Identifying Planets Using Kalman Filters

Stat 624 Project

W. Zachary Horton

Brigham Young University

Abstract. Light measurements from distant stars can be used to identify if there are orbiting planets. These data are noisy and the effects of orbiting planets can be small. We use Kalman filters to reduce noise and build an outlier model to help classify stars. We propose a modified Kalman filter to deal with unnatural sources of noise and discuss its possible implications.

1 Introduction

Transit photometry is a method used to find planets orbiting stars in deep space. A transit is the event when a planet passes between a star and Earth. During a transit, the perceived brightness of that star will decrease briefly but shortly return to normal levels (The Planetary Society (2017)). Measuring light levels, or flux, over time produces light curves which can be analyzed through time-series or functional data methods. Regular and consistent drops in a light curve are strong evidence that there is a planet orbiting that star. When these drops are distinct or when the data are clear, transit photometry works well, however these are rarely the case. Photometric observations from telescopes such as the Keppler space telescope are often noisy both from the measurement devices and from actual variation in the brightness of a star caused by solar flares, dust clouds, or other phenomena. Another source of error is the telescope being pointed at the wrong region of the star where the orbiting planet may not fully cover. This will reduce the dimming effect and make it harder to differentiate between the noise and the actual dips. In this paper we impliment a Kalman filter to estimate the true state of brightness or flux. We propose a method of outlier searching to build a classification model for new stars to see if they have orbiting planets. We assess our method with simulation study and discuss the results. We propose a modified filter to help reduce unnatural sources of noise.

2 Methodology

2.1 Modeling the Data

We used an AR(1) model with noise to model the data. The model is formulated as

$$y_t \sim N(x_t, \sigma^2) \tag{2.1}$$

$$x_t \sim N(\phi x_{t-1}, \tau^2) \tag{2.2}$$

where y_t is the observed value at time t, x_t is the corresponding true state, σ^2 is called the measurement noise variance, τ^2 is called the process noise variance, and ϕ is an AR model coefficient. When using filtering methods, writing the model in state-space equations is usually required for implementing the algorithms. These are given as

$$y_t = x_t + v_t, \quad v_t \sim N(0, \sigma^2) \tag{2.3}$$

$$x_t = \phi x_{t-1} + w_t, \quad w_t \sim N(0, \tau^2) \tag{2.4}$$

This model assumes that the data follow a generally flat path that is centered at zero and that the noise is normally distributed with constant variance. These assumptions may not fit well in all cases of the data, however these assumptions make the filtering process efficient.

2.2 Kalman Filter

The Kalman filter is a method of estimating the true state of a process occuring over time given some noisy observations. It is an iterative procedure that takes only the previosly estimated state and the next data point into account. This makes Kalman filters very popular in situations where estimations are needed quickly because it does not require the full data set to make a prediction. In order to use a Kalman filter, the data model must be able to be written in terms of a state space model. The general state space model equations relevant to a Kalman filter are given by

$$y_t = Hx_t + v_t, \quad v_t \sim N(0, R) \tag{2.5}$$

$$x_t = Ax_{t-1} + Bu_t + w_t, \quad w_t \sim N(0, Q) \tag{2.6}$$

Whete y_t is the observation at time t, x_t is the true state of the process at time t, Q is the process noise variance, R is the measurement noise variance, u_t is a control signal, and A, B, and H are coefficients.

Comparing these to the model equations (2.3) and (2.4) a mapping is easy to see. This mapping is useful in simplifying the Kalman filter algorithm.

The source of noise is an important consideration in the Kalman filter as they are parameters, as we will see in the algorithm section. Measurement noise is the tendancy for a measurement device to vary from the true underlying process. In many contexts this is fixed and known such as in electrical measurment devices. Process noise is randomness introduced into the process itself. An example would be the altitude of a plane as it descends. Theoretically this should change smoothly, however turbulance introduces randomness in the actual altitude. In many cases, this form of noise should be very small compared to the process noise. The Kalman filter responds to these noise levels differently. When measurement noise is large, the filter will tend to create state estimates that are smoother and closer to the underlying process. When process noise is large, the filter will assume that the majority of noise in the process is due to actual process movement and will tend to trace the data. Accurately estimating the ratio between these two sources of noise is very important to the success of the Kalman filter because it governs how much the data are "believed" as opposed to the estimated states.

2.2.1 Kalman Filter Algorithm The Kalman filter algorithm is given in terms of the state-space equations given in equations (2.5) and (2.6), however we can simplify the algorithm formulas by pluggins in corresponding values from our model equations (2.3) and (2.4). The filter is implemented in two parts. First, using the provided state-space equation, the filter will make a preliminary prediction of the next state, denoted $\hat{x}_t^-$ using the previously estimated state, $\hat{x}_{t-1}$. Second, the filter will take a weighted average between the preliminary prediction and the observed value, y_t, essentially correcting the predicted state to obtain the state estimate $\hat{x}_t$. Below are the algorithm steps with needed equations and details needed to fit a Kalman filter given our current model choice.

1. Provide estimates for the initial state $\hat{x}_0$ and initial error variance P_0. These values analogously could be considered the state at time $t = 0$. The Kalman filter is not very sensitive to initial values, so a few rules of thumb will suffice. We set $\hat{x}_0 = 0$ due to our using of standardized data. The same logic could also be used for setting $P_0 = 1$, however, in practice we saw very few values of P_t greater than 1, so we chose $P_0 = 0.5$. In reality, the Kalman filter is very efficient and will converge quickly on an error variance, so almost any choice of P_0 will do except for 0.

2. Sequentially move through each data point and compute the following values

$$\hat{x}_t^- = \phi \hat{x}_{t-1} \tag{2.7}$$

$$P_t^- = \phi^2 P_{t-1} + \tau^2 \tag{2.8}$$

$$K_t = \frac{P_t^-}{P_t^- + \sigma^2} \tag{2.9}$$

$$\hat{x}_t = \hat{x}_t^- + K_t(y_t - \hat{x}_t^-) \tag{2.10}$$

$$P_t = (1 - K_t)P_t^- \tag{2.11}$$

Once this is completed for all data points then the values of $\hat{x}_t$ for $t = 1, \ldots, n$ serve as our state estimates. This yeilds the following distribution on the true state x_t

$$x_t \sim N(\hat{x}_t, P_t) \tag{2.12}$$

2.2.2 Parameter Estimation Unless the parameters ϕ, σ^2, and τ^2 are known (which on rare occassion they are), they must be estimated. We use maximum the likelihood function

$$L(\phi, \tau^2, \sigma^2 | \mathbf{y}) = \prod_{t=1}^{n} f(y_t | \hat{x}_t, P_t) \tag{2.13}$$

where f is the normal distribution density function and both $\hat{x}_t$ and P_t are the results of fitting a Kalman filter given ϕ, τ^2, and σ^2. Notice that the likelihood depends implicitly on the output of the Kalman filter. Due to the recursive nature of the likelihood function, maximization through analytic methods is difficult, as are methods that require closed form expressions for the gradient vector or Hessian matrix. For the sake of simplifying the implimentation, we use the L-BFGS-B method, a quasi-Newton optimization algorithm, on our likelihood function to find the best estimates of the unknown parameters. Note that maximization implies repeatedly fitting another Kalman filter with each iteration.

There are cases when the optimization technique fails due to the flat process model assumption being false. The failure takes the form of estimating the measurement noise variance, σ^2, to be 0 which causes the Kalman filter to simply return the original data values it was given. We want to assume a very low process variance in order to seperate noise from the changes caused by planets, so we will impose a fixed signal-to-noise ratio. This constraint both fixes the optimization failure and imposes something we already believed to be true. By looking at cases of the data where model assumptions were met, we chose a ratio of $\frac{\sigma^2}{\tau^2} = 25$, meaning there is 25 times more measurement noise than process noise.

2.3 Planet Identification

To identify planets, we chose to look at the distribution of standardized residuals. The logic follows that if a star has large, regular drops in light levels, then there will be a higher number of outliers than normal. Our method is to optimize the number of outliers and the outlier threshold to maximize the difference in percents between planet vs non-planet observations that satisfy those criteria. To optimize we set up a grid where the number of outliers ranged from 1 to 20 and the outlier threshold ranged from 2 to 20, ranges that captured the majority of all data cases, and chose the values that maximized the difference in percent.

3 Data

3.1 Data Source

All of the data presented in this paper were obtained from the Mikulski Archive for Space Telescopes (MAST). STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS5-26555. Support for MAST for non-HST data is provided by the NASA Office of Space Science via grant NNX09AF08G and by other grants and contracts. This paper includes data collected by the Kepler mission. Funding for the Kepler mission is provided by the NASA Science Mission directorate. The data were cleaned and compiled by kaggle.com user "kepplersmachines". The description we give in this section is inspired by his/her description. (NASA Science Mission (2017))

The data consist of 5657 light curves, each corresponding with a different star. Each curve consists of 3197 light intensity measurements, also called flux, taken over the space of 3 months. Each star has also been classified as wither "planet" or "non-planet" which indicates whether or not there is evidence for an orbiting planet.

3.2 Data Properties

In order to bring the data into closer alignment with the model assumptions, we standardize within each curve such that the underlying process is centered at 0 and the overall noise variance is 1. Figure 1 below shows an example light curve that demonstrates the level of noise in the standardized curves. Figure 2 shows what it looks like when there is a planet orbitting the star. Notice the scale between the two graphs and that the bulk of the noise is within the same region.

3.3 Model Assumption Checks

Not all of the data fit the model assumptions. Figure 3 demonstrates failure of the flat process assumption and failure of the constant variance assumption. When looking at the data as a whole, there aren't any

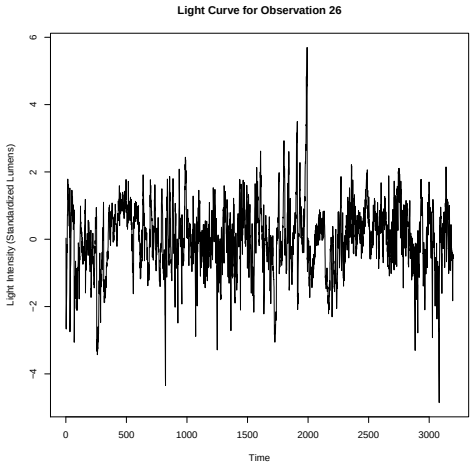

Fig. 1: Noisy Light Curve

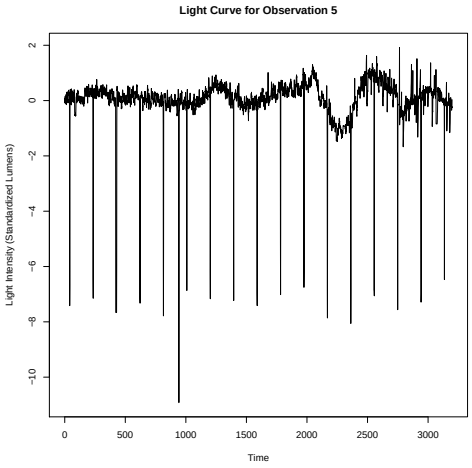

Fig. 2: Light Curve With Planet

extreme violations of the constant variance assumption, including the one pictured. However, there are a large portion of the data that do not follow the flat process assumption. A more complex model could account for this, however the calculations from the Kalman filter would become extremely difficult. Since the focus of this paper is predictive power, we chose to leave all the data in that did not break our optimization algorithm (which a few would). The assumption of being centered at zero was maintained due to our standardization of the curve points.

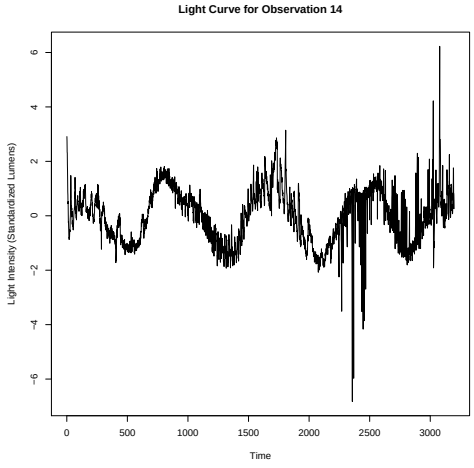

Fig. 3: Violation of Assumptions

4 Simulation Study

In this section we discuss the results of a simulation study done to verify our parameter estimation technique. It has been shown that the MLE is unbiased(Casella and Berger (2002)). This study shows that the MLE under constrained signal-to-noise ratio is also unbiased. Data were generated for this study by walking through the following algorithm.

- Use 0 as the initial value.
- For each subsequent data point, generate a random normal with the previous data point times ϕ as the mean and a variance of τ^2. This is process noise.
- Add a random normal value with mean 0 and variance σ^2 to each data point. This is measurement noise.

For each combination of parameters we generated data and estimated the parameters 350 times, allowing calculation of bias and MSE. Table 1 shows the combinations used and the results of the simulation.

True tau2	True sig2	True phi	Bias tau2	Bias sig2	Bias phi	MSE tau2	MSE sig2	MSE phi
0.008	0.2	0.2	0.334	8.348	0.163	0.115	71.717	0.159
0.008	0.2	0.9	0.131	3.285	-0.005	0.017	10.827	0.001
0.040	1.0	0.2	0.850	21.251	-0.301	0.742	463.760	0.219
0.040	1.0	0.9	0.265	6.616	0.002	0.070	43.916	0.001
0.400	10.0	0.2	2.300	57.501	-0.300	5.576	3484.909	0.261
0.400	10.0	0.9	0.624	15.597	-0.013	0.394	245.963	0.001

Table 1: Simulation Study Results

These results show that the procedure is biased for τ^2, ϕ and especially σ^2. An important consideration though is that when the true value of ϕ is close to 1, the results are must less biased and the MSE is much smaller. This is evidence that the optimization algorithm struggles to seperate the effects of the three variables when ϕ is small. This is expected since small values of ϕ will reduce the effect of process noise and exagerate the effect of measurement noise, confounding all three. Although these results further reveal inadequate model assumptions, the result in filtering is not as sensitive to biases as it is to relative sizes of the parameters. Figures 4 and 5 show how successful the filter is despite parameter estimation flaws.

5 Results and Discussion

Optimizing over the greatest difference in percent of residuals greater than a particular outlier threshold gave a best threshold value of 2 and a count of 13. Of the non-planet light curves, 35% have at least 13 points with a residual of 2 or more, which is less than the 46% of planet light curves under the same comparison.

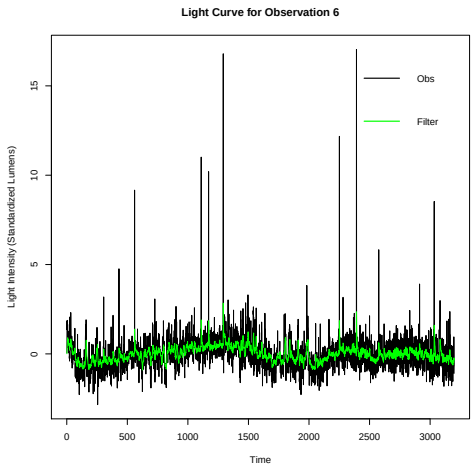

Fig. 4: Estimated Filter - No Planet

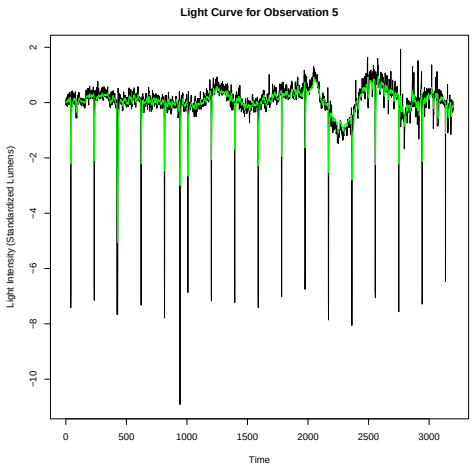

Fig. 5: Estimated Filter - Planet

This 11% difference is the largest possible value from our data. This means that, although these criterion are optimized, they have very little predictive capability.

These results come back to the fact that much of the data breaks model assumptions, which throws our estimations of noise off. A more flexible model would suit this problem much better, however that would complicate the implementation of the Kalman filter. Another critisism of our approach is that it assumes outliers are what indicate planets. In theory this would only help distinguish a small number of cases where the planets obscure a large portion of the stars they orbit. A more general approach would be to use functional data analysis to look for curve patterns and structures more subtle than outliers.

6 Modified Kalman Filter

Here we propose a modification to the Kalman filter and motivate future research. In analyzing filter performance it would be useful to filter out sources of "unnatural" noise. Unnatural noise could be defined as noise that is neither process noise nor measurement noise, perhaps probabilistic in nature. For example, the light emitted by a star will fluctuate due to solor flares and other surface activity. This would be considered process noise. Our readings of that same star will have noise introduced by the instability in the devices used, which would be considered measurement noise. Considering these two examples, it is clear that noise created by planets occassionally blocking the light belongs in its own category as a source of "unnatural" noise.

Currently, the Kalman filter has no way of dealing with this third noise source, which results in poor state estimates at those points. In order to filter these out, a modification must be made. A more complex model could be used but that complicates filter implimentation. Smoothing techniques like splines can be

used, but these do not have the advantages a Kalman filter has mainly that computations are extremely fast and only require the last estimated state instead of the whole dataset. We propose a modification be made to the Kalman filter algorithm: include a squared residual term in the gain. Equation (6.1) shows the modification to equation (2.9).

$$K_t = \frac{P_t^-}{P_t^- + \sigma^2 + (y_t - \hat{x}_t^-)^2} \tag{6.1}$$

Recall that K_t is what determines how much weight is given to the observation y_t. This modification forces the weight to become very small when the distance is very large. Although this method lacks proper mathematical foundation, it is heuristically sound and functions well in practice. Figure 6 shows the standard Kalman algorithm picking up large portions of unnatural noise in the outlier cases. Figure 7 shows how the modified filter handles the spikes much better.

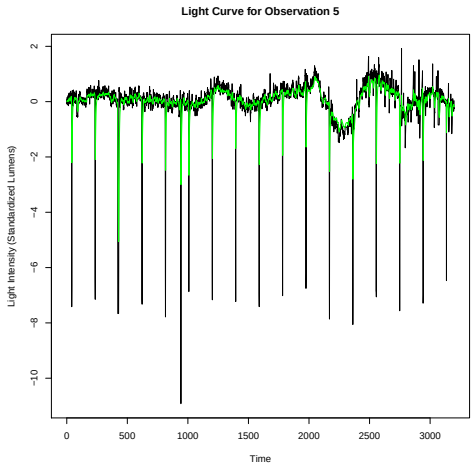

Fig. 6: Standard Filter

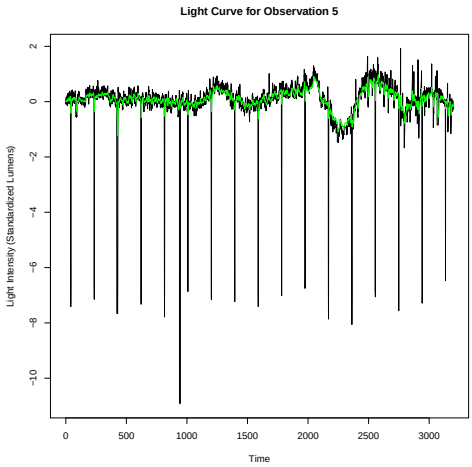

Fig. 7: Modified Filter

Notice how the filter appears to be the same in all other places except the spikes. Essentially this modification is adding the squared residual to the signal noise variance, changing our assumption about a constant noise variance. More research would reveal more stable and justifiable modifications that accomplish this same thing. This type of modification would be useful in fields such as wireless signal transmission where unnatural noise is common.

7 Conclusion

Kalman filters work well at filtering out the noise in our light data. With appropriate constraints we obtain reasonable flux estimates. The outlier searching method performed very poorly due to unmet model assump-

tions and a low correspondence of outliers with planet data. This method serves better as a preliminary tool to guide astronomers as to which regions deserve more thoughtful investigation. The modified Kalman algorithm, despite lacking mathematical background, performs well. Further research is required to turn its heuristic argument into a logical one, but a few test cases show promising results in filtering out unnatural variation.

Bibliography

Casella G, Berger R (2002) Statistical Inference, 2nd edn. Wadsworth Group

NASA Science Mission (2017) Kepler Telescope Time-Series Data. Accessed through Kaggle database, https://www.kaggle.com/keplersmachines/kepler-labelled-time-series-data

The Planetary Society (2017) Transit photometry http://www.planetary.org/explore/space-topics/exoplanets/transit-photometry.html