount of data as a refinement step and possibly exclude said artifacts. This isn't guaranteed to work though, and fails if the artifacts are close to the eclipses.

4. Highly elliptical systems.

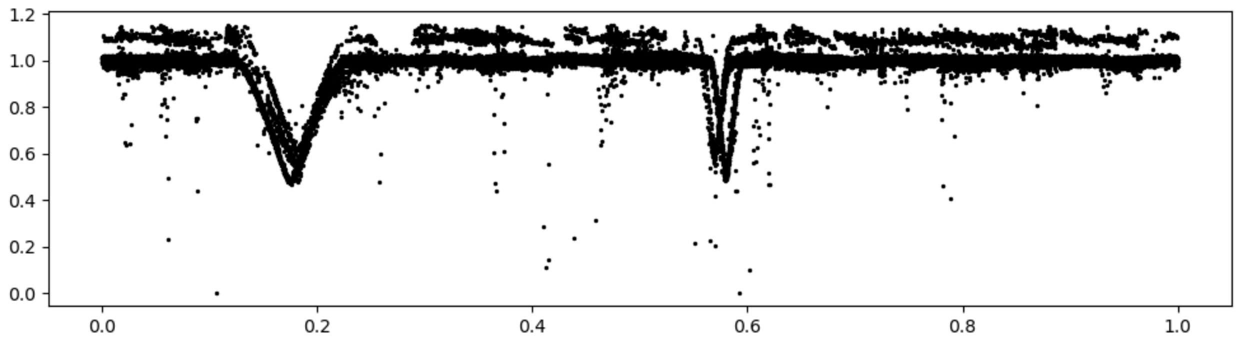
This is just an edge case with an implemented solution, but I think it is worth taking note of anyway. In the rare case that the system is highly elliptical, it would be possible to get 2 eclipses very close in time to each other and have them both included in every window like in the plot below.



Following on from the same logic as problem 3, the deeper eclipse will almost always be fit, even if it is the incorrect eclipse like in the above case. The solution here is to just narrow the window down until only one eclipse is visible.

5. Per-sector differences

This issue comes in at the lightcurve level. Since each lightcurve was generated independently, they were reduced/processed independently. This can lead to some discontinuities between different lightcurves for the same source, as shown below.

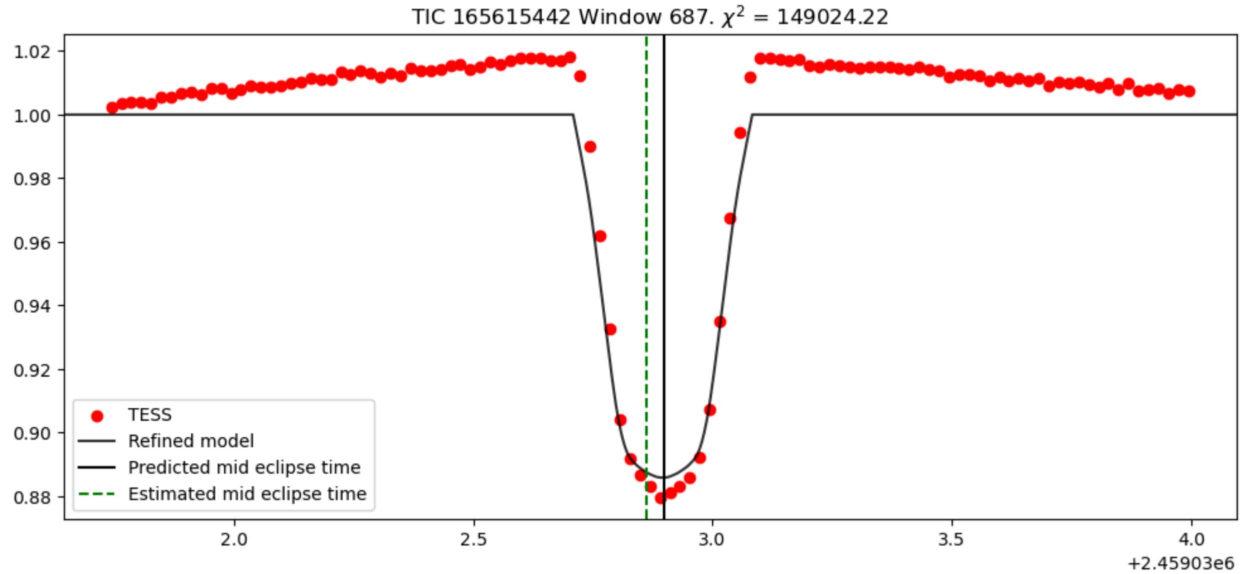


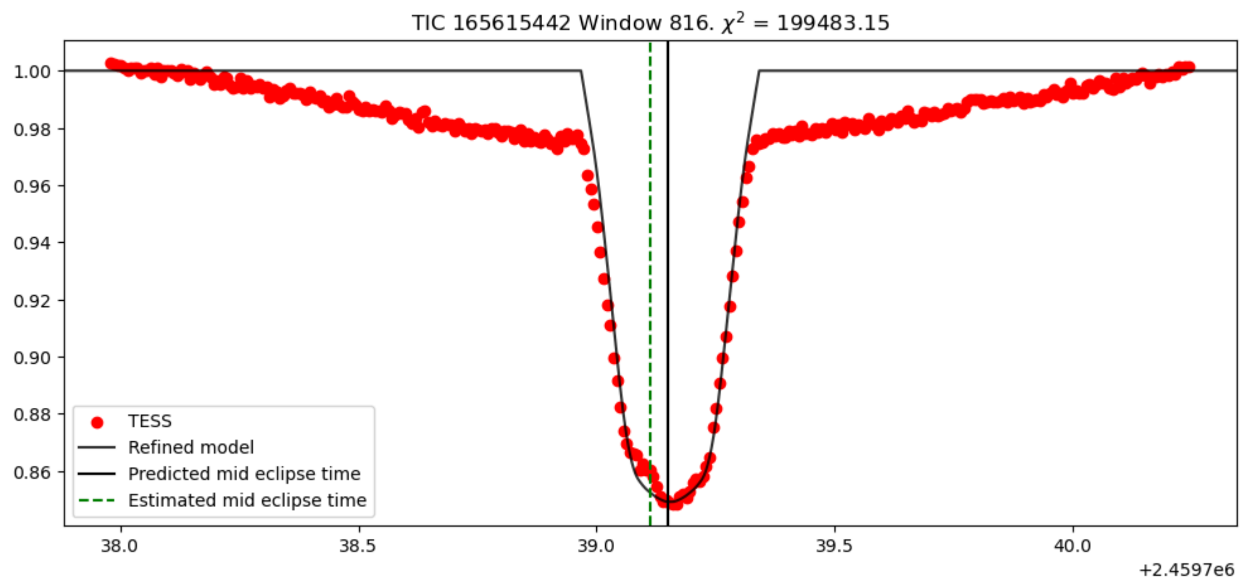
It is clear that some sectors have a slightly higher baseline flux, leading to this sort of ‘double’ effect. This could not be solved by renormalising, since the offending lightcurves jumped between fluxes of ~1.2 to ~1 between sampling points, pointing to some weird processing of the lightcurve when it was created. To deal with this, I just checked if the scatter was significantly larger than the error (see the too_noisy function in mid_eclipse_timing), however this process is not physically motivated, and the better solution would be to discard or find better lightcurves for these sectors.

6. Variable stars

My method implicitly assumes that the baseline flux of the source out of eclipse is ~1. I.e. that the stars exhibit no additional variability outside of their eclipses. This isn’t always true, as there is nothing stopping a variable star from being in an eclipsing binary.

The biggest issue here is the range of variability that stars can exhibit, making it difficult to find a universal solution. In some cases, the variability is linked to the period of the binary (i.e. if the binary is semi attached) like in the plot shown in problem 4 (note an increase in flux between the 2 eclipses). Other times (and more commonly), the variability is on a different period to the binary, meaning the flux around the eclipse will change every time it is observed (i.e. star spots or any of the other dozens of types of stellar variability).





I.e. in the plots above, we can see that the flux surrounding the eclipse behaves in very different ways. This is an issue because it makes it hard to properly estimate the exact width/shape of the eclipse, as it is a combination of eclipse effects and variable star effects. This is mostly mitigated by our approach of modelling the phase folded lightcurves rather than the time domain lightcurves, but in the case that you fit to time domain lightcurves, make sure to notice any stellar variability before taking note of the width/depth of the eclipse.