Draft version March 24, 2017
Preprint typeset using LaTeX style AASTeX6 with modifications

FAST AND SCALABLE GAUSSIAN PROCESS MODELING WITH APPLICATIONS TO ASTRONOMICAL TIME SERIES

Daniel Foreman-Mackey[1,2], Eric Agol[2], Ruth Angus[3,4], and Sivaram Ambikasaran[5]

[1]Sagan Fellow

[2]Astronomy Department, University of Washington, Seattle, WA

[3]Simons Fellow

[4]Department of Astronomy, Columbia University, New York, NY

[5]Department of Computational and Data Sciences, Indian Institute of Science, Bangalore, India

ABSTRACT

The growing field of large-scale time domain astronomy requires methods for probabilistic inference that are computationally tractable, even with large datasets. Gaussian Processes are a popular class of models used for this purpose but, since the computational cost scales as the cube of the number of data points, their application has been limited to relatively small datasets. In this paper, we present a method for Gaussian Process regression in one-dimension where the computational requirements scale linearly with the size of the dataset. This method exploits structure in the problem when the covariance function is expressed as a mixture of complex exponentials, without requiring evenly spaced observations or uniform noise. This form of covariance arises naturally when the process is a mixture of stochastically-driven damped harmonic oscillators – providing a physical motivation for and interpretation of this choice – but we also demonstrate that it is effective in many other cases. We present a mathematical description of the method, the details of the implementation, and a comparison to existing scalable Gaussian Process methods. We demonstrate the method by applying it to simulated and real astronomical time series datasets. These demonstrations are examples of probabilistic inference of stellar rotation periods, asteroseismic oscillation spectra, and transiting planet parameters. In all of these cases, we infer posterior distributions over models for the power spectrum of the process that generated these irregularly sampled time series without computing a standard algebraic estimator. The method is flexible, fast, and most importantly, interpretable, with a wide range of potential applications within astronomical data analysis and beyond. We provide well-tested and documented open-source implementations of this method in C++, Python, and Julia.

Keywords: methods: data analysis — methods: data analysis — methods: statistical — asteroseismology — stars: rotation — planetary systems

1. INTRODUCTION

Gaussian Processes (GPs; Rasmussen & Williams 2006) are popular stochastic models for time-series analysis. For GP modeling, a functional form is chosen to describe the autocovariance of the data and the parameters of this function are fit for or marginalized. In the astrophysical literature, GPs have been used to model stochastic variability in light curves of stars (Brewer & Stello 2009), active galactic nuclei (Kelly et al. 2014), and the logarithmic flux of X-ray binaries (Uttley et al. 2005). They have also been used as models for the cosmic microwave background (Bond & Efstathiou 1987; Bond et al. 1999; Wandelt & Hansen 2003), correlated instrumental noise (Gibson et al. 2012), and spectroscopic calibration (Czekala et al. 2017; Evans et al. 2015). While these models are widely applicable, their use has been limited, in practice, by the computational cost and scaling. In general, the cost of computing a GP likelihood scales as the cube of the number of data points $\mathcal{O}(N^3)$ and in the current era of large time domain surveys – with as many as $\sim 10^{4-9}$ targets with $\sim 10^{3-5}$ observations each — this scaling is prohibitive.

In this paper, we present a method for computing a class of GP models that scales linearly with the number of data points $\mathcal{O}(N)$ for one dimensional data sets. This method is a generalization of a method developed by Ambikasaran (2015) that was, in turn, built on intuition from a twenty year old paper (Rybicki & Press 1995). For this method to be applicable, the data must be one-dimensional and the covariance function must have a specific form. However, there is no further constraint on the data or the model. In particular, the measurements don't need to be evenly spaced and the uncertainties can be heteroscedastic (non-uniform). This method is especially appealing compared to other similar methods – we return to these below – because it is exact, flexible, simple, and fast.

In the following pages, we motivate the general problem of GP regression, describe the previously published scalable method (Rybicki & Press 1995; Ambikasaran 2015) and our generalization, and demonstrate the model's application on various real and simulated data sets. Alongside this paper, we have released well-tested and documented open source implementations written in C++, Python, and Julia. These are available online at GitHub `https://github.com/dfm/celerite` and Zenodo *DFM: add zenodo archive*.

2. GAUSSIAN PROCESSES

GP are stochastic models consisting of a mean function $\mu_{\boldsymbol{\theta}}(\boldsymbol{x})$ and a covariance, autocorrelation, or "kernel" function $k_{\boldsymbol{\alpha}}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ parameterized by the parameters $\boldsymbol{\theta}$

and $\boldsymbol{\alpha}$ respectively. Under this model, the log-likelihood of observing a dataset

$$\boldsymbol{y} = \begin{pmatrix} y_1 & \cdots & y_N \end{pmatrix}^{\mathrm{T}} \tag{1}$$

at coordinates

$$X = \begin{pmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_N \end{pmatrix}^{\mathrm{T}} \tag{2}$$

is

$$\ln p(\boldsymbol{y} \,|\, X, \boldsymbol{\theta}, \boldsymbol{\alpha}) = -\frac{1}{2} \boldsymbol{r}_{\boldsymbol{\theta}}{}^{\mathrm{T}} \, {K_{\boldsymbol{\alpha}}}^{-1} \, \boldsymbol{r}_{\boldsymbol{\theta}} - \frac{1}{2} \ln \det K_{\boldsymbol{\alpha}} - \frac{N}{2} \ln (2\pi) \tag{3}$$

where

$$\boldsymbol{r}_{\boldsymbol{\theta}} = \begin{pmatrix} y_1 - \mu_{\boldsymbol{\theta}}(\boldsymbol{x}_1) & \cdots & y_N - \mu_{\boldsymbol{\theta}}(\boldsymbol{x}_N) \end{pmatrix}^{\mathrm{T}} \tag{4}$$

is the vector of residuals and the elements of the covariance matrix K are given by $[K_{\boldsymbol{\alpha}}]_{nm} = k_{\boldsymbol{\alpha}}(\boldsymbol{x}_n, \boldsymbol{x}_m)$. The maximum likelihood values for the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$ for a given dataset $(\boldsymbol{y}, X)$ can be found by maximizing Equation (3) with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$ using a non-linear optimization routine (Nocedal & Wright 2006). Similarly, probabilistic constraints on $\boldsymbol{\theta}$ and $\boldsymbol{\alpha}$ can be obtained by multiplying the likelihood by a prior $p(\boldsymbol{\theta}, \boldsymbol{\alpha})$ and using a Markov Chain Monte Carlo (MCMC) algorithm to sample the joint posterior probability density.

GP models have been widely used across the physical sciences but their application is generally limited to small datasets because the cost of computing the inverse and determinant of the matrix $K_{\boldsymbol{\alpha}}$ is $\mathcal{O}(N^3)$. In other words, this cost is proportional to the cube of the number of data points N. This means that for large datasets, every evaluation of the likelihood quickly becomes computationally intractable. In this case, the use of non-linear optimization or MCMC is no longer practical.

In the following section, we present a method for substantially improving this scaling in many circumstances. We call our method and its implementations celerite.[1] The celerite method requires using a specific model for the covariance $k_{\boldsymbol{\alpha}}(\boldsymbol{x}_n, \boldsymbol{x}_m)$ and, although it has some limitations, we demonstrate in subsequent sections that it can increase the computational efficiency of many astronomical data analysis problems. The main limitation of this method is that it can only be applied to one-dimensional datasets, where by "one-dimensional" we mean that the *input coordinates* $\boldsymbol{x}_n$ are scalar, $\boldsymbol{x}_n \equiv t_n$.[2] Furthermore, the covariance function for the celerite method is "stationary". In other words, $k_{\boldsymbol{\alpha}}(t_n, t_m)$ is only a function of $\tau_{nm} \equiv |t_n - t_m|$.

3. THE CELERITE MODEL

[1] The name celerite comes from the French word *célerité* meaning the speed of light in a vacuum.

[2] We use t as the input coordinate because one-dimensional GPs are often applied to time series data but this isn't a real restriction and the celerite method can be applied to *any* one-dimensional dataset.

To scale GP models to larger datasets, Rybicki & Press (1995) presented a method of computing the first term in Equation (3) in $\mathcal{O}(N)$ operations when the covariance function is given by

$$k_{\boldsymbol{\alpha}}(\tau_{nm}) = \sigma_n^2\,\delta_{nm} + a\,\exp(-c\,\tau_{nm}) \tag{5}$$

where $\{{\sigma_n}^2\}_{n=1}^N$ are the measurement uncertainties, δ_{nm} is the Kronecker delta, and $\boldsymbol{\alpha} = (a,\,c)$. The intuition behind this method is that, for this choice of $k_{\boldsymbol{\alpha}}$, the inverse of $K_{\boldsymbol{\alpha}}$ is tridiagonal and can be computed with a small number of operations for each data point. Subsequently, Ambikasaran (2015) generalized this method to arbitrary mixtures of exponentials

$$k_{\boldsymbol{\alpha}}(\tau_{nm}) = \sigma_n^2\,\delta_{nm} + \sum_{j=1}^{J} a_j\,\exp(-c_j\,\tau_{nm}) \quad . \tag{6}$$

In this case, the inverse is dense but Equation (3) can still be evaluated in $\mathcal{O}(N\,J^2)$, where J is the number of components in the mixture.

This kernel function can be made even more general by introducing complex parameters $a_j \to a_j \pm i\,b_j$ and $c_j \to c_j \pm i\,d_j$. In this case, the covariance function becomes

$$\begin{aligned} k_{\boldsymbol{\alpha}}(\tau_{nm}) = \sigma_n^2\,\delta_{nm} + \sum_{j=1}^{J} \Bigg[&\frac{1}{2}(a_j + i\,b_j)\,\exp\left(-(c_j + i\,d_j)\,\tau_{nm}\right) \\ &+ \frac{1}{2}(a_j - i\,b_j)\,\exp\left(-(c_j - i\,d_j)\,\tau_{nm}\right)\Bigg] \end{aligned} \tag{7}$$

and, for this function, Equation (3) can still be evaluated with $\mathcal{O}(N\,J^2)$ operations. The details of this method and a few implementation considerations are outlined in the following section but we first discuss some properties of this covariance function.

By rewriting the exponentials in Equation (7) as sums of sine and cosine functions, we can see the autocorrelation structure is defined by a mixture of quasiperiodic oscillators

$$\begin{aligned} k_{\boldsymbol{\alpha}}(\tau_{nm}) = \sigma_n^2\,\delta_{nm} + \sum_{j=1}^{J} \Big[&a_j\,\exp\left(-c_j\,\tau_{nm}\right)\cos\left(d_j\,\tau_{nm}\right) \\ &+ b_j\,\exp\left(-c_j\,\tau_{nm}\right)\sin\left(d_j\,\tau_{nm}\right)\Big] \quad . \end{aligned} \tag{8}$$

For clarity, we refer to the argument within the sum as a "celerite term" for the remainder of this paper. The Fourier transform[3] of this covariance function is the power spectral density (PSD) of the process and it is given by

$$S(\omega) = \sum_{j=1}^{J} \sqrt{\frac{2}{\pi}}\,\frac{(a_j\,c_j + b_j\,d_j)\,({c_j}^2 + {d_j}^2) + (a_j\,c_j - b_j\,d_j)\,\omega^2}{\omega^4 + 2\,({c_j}^2 - {d_j}^2)\,\omega^2 + ({c_j}^2 + {d_j}^2)^2} \quad . \tag{9}$$

[3] Here and throughout we have defined the Fourier transform of the function $f(t)$ as $F(\omega) = (2\,\pi)^{-1/2}\,\int_{-\infty}^{\infty} f(t)\,e^{i\,\omega\,t}\,\mathrm{d}t$.

The physical interpretation of this model isn't immediately obvious and we return to a more general discussion of the physical intuition in a moment but we start by investigating some special cases.

If we set the imaginary amplitude b_j for some component j to zero, that term of Equation (8) becomes

$$k_j(\tau_{nm}) = a_j \, \exp\left(-c_j \, \tau_{nm}\right) \cos\left(d_j \, \tau_{nm}\right) \tag{10}$$

and the PSD for the this component is

$$S_j(\omega) = \frac{1}{\sqrt{2\,\pi}} \, \frac{a_j}{c_j} \left[\frac{1}{1 + \left(\frac{\omega - d_j}{c_j}\right)^2} + \frac{1}{1 + \left(\frac{\omega + d_j}{c_j}\right)^2} \right] \quad . \tag{11}$$

This is the sum of two Lorentzian or Cauchy distributions with width c_j centered on $\omega = \pm d_j$. This model can be interpreted intuitively as a quasiperiodic oscillator with amplitude $A_j = a_j$, quality factor $Q_j = d_j \, (2\,c_j)^{-1}$, and period $P_j = 2\,\pi\,{d_j}^{-1}$.

Similarly, setting both b_j and d_j to zero, we get an Ornstein–Uhlenbeck process

$$k_j(\tau_{nm}) = a_j \, \exp\left(-c_j \, \tau_{nm}\right) \tag{12}$$

with the PSD

$$S_j(\omega) = \sqrt{\frac{2}{\pi}} \, \frac{a_j}{c_j} \, \frac{1}{1 + \left(\frac{\omega}{c_j}\right)^2} \quad . \tag{13}$$

Finally, we note that the product of two terms of the form found inside the sum in Equation (8) can also be re-written as a sum with updated parameters

$$k_j(\tau) \, k_k(\tau) = e^{-\tilde{c}\,\tau} \left[\tilde{a}_+ \, \cos(\tilde{d}_+ \, \tau) + \tilde{b}_+ \, \sin(\tilde{d}_+ \, \tau) + \tilde{a}_- \, \cos(\tilde{d}_- \, \tau) + \tilde{b}_- \, \sin(\tilde{d}_- \, \tau)\right] \tag{14}$$

where

$$\begin{aligned}
\tilde{a}_\pm &= \frac{1}{2} \left(a_j \, a_k \pm b_j \, b_k\right) && (15)\\
\tilde{b}_\pm &= \frac{1}{2} \left(b_j \, a_k \mp a_j \, b_k\right) && (16)\\
\tilde{c} &= c_j + c_k && (17)\\
\tilde{d}_\pm &= d_j \mp d_k \quad . && (18)
\end{aligned}$$

Therefore, the method described in the following section can be used to perform scalable inference on large datasets for any model, where the kernel function is a sum or product of celerite terms.

4. IMPLEMENTATION & PERFORMANCE

Rybicki & Press (1995) demonstrated that the inverse of a matrix K, with elements given by Equation (5), could be computed efficiently by taking advantage of the structure of this covariance function. Ambikasaran (2015) generalized this computation

to apply to the full mixture of J terms in Equation (7) and derived an equally efficient method for computing the determinant of K.

4.1. *An example*

To provide some insight for this method, we follow Ambikasaran (2015) and start by working through a simple example. In this case, we assume that we have three data points $\{y_1, y_2, y_3\}$ observed at times $\{t_1, t_2, t_3\}$ with measurement variances $\{{\sigma_1}^2, {\sigma_2}^2, {\sigma_3}^2\}$ and we would like to compute the likelihood of these data under a GP model with the covariance function

$$k(\tau_{nm}) = \sigma_n^2\,\delta_{nm} + a\,\exp(-c\,\tau_{nm}) \quad . \tag{19}$$

This is the result of setting $J = 1$, $b = 0$, and $d = 0$ in Equation (7) and is the model studied by Rybicki & Press (1995). To demonstrate that the likelihood of this model can be computed in $\mathcal{O}(N)$, we write out the full system of equations that must be solved to apply the inverse of K and compute the first term of Equation (3). In matrix notation, this is

$$K\,\boldsymbol{z} = \boldsymbol{y}, \tag{20}$$

$$\begin{pmatrix} a + {\sigma_1}^2 & a\,e^{-c\,\tau_{2,1}} & a\,e^{-c\,\tau_{3,1}} \\ a\,e^{-c\,\tau_{2,1}} & a + {\sigma_2}^2 & a\,e^{-c\,\tau_{3,2}} \\ a\,e^{-c\,\tau_{3,1}} & a\,e^{-c\,\tau_{3,2}} & a + {\sigma_3}^2 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \tag{21}$$

where our goal is to solve for the unknown vector $\boldsymbol{z}$ for a given matrix K and vector $\boldsymbol{y}$. In Equation (20), we have assumed that the mean function is zero but a non-zero mean could be included by replacing $\boldsymbol{y}$ by $\boldsymbol{r_\theta}$ as defined in Section 2. Now, if we introduce the variables

$$u_n = e^{-c\,\tau_{n+2,n+1}}\,u_{n+1} + a\,z_{n+1} \tag{22}$$

where $u_N = 0$, and

$$g_n = e^{-c\,\tau_{n+1,n}}\,g_{n-1} + e^{-c\,\tau_{n+1,n}}\,z_n \tag{23}$$

where $g_0 = 0$, the system of equations can be rewritten as

$$(a + {\sigma_1}^2)\,z_1 + e^{-c\,\tau_{2,1}}\,u_1 = y_1 \tag{24}$$

$$a\,g_1 + (a + {\sigma_2}^2)\,z_2 + e^{-c\,\tau_{3,2}}\,u_2 = y_2 \tag{25}$$

$$a\,g_2 + (a + {\sigma_3}^2)\,z_3 = y_3 \quad . \tag{26}$$

Rewriting the system defined by Equation (22) through Equation (26) as a matrix equation shows the benefit that this seemingly trivial reformulation provides:

$$\begin{pmatrix} a+{\sigma_1}^2 & e^{-c\,\tau_{2,1}} & 0 & 0 & 0 & 0 & 0 \\ e^{-c\,\tau_{2,1}} & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & a & e^{-c\,\tau_{3,2}} & 0 & 0 \\ 0 & 0 & a & a+{\sigma_2}^2 & e^{-c\,\tau_{3,2}} & 0 & 0 \\ 0 & 0 & e^{-c\,\tau_{3,2}} & e^{-c\,\tau_{3,2}} & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & a \\ 0 & 0 & 0 & 0 & 0 & a & a+{\sigma_3}^2 \end{pmatrix} \begin{pmatrix} z_1 \\ u_1 \\ g_1 \\ z_2 \\ u_2 \\ g_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} y_1 \\ 0 \\ 0 \\ y_2 \\ 0 \\ 0 \\ y_3 \end{pmatrix}$$

Following Ambikasaran (2015) we call this the "extended" system and rewrite Equation (20) as

$$K_{\text{ext}}\,\boldsymbol{z}_{\text{ext}} \quad = \quad \boldsymbol{y}_{\text{ext}} \quad . \tag{27}$$

Even though K_{ext} is a larger matrix than the K we started with, it is now sparse with banded structure that can be exploited to solve the system efficiently. In particular, sparse solvers are available that can perform a LU-decomposition of matrices like this in $\mathcal{O}(N)$ operations – instead of the $\mathcal{O}(N^3)$ that would be required in general – and we can use these algorithms to solve our system exactly because the target vector $\boldsymbol{z}$ is a subset of the elements of $\boldsymbol{z}_{\text{ext}}$.

In the following section we discuss this method more generally, but it's worth noting a few important facts that can already be seen in this example. First, the fundamental reason why this matrix K can be solved efficiently is the following property of exponentials

$$e^{-c\,(t_3-t_2)}\,e^{-c\,(t_2-t_1)} = e^{-c\,(t_3-t_2+t_2-t_1)} = e^{-c\,(t_3-t_1)} \tag{28}$$

and it is important to note that this property does not extend to other common covariance functions like the "exponential-squared" function

$$k(\tau) \propto e^{-c\,\tau^2} \quad . \tag{29}$$

Furthermore, our derivation of the extended matrix requires that the data points be monotonically sorted in time. Neither of these properties will be satisfied in general for multidimensional inputs and all of our following discussion assumes a sorted one-dimensional dataset.

Ambikasaran (2015) demonstrated two key facts that allow us to use this extended matrix formalism in practice. First, even if the covariance function is a mixture of exponentials, the extended matrix will still be banded with a bandwidth that scales linearly with the number of components, J. Second, Ambikasaran (2015) proved that the absolute value of the determinant of K_{ext} is equal to the absolute value of the determinant of K. This means we can use this extended matrix formalism to compute

the marginalized likelihood in $\mathcal{O}(N)$ operations.

4.2. *The algorithm*

In this section, we generalize the method from the previous section to the covariance function given by Equation (8). This derivation follows Ambikasaran (2015) but it includes explicit treatment of complex parameters, and their complex conjugates.

In the case of the full celerite covariance function (Equation 8), we introduce the following auxiliary variables in analogy to the u_n and g_n that we introduced in the previous section

$$\begin{aligned}
\phi_{n,j} &= e^{-c_j\,\tau_{n+1,n}}\,\cos\left(d_j\,\tau_{n+1,n}\right) && (30)\\
\psi_{n,j} &= -e^{-c_j\,\tau_{n+1,n}}\,\sin\left(d_j\,\tau_{n+1,n}\right) && (31)\\
g_{n,j} &= \phi_{n,j}\,g_{n-1,j} + \phi_{n,j}\,z_n + \psi_{n,j}\,h_{n-1,j} && (32)\\
h_{n,j} &= \phi_{n,j}\,h_{n-1,j} - \psi_{n,j}\,z_n - \psi_{n,j}\,g_{n-1,j} && (33)\\
u_{n,j} &= \phi_{n+1,j}\,u_{n+1,j} + a_j\,z_{n+1} + \psi_{n+1,j}\,v_{n+1,j} && (34)\\
v_{n,j} &= \phi_{n+1,j}\,v_{n+1,j} - b_j\,z_{n+1} - \psi_{n+1,j}\,u_{n+1,j} && (35)
\end{aligned}$$

with the boundary conditions

$$g_{0,j} = 0 \quad , \quad h_{0,j} = 0 \quad , \quad u_{N,j} = 0 \quad , \quad \text{and} \quad v_{N,j} = 0 \tag{36}$$

for all j. Using these variables and some algebra, we find that the following expression

$$\sum_{j=1}^{J}\left[a_j\,g_{n,j} + b_j\,h_{n,j}\right] + \left[{\sigma_n}^2 + \sum_{j=1}^{J} a_j\right] + \sum_{j=1}^{J}\left[\phi_{n,j}\,u_{n,j} + \psi_{n,j}\,v_{n,j}\right] \quad = \quad r_{\boldsymbol{\theta},n} \tag{37}$$

is equivalent to the target matrix equation

$$K\,\boldsymbol{z} \quad = \quad \boldsymbol{r}_{\boldsymbol{\theta}} \tag{38}$$

if $r_{\boldsymbol{\theta},n}$ is the n-th element of the residual vector $\boldsymbol{r}_{\boldsymbol{\theta}}$ defined in Section 2. Equation (30) through Equation (37) define a banded matrix equation in the "extended" space and, as before, this can be used to solve for $K^{-1}\,\boldsymbol{r}_{\boldsymbol{\theta}}$ and $\det K$ in $\mathcal{O}(N)$ operations. Figure 1 shows a pictorial representation of the sparsity pattern of the extended matrix K_ext. Given this definition of K_ext, the corresponding extended vectors $\boldsymbol{z}_\mathrm{ext}$ and $\boldsymbol{r}_\mathrm{ext}$ are defined schematically as

$${\boldsymbol{z}_\mathrm{ext}}^\mathrm{T} = \begin{pmatrix} z_1 & u_{1,j} & v_{1,j} & g_{1,j} & h_{1,j} & z_2 & u_{2,j} & \cdots & h_{N-1,j} & z_N \end{pmatrix} \tag{39}$$

and

$${\boldsymbol{r}_\mathrm{ext}}^\mathrm{T} = \begin{pmatrix} r_{\boldsymbol{\theta},1} & 0 & 0 & 0 & 0 & r_{\boldsymbol{\theta},2} & 0 & \cdots & 0 & r_{\boldsymbol{\theta},N} \end{pmatrix} \quad . \tag{40}$$

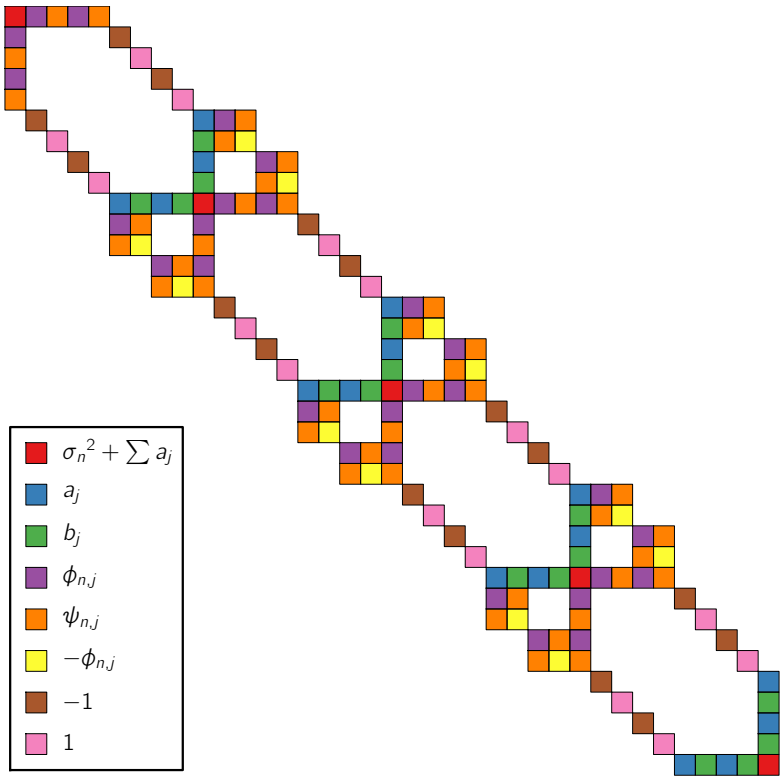

Figure 1. A pictorial representation of the sparse extended matrix $K_{\rm ext}$ with $N = 5$ and $J = 2$. Each colored block corresponds to a non-zero entry in the matrix as described in the legend.

After constructing the extended matrix (using a compact storage format), the extended matrix can be factorized using a LU-decomposition[4] routine optimized for band or sparse matrices. This decomposition can then be used to compute the determinant of K, solve $K^{-1}\,\boldsymbol{r}_{\boldsymbol{\theta}}$, and subsequently calculate the marginalized likelihood in Equation (3).

In practice, it is worth treating terms with $b_j = 0$ and $d_j = 0$ as a special case because, for this term j, $\psi_{n,j}$, $h_{n,j}$, and $v_{n,j}$ will also be identically zero for all n. Removing these trivial rows from the extended matrix results in factor of two increase in the computational efficiency for these terms. In our implementation, we refer to "real" terms as those where $b_j = 0$ and $d_j = 0$, and the general terms are called "complex".

4.3. *Implementation considerations & scaling*

The extended system defined in the previous section is sparse with typically fewer than a few percent non-zero entries and band structure. In this section, we empirically investigate the performance and scaling of three different algorithms for solving this

[4] Even though $K_{\rm ext}$ is symmetric, it is not positive definite so a Cholesky solver cannot be used for increased efficiency.

extended system:

1. `vanilla`: A simple algorithm for computing the LU decomposition for banded matrices using Gaussian elimination (Press et al. 1992; Press et al. 2007),
2. `lapack`: The general banded LU decomposition implementation from LAPACK[5] (Anderson et al. 1999) using optimized BLAS routines,[6] and
3. `sparse`: A general sparse LU solver – the `SparseLU` solver from Eigen (Guennebaud et al. 2010) – that exploits the sparsity but not the band structure.

The theoretical scaling for a band LU decomposition is $\mathcal{O}(N\,J^3)$ because the dimension of the extended matrix scales as $N\,J$ and the bandwidth scales with J (Press et al. 1992; Press et al. 2007). Ambikasaran (2015) found an empirical scaling of $\mathcal{O}(N\,J^2)$ that used the sparse LU decomposition implemented in the SuperLU package (Demmel et al. 1999). We find that, while the `vanilla` solver scales as expected, the `lapack` implementation scales empirically as $\mathcal{O}(N\,J^2)$ and offers the fastest solves for $J \gtrsim 8$ on all platforms that we tested.

The benchmark experiments shown here were performed on a MacBook Pro with two 2.6 GHz CPUs but we find similar results on a Dell workstation with 16 2.7 GHz CPUs and running Ubuntu. Figure 2 shows how the cost of computing Equation (3) scales with N and J using the `vanilla` solver. As expected theoretically, the scaling is linear in N for all N and cubic in J for large J. Figure 3 and Figure 4 are the same plots for the `lapack` and `sparse` solvers respectively. For each of these optimized solvers, the empirical scaling is $\mathcal{O}(N\,J^2)$ at the cost of some extra overhead. Therefore, for $J \lesssim 5$ or 10 the `vanilla` solver is more efficient than the other algorithms. The real world performance of celerite depends on the specific platform, hardware, and LAPACK/BLAS implementation but we have found qualitatively similar results across popular platforms and state-of-the-art libraries.

5. CELERITE AS A MODEL OF STELLAR VARIATIONS

We now turn to a discussion of celerite as a model of astrophysical variability. A common concern in the context of GP modeling in astrophysics is the lack of physical motivation for the choice of kernel functions. Kernels are often chosen simply because they are popular with little consideration of the impact of this decision. While this isn't necessarily a problem for the inferred results, in this section we discuss an exact physical interpretation of the celerite kernel that will be applicable to many astrophysical systems but especially the time-variability of stars.

Many astronomical objects are variable on timescales determined by their physical structure. For phenomena such as stellar (asteroseismic) oscillations, variability is

[5] We use the `dgbtrf` and `dgbtrs` methods from LAPACK.

[6] The experiments below use the Intel Math Kernel Library (MKL): `https://software.intel.com/en-us/intel-mkl`.

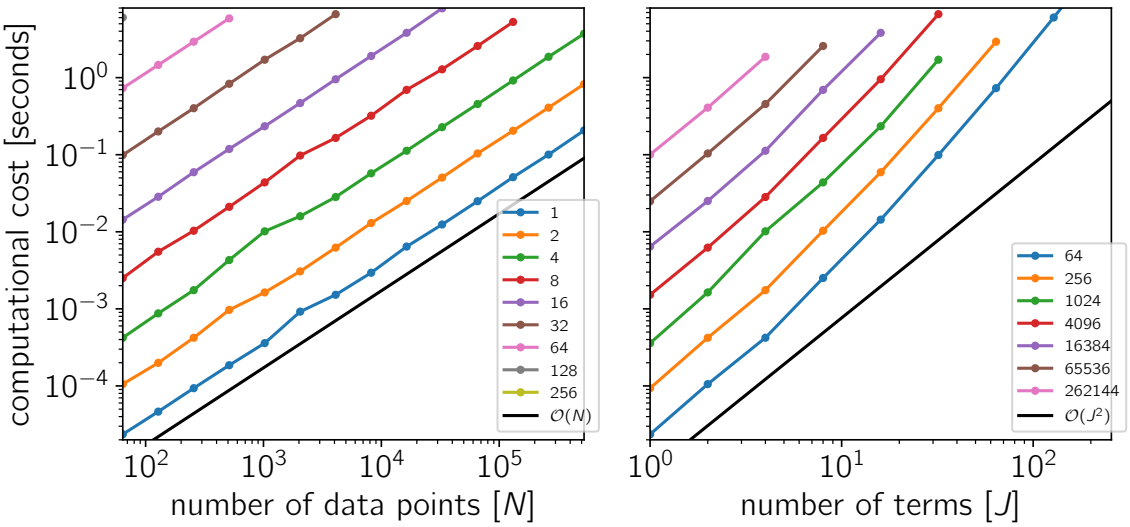

Figure 2. A benchmark showing the computational scaling of celerite using the `vanilla` band solver as a function of the number of data points and the number of terms. *(left)* The cost of computing Equation (3) with a covariance matrix given by Equation (8) as a function of the number of data points N. The different colors show the cost for different numbers of terms J as listed in the legend. To guide the eye, the black line shows linear scaling in N. *(right)* The same information plotted as a function of J for different values of N. The legend shows the value of N for each color and the black line shows quadratic scaling in J.

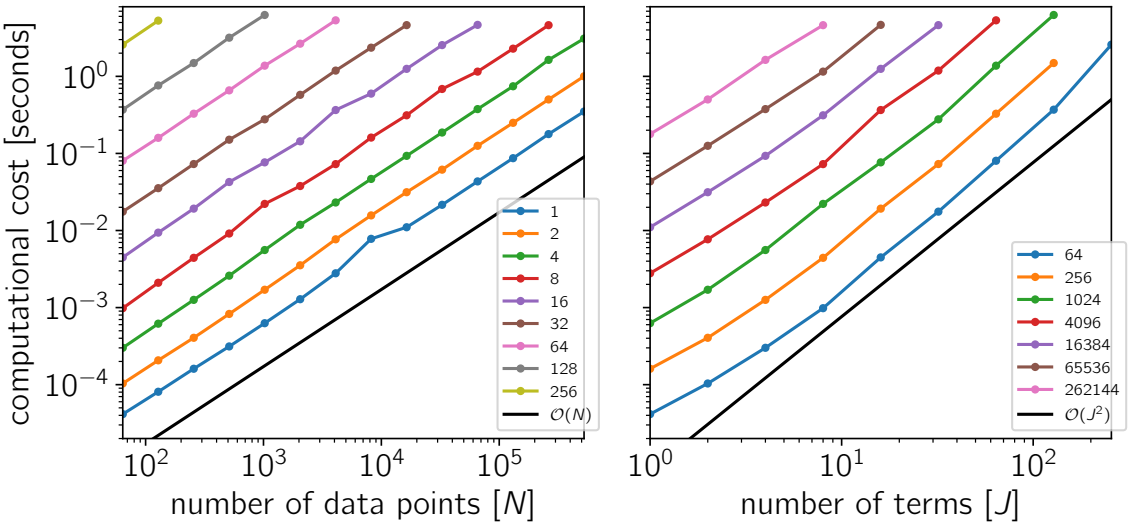

Figure 3. The same as Figure 2 but using the `lapack` solver with the MKL BLAS library.

excited by noisy physical processes and grows most strongly at the characteristic timescale but is also damped due to dissipation in the system. These oscillations are strong at resonant frequencies determined by the internal stellar structure, which are both excited and damped by convective turbulence.

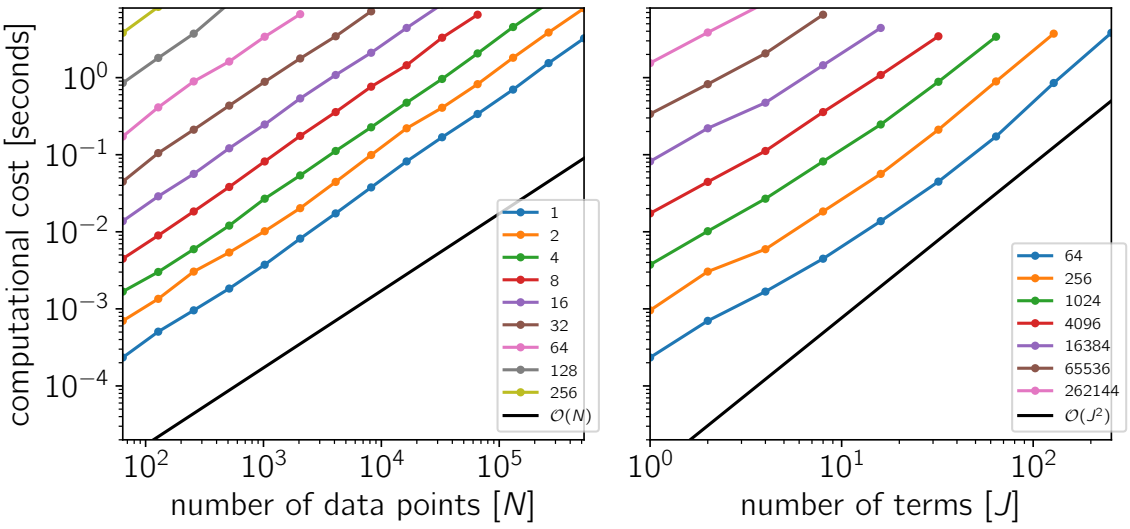

Figure 4. The same as Figure 2 but using the `sparse` solver from Eigen (Guennebaud et al. 2010).

To relate this physical picture to celerite, we consider the dynamics of a stochastically-driven damped simple harmonic oscillator. The differential equation for this system is

$$\left[\frac{\mathrm{d}^2}{\mathrm{d}t^2} + \frac{\omega_0}{Q}\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2\right] y(t) = \epsilon(t) \tag{41}$$

where ω_0 is the frequency of the undamped oscillator, Q is the quality factor of the oscillator, and $\epsilon(t)$ is a stochastic driving force. If $\epsilon(t)$ is white noise, the PSD of this process is (Anderson et al. 1990)

$$S(\omega) = \sqrt{\frac{2}{\pi}}\,\frac{S_0\,\omega_0^4}{(\omega^2-\omega_0^2)^2 + \omega_0^2\omega^2/Q^2} \tag{42}$$

where S_0 is proportional to the power at $\omega = \omega_0$, $S(\omega_0) = \sqrt{2/\pi}\,S_0\,Q^2$. The power spectrum in Equation (42) matches Equation (9) if

$$a_j = S_0\,\omega_0\,Q \tag{43}$$

$$b_j = \frac{S_0\,\omega_0\,Q}{\sqrt{4\,Q^2-1}} \tag{44}$$

$$c_j = \frac{\omega_0}{2\,Q} \tag{45}$$

$$d_j = \frac{\omega_0}{2\,Q}\sqrt{4\,Q^2-1} \quad , \tag{46}$$

for $Q \geq \frac{1}{2}$. For $0 < Q \leq \frac{1}{2}$, Equation (42) can be captured by a pair of celerite terms with parameters

$$\begin{aligned} a_{j\pm} &= \frac{1}{2}\,S_0\,\omega_0\,Q\,\left[1 \pm \frac{1}{\sqrt{1-4\,Q^2}}\right] \\ b_{j\pm} &= 0 \\ c_{j\pm} &= \frac{\omega_0}{2\,Q}\,\left[1 \mp \sqrt{1-4\,Q^2}\right] \\ d_{j\pm} &= 0 \quad . \end{aligned} \tag{47}$$

For these definitions, the kernel is

$$k(\tau) = S_0\,\omega_0\,Q\,e^{-\frac{\omega_0\,\tau}{2Q}}\,\begin{cases} \cosh\left(\eta\,\omega_0\,\tau\right) + \frac{1}{2\,\eta\,Q}\,\sinh\left(\eta\,\omega_0\,\tau\right), & 0 < Q < 1/2 \\ 2\,(1+\omega_0\,\tau), & Q = 1/2 \\ \cos\left(\eta\,\omega_0\,\tau\right) + \frac{1}{2\,\eta\,Q}\,\sin\left(\eta\,\omega_0\,\tau\right), & 1/2 < Q \end{cases} \tag{48}$$

where $\eta = |1 - (4\,Q^2)^{-1}|^{1/2}$. We note that, because of the damping, the characteristic oscillation frequency in this model, d_j, for any finite quality factor $Q > 1/2$, is not equal to the frequency of the undamped oscillator, ω_0.

The power spectrum in Equation (42) has several limits of physical interest:

- For $Q = 1/\sqrt{2}$, Equation (42) simplifies to

$$S(\omega) = \sqrt{\frac{2}{\pi}}\,\frac{S_0}{(\omega/\omega_0)^4 + 1} \quad . \tag{49}$$

  This functional form is commonly used to model for the background granulation noise in asteoreseismic and helioseismic (Harvey 1985; Michel et al. 2009; Kallinger et al. 2014) analyses. The kernel for this value of Q is

$$k(\tau) = S_0\,\omega_0\,e^{-\frac{1}{\sqrt{2}}\,\omega_0\,\tau}\,\cos\left(\frac{\omega_0\,\tau}{\sqrt{2}} - \frac{\pi}{4}\right) \quad . \tag{50}$$

- Substituting $Q = 1/2$, Equation (42) becomes

$$S(\omega) = \sqrt{\frac{2}{\pi}}\,\frac{S_0}{\left[(\omega/\omega_0)^2 + 1\right]^2} \tag{51}$$

  with the corresponding covariance function (using Equation 8 and Equation 47)

$$\begin{aligned} k(\tau) &= \lim_{f\to 0}\,\frac{1}{2}\,S_0\,\omega_0\,\left[(1+1/f)\,e^{-\omega_0\,(1-f)\,\tau} + (1-1/f)\,e^{-\omega_0\,(1+f)\,\tau}\right] &(52)\\ &= S_0\,\omega_0\,e^{-\omega_0\,\tau}\,\left[1+\omega_0\,\tau\right] &(53)\end{aligned}$$

  or equivalently (using Equation 8 and Equation 43)

$$\begin{aligned} k(\tau) &= \lim_{f\to 0}\,S_0\,\omega_0\,e^{-\omega_0\,\tau}\,\left[\cos(f\,\tau) + \frac{\omega_0}{f}\,\sin(f\,\tau)\right] &(54)\\ &= S_0\,\omega_0\,e^{-\omega_0\,\tau}\,\left[1+\omega_0\,\tau\right] \quad . &(55)\end{aligned}$$

This covariance function is also known as the Matérn-3/2 function (Rasmussen & Williams 2006). This suggests that the Matérn-3/2 covariance can be approximated using the celerite framework with a small value of f in Equation (54) but we caution that this might lead to numerical issues for the solver.

- Finally, in the limit of large Q, the model approaches a high quality oscillation with frequency ω_0 and covariance function

$$k(\tau) \approx S_0\,\omega_0\,Q\,\exp\left(-\frac{\omega_0\,\tau}{2\,Q}\right)\,\cos\left(\omega_0\,\tau\right) \quad . \tag{56}$$

Figure 5 shows a plot of the PSD for these limits and several other values of Q. From this figure, it is clear that for $Q \le 1/2$, the model has no oscillatory behavior and that for large Q, the shape of the PSD near the peak frequency approaches a Lorentzian.

These special cases demonstrate that the stochastically-driven simple harmonic oscillator provides a physically motivated model that is flexible enough to describe a wide range of stellar variations. Low $Q \approx 1$ can capture granulation noise and high $Q \gg 1$ is a good model for asteroseismic oscillations. In practice, we take a sum over oscillators with different values of Q, S_0, and ω_0 to give a sufficient accounting of the power spectrum stellar time series. Since this kernel is exactly described by the exponential kernel, the likelihood (Equation 3) can be evaluated for a time series with N measurements in $\mathcal{O}(N)$ operations using the celerite method described in the previous section.

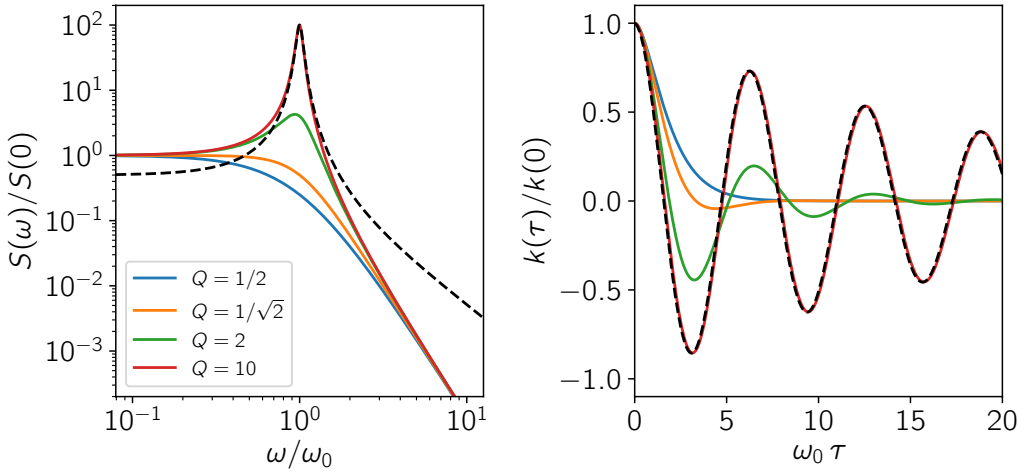

Figure 5. (left) The power spectrum of a stochastically-driven simple harmonic oscillator (Equation 42) plotted for several values of the quality factor Q. For comparison, the dashed line shows the Lorentzian function from Equation (11) with $c_j = \omega_0/(2\,Q) = 1/20$ and normalized so that $S(d_j)/S(0) = 100$. (right) The corresponding autocorrelation functions with the same colors.

6. EXAMPLES WITH SIMULATED DATA

To demonstrate the application of celerite, we start by inferring posterior constraints on the parameters of a GP model applied to several simulated datasets with known properties. In the following section, we expand on these examples by applying celerite to real datasets. In the first example, we demonstrate that celerite can be used to infer the power spectrum of a process when the data are generated from a celerite model. In the second example, we demonstrate that celerite can be used as an effective model even if the true process cannot be represented in the space of allowed models. This is an interesting example because, when analyzing real data, we rarely have any fundamental reason to believe that the data were generated by a GP model with a specific kernel. Even in these cases, GPs can be useful effective models and celerite provides computational advantages over other GP methods.

6.1. *Recovery of a celerite process*

In this first example, we simulate a dataset using a known celerite process and fit it with celerite to demonstrate that valid inferences can be made in this idealized case. The simulated dataset is shown in the left panel of Figure 6 and it was generated using a SHO kernel (Equation 48) with parameters $S_0 = 1$, $\omega_0 = e^2$, and $Q = e^2$. The true PSD is shown as a dashed line in the right panel of Figure 6. We applied log-uniform priors to all of the parameters and used emcee (Foreman-Mackey et al. 2013) to sample the joint posterior density and computed the marginalized posterior inference of the PSD. This inference is shown in the right panel of Figure 6 as a blue contour indicating 68% of the posterior mass. It is clear from this figure that, as expected, the inference correctly reproduces the true PSD.

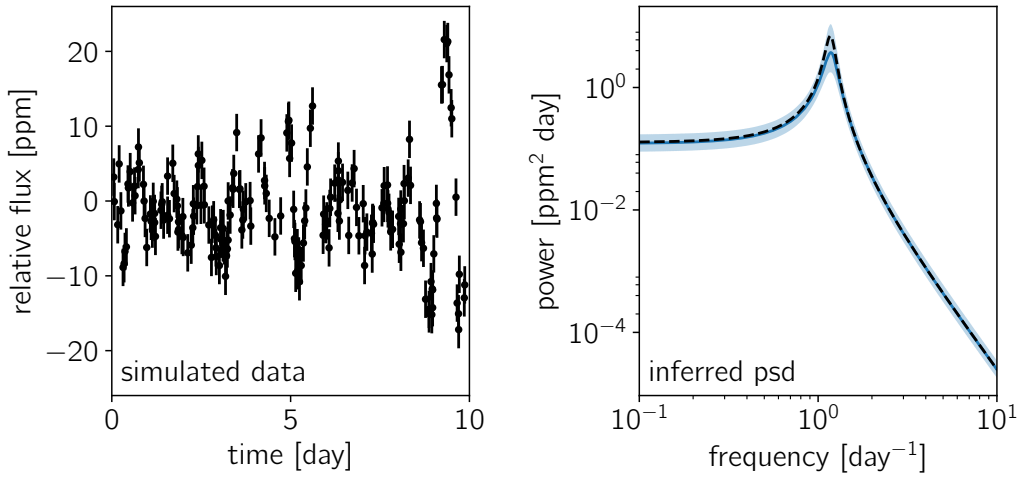

Figure 6. (left) A simulated dataset. (right) The inferred PSD – the blue contours encompass 68% of the posterior mass – compared to the true PSD (dashed black line).

6.2. *Inferences with the "wrong" model*

For this example, we simulate a dataset using a known GP model with a kernel outside of the support of a celerite process. This means the true autocorrelation of the process can never be correctly represented by the model that we're using to fit but we use this example to demonstrate that, at least in this case, valid inferences can still be made about the physical parameters of the model.

For this example the data are simulated from a quasiperiodic GP with the kernel

$$k_{\rm true}(\tau) = \alpha\,\exp\left(-\frac{\tau^2}{2\,\lambda^2}\right)\,\cos\left(\frac{2\,\pi\,\tau}{P_{\rm true}}\right) \tag{57}$$

where $P_{\rm true}$ is the fundamental period of the process. This autocorrelation structure corresponds to the power spectrum

$$S_{\rm true}(\omega) = \frac{\lambda\,\alpha}{2}\,\left[\exp\left(-\frac{\lambda^2}{2}\,\left(\omega - \frac{2\,\pi}{P_{\rm true}}\right)^2\right) + \exp\left(-\frac{\lambda^2}{2}\,\left(\omega + \frac{2\,\pi}{P_{\rm true}}\right)^2\right)\right] \tag{58}$$

which, for large values of ω, falls off exponentially. When compared to Equation (9) – which, for large ω, goes as ω^{-4} at most – it is clear that a celerite model can never perfectly reproduce the structure of this process. That being said, we demonstrate that rigorous inferences can be made about $P_{\rm true}$ even with an effective model. The left panel of Figure 7 shows the simulated dataset. We then fit this simulated data using the product of two SHO terms (Equation 48) where one of the terms has $S_0 = 1$ and $Q = 1/\sqrt{2}$ and the other has $\omega_0 = 2\,\pi/P$. We note that using Equation (14), the product of two celerite terms can also be expressed using celerite. We apply log-uniform priors on all the parameters and use emcee to sample the joint posterior probability for all of the parameters. The inferred distribution for the parameter P is shown in the right panel of Figure 7 and compared to the true period $P_{\rm true}$ and the inferences made using the correct model (Equation 57). The inference made using this effective celerite model are indistinguishable from the inferences made using the correct model but substantially less computation time is required for the celerite inference.

7. EXAMPLES WITH REAL DATA

In this section, we demonstrate several use cases of celerite when applied to real datasets. Each of these examples touches on an active area of research so we limit our examples to be qualitative in nature and do not claim that celerite is the optimum method but we hope these examples encourage interested readers to investigate the applicability of celerite to their research.

All of the following examples show time domain datasets with a clear bias in favor of large homogeneous photometric surveys but these methods can similarly be applied to spectroscopy, where wavelength – instead of time – is the independent coordinate and other one-dimensional domains (see Czekala et al. 2017, for a potential application).

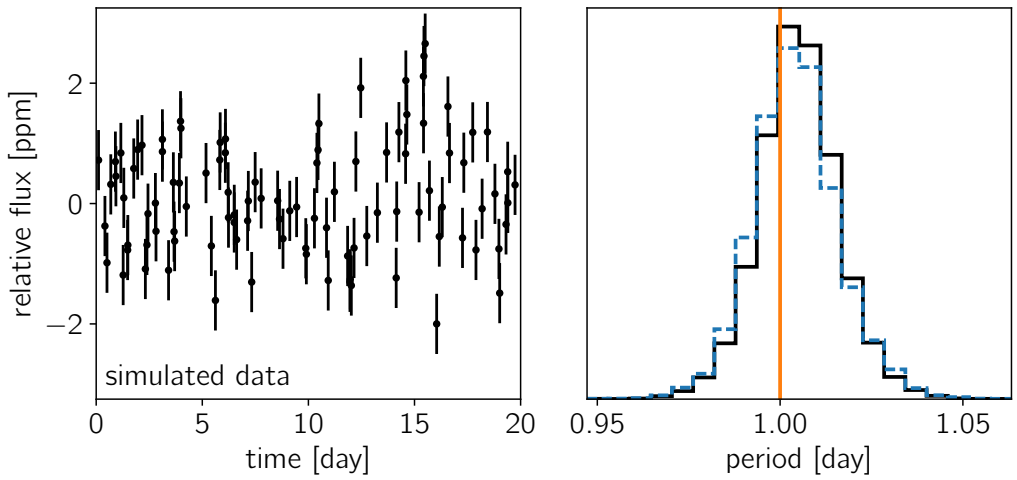

Figure 7. (left) A simulated dataset. (right) The inferred period of the process. The true period is indicated by the vertical orange line, the posterior inference using the correct model is shown as the blue dashed histogram, and the inference made using the "wrong" effective model is shown as the black histogram.

7.1. *Asteroseismic oscillations*

The asteroseismic oscillations of thousands of stars were measured using light curves from the Kepler mission (Gilliland et al. 2010; Huber et al. 2011; Chaplin et al. 2011, 2013; Stello et al. 2013) and asteroseismology is a key science driver for many of the upcoming large scale photometric surveys (Campante et al. 2016; Rauer et al. 2014; Gould et al. 2015). Most asteroseismic analyses have been limited to relatively high signal-to-noise oscillations because the standard methods based on statistics of the empirical periodogram of the data cannot be used to formally propagate the measurement uncertainties to the constraints on physical parameters – instead relying on bootstrapped uncertainty estimators (Huber et al. 2009) – but more sophisticated methods that compute the likelihood function in the time domain scale poorly to state-of-the-art datasets (Brewer & Stello 2009; Corsaro & Ridder 2014).

celerite alleviates these problems by providing a physically motivated probabilistic model that can be evaluated efficiently even for large datasets. In practice, one would model the star as a mixture of stochastically-driven simple harmonic oscillators where the amplitudes and frequencies of the oscillations are computed using a physical model and evaluate the probability of the observed time series using a GP where the PSD is a sum of terms given by Equation (42). This gives us a method of computing the likelihood function for the parameters of the physical model (for example, $\nu_{\max}$ and $\Delta\nu$, or other, more fundamental parameters) *conditioned on the observed time series* in $\mathcal{O}(N)$ operations. In other words, celerite provides a computationally efficient framework that can be combined with physically-motivated models of stars and numerical inference methods to make rigorous probabilistic measurements of asteroseismic parameters in the time domain. We expect that this has the potential to push asteroseismic analysis to lower signal-to-noise datasets and we hope to revisit

this idea in a subsequent paper.

To demonstrate the method, we use a very simple heuristic model where the PSD is given by a mixture of 8 components with amplitudes and frequencies specified by $\nu_{\max}$, $\Delta\nu$, and several nuisance parameters. The first term is used to capture the granulation "background" (Kallinger et al. 2014) using Equation (49) with two free parameters S_g and ω_g. The remaining 7 terms are given by Equation (42) where Q is a nuisance parameter shared between terms and the frequencies are given by

$$\omega_{0,\,j} = 2\,\pi\,(\nu_{\max} + j\,\Delta\nu + \epsilon) \tag{59}$$

and the amplitudes are given by

$$S_{0,\,j} = \frac{A}{Q^2}\,\exp\left(-\frac{[j\,\Delta\nu + \epsilon]^2}{2\,W^2}\right) \tag{60}$$

where j is an integer running from -3 to 3 and ϵ, A, and W are shared nuisance parameters. This model could be easily extended to include small frequency splitting and $\nu_{\max}$ and $\Delta\nu$ could be replaced by the fundamental physical parameters of the star.

To demonstrate the applicability of this model, we apply it to infer the asteroseismic parameters of the giant star KIC 11615890, observed by the Kepler Mission. The goal of this example is to show that, even for a low signal-to-noise dataset with a short baseline, it is possible to infer asteroseismic parameters with formal uncertainties that are consistent with the parameters inferred with a much larger dataset. Looking forward to TESS (Ricker et al. 2014; Campante et al. 2016), we measure $\nu_{\max}$ and $\Delta\nu$ using only one month of Kepler data and compare our results to the results inferred from the full 4 year baseline of the Kepler mission. For KIC 11615890, the published asteroseismic parameters measured using several years of Kepler observations are (Pinsonneault et al. 2014)

$$\nu_{\max} = 171.94 \pm 3.62\,\mu\text{Hz} \quad \text{and} \quad \Delta\nu = 13.28 \pm 0.29\,\mu\text{Hz} \quad . \tag{61}$$

We randomly select a month-long segment of Kepler data, initialize our celerite model using a grid search in the parameter space, and then use emcee (Foreman-Mackey et al. 2013) to sample the joint posterior density for the full set of parameters. Figure 8 shows the marginalized density for $\nu_{\max}$ and $\Delta\nu$ compared to the results from the literature.

This model requires about a minute of computation time to find the maximum likelihood parameters and then it takes about an hour to run the MCMC sampling to convergence on a dual-CPU laptop. This is more computationally intensive than traditional methods of measuring asteroseismic oscillations but is is also orders of magnitude cheaper than the same analysis using another GP solver. An in-depth discussion of the benefits of rigorous probabilistic inference of asteroseismic parameters in the time domain is beyond the scope of this paper but we hope to revisit this opportunity in the future.

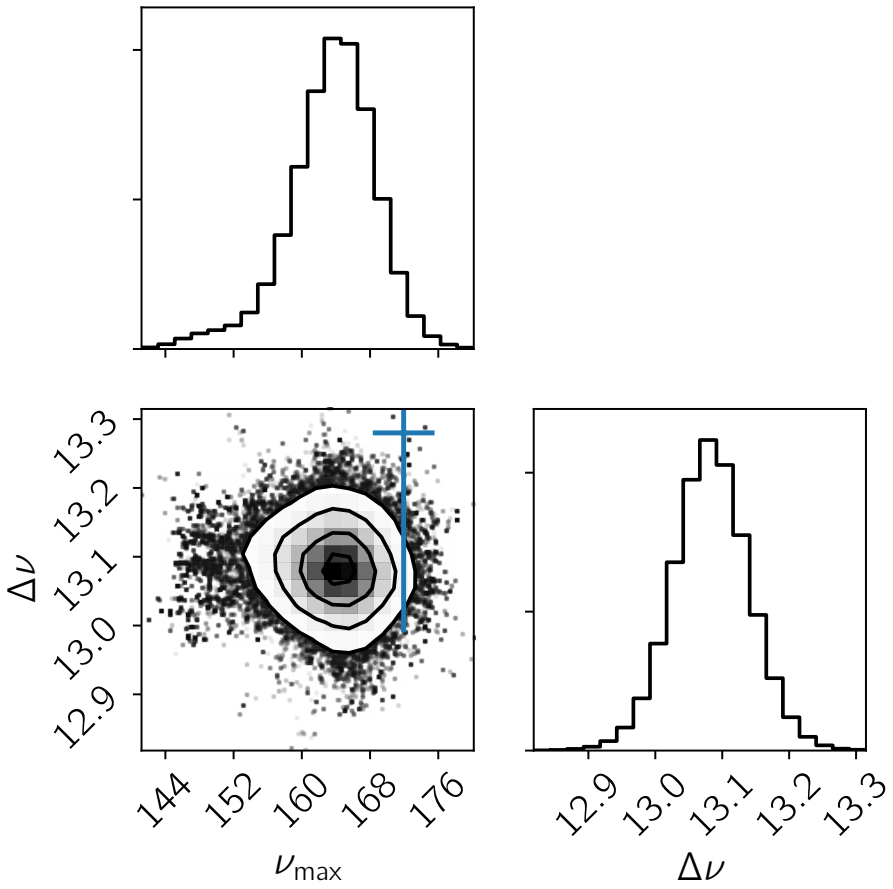

Figure 8. The probabilistic constraints on ν_{max} and $\Delta\nu$ from the inference shown in Figure 9 compared to the published value (error bar) based on several years of Kepler observations (Pinsonneault et al. 2014).

7.2. *Stellar rotation*

Another source of variability that can be measured from time series measurements of stars is rotation. The inhomogeneous surface of the star (spots, plage, *etc.*) imprints itself as quasiperiodic variations in photometric or spectroscopic observations (Dumusque et al. 2014). It has been demonstrated that for light curves with nearly uniform sampling, the empirical autocorrelation function provides a reliable estimate of the rotation period of a star (McQuillan et al. 2013, 2014; Aigrain et al. 2015) and that a GP model with a quasiperiodic covariance function can be used to make probabilistic measurements even with sparsely sampled data (R. Angus, *et al.* in prep.). The covariance function used for this type of analysis has the form

$$k(\tau) = A\,\exp\left(-\frac{\tau^2}{2\,\ell^2} - \Gamma\,\sin^2\left(\frac{\pi\,\tau}{P}\right)\right) \tag{62}$$

where P is the period of the oscillation. The key difference between this function and other quasiperiodic kernels is that it is positive everywhere. We can construct a

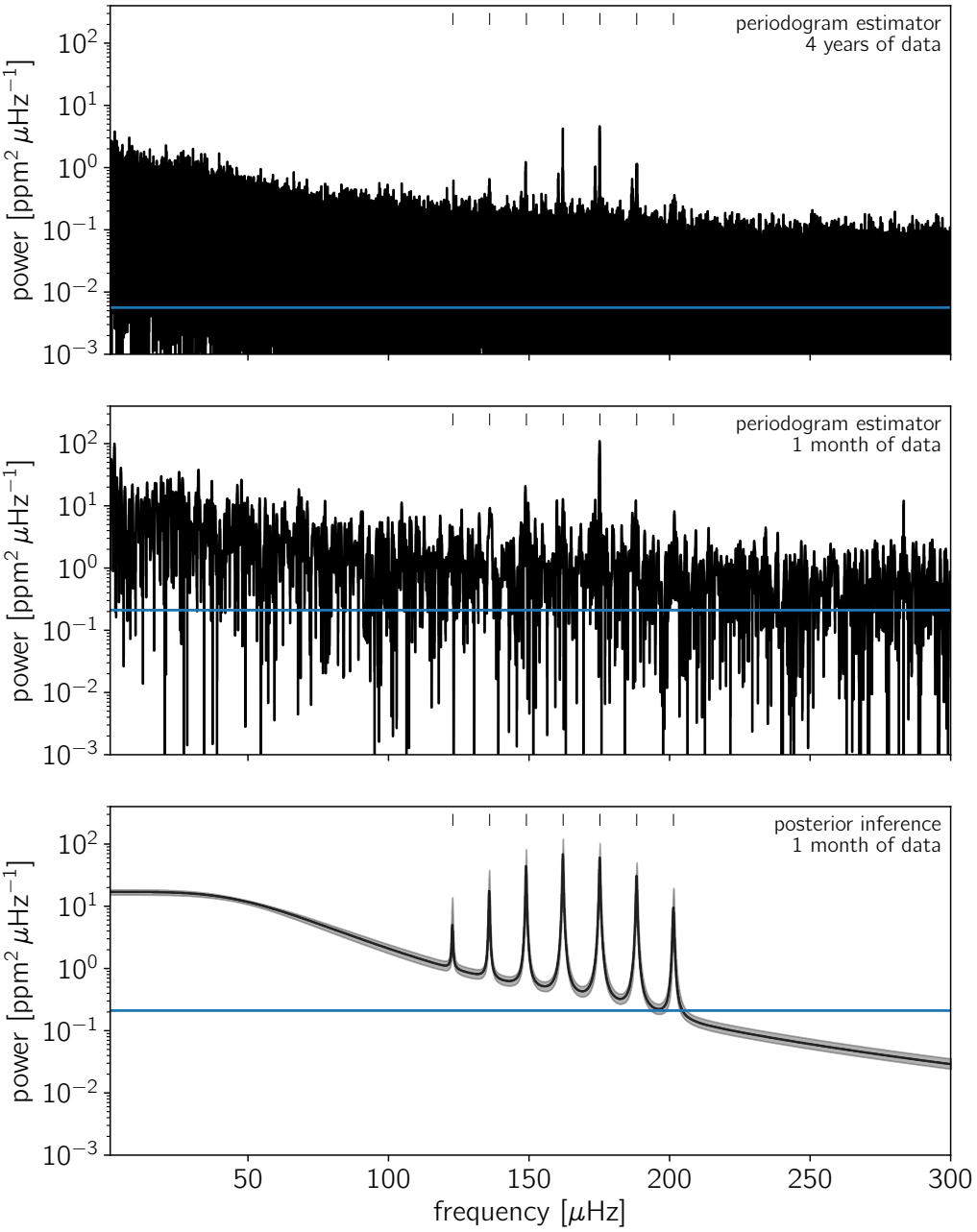

Figure 9. A comparison between the Lomb-Scargle estimator of the PSD and the posterior inference of the PSD as a mixture of stochastically-driven simple harmonic oscillators. (top) The periodogram of the Kepler light curve for KIC 11615890 computed on the full four year baseline of the mission. (middle) The same periodogram computed using about a month of data. (bottom) The power spectrum inferred using the mixture of SHOs model described in the text and only one month of Kepler data. The black line shows the median of posterior PSD and the gray contours show the 68% credible region.

simple celerite covariance function with similar properties as follows

$$k(\tau) = \frac{a}{2+b}\, e^{-c\,\tau} \left[\cos\left(\frac{2\,\pi\,\tau}{P}\right) + (1+b)\right] \tag{63}$$

for $a > 0$, $b > 0$, and $c > 0$. The covariance function in Equation (63) cannot exactly reproduce Equation (62) but, since Equation (62) is only an effective model, Equation (63) can be used as a drop-in replacement for a substantial gain in computational efficiency.

To demonstrate this method, we fit a celerite model with a kernel given by Equation (63) to a Kepler light curve for the star KIC 1430163. This star has a published rotation period of 3.88 ± 0.58, measured using traditional periodogram and autocorrelation function approaches applied to Kepler data from Quarters 0–16 (Mathur et al. 2014). We used emcee to sample the joint posterior probability density for the four parameters in Equation (63), conditioned on two quarters of Kepler data, and find a constraint on the rotation period of $P = 3.80^{+0.15}_{-0.15}$. This is in good agreement with the literature value. Figure 10 shows a subset of the data used in this example and the posterior inferences of the PSD and autocorrelation function of the process. Figure 11 shows the marginalized posterior distribution for the rotation period.

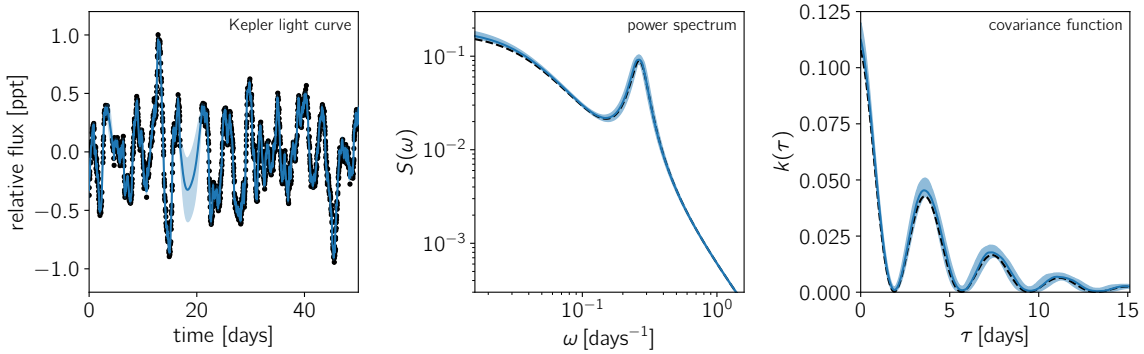

Figure 10. Inferred constraints on a quasiperiodic GP model using the covariance function in Equation (63) and two quarters of Kepler data. (left) The Kepler data (black points) and the maximum likelihood model prediction (blue curve) for a 50 day subset of the data used. The solid blue line shows the predictive mean and the blue contours show the predictive standard deviation. (center) Inferred constraints on the model PSD. The dashed line shows the maximum likelihood PSD, the blue solid line shows the median of posterior PSD, and the blue contours show the 68% credible region. (right) Inferred constraints on the model covariance function. The dashed line shows the maximum likelihood model, the blue solid line shows the median of posterior, and the blue contours show the 68% credible region.

7.3. *Exoplanet transit fitting*

In this example, we inject the signal of a simulated exoplanet transit into a real Kepler light curve and then demonstrate that we can recover the true physical parameters of the exoplanet while modeling the stellar variability using celerite. This example is

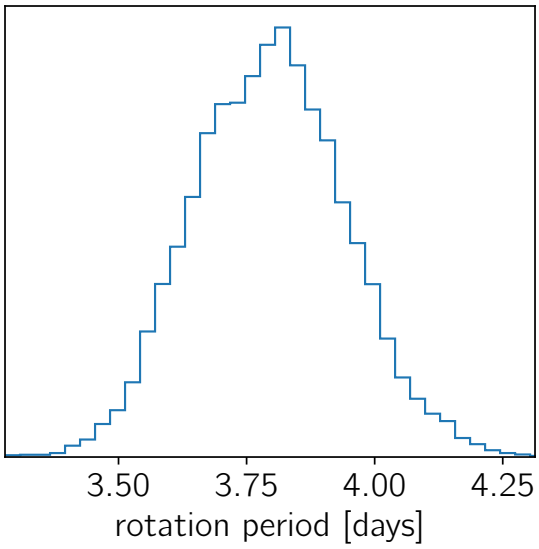

Figure 11. The posterior constraint on the rotation period of KIC 1430163 using the dataset and model from Figure 10. The period is the parameter P in Equation (63) and this figure shows the posterior distribution marginalized over all other nuisance parameters in Equation (63). This is consistent with the published rotation period made using the full Kepler baseline (Mathur et al. 2014).

different from all the previous examples because in this case, we are uninterested in the inferred parameters of the covariance model. Instead, we're interested in inferring constraints on the parameters of the mean model. In Equation (3) these parameters are called $\boldsymbol{\theta}$ and in this example, the mean function $\mu_{\boldsymbol{\theta}}(t)$ is a limb-darkened transit light curve (Mandel & Agol 2002) parameterized by a period P, a transit duration T, a phase or epoch t_0, an impact parameter b, the radius of the planet in units of the stellar radius $R_P/R_\star$, and several parameters describing the limb-darkening profile of the star (Claret & Bloemen 2011). As in the previous example, we model the stellar variability using a GP model with a kernel given by Equation (63).

We take a month-long segment of the Kepler light curve for KIC 1430163 – the target from the previous example – and multiply it by a simulated transit model with known parameters. The top panel of Figure 12 shows the data including the simulated transit. The bottom panel shows the same data with the maximum likelihood prediction for the stellar variation subtracted. To find this model, we maximized the likelihood function in Equation (3) with respect to the transit parameters $\boldsymbol{\theta}$ and the variability parameters $\boldsymbol{\alpha}$ simultaneously. Figure 13 shows the marginalized posterior constraints on the physical properties of the planet compared to the true values. This procedure produces estimates of the planet parameters that are consistent with the true values using celerite as an effective model for the stellar variability.

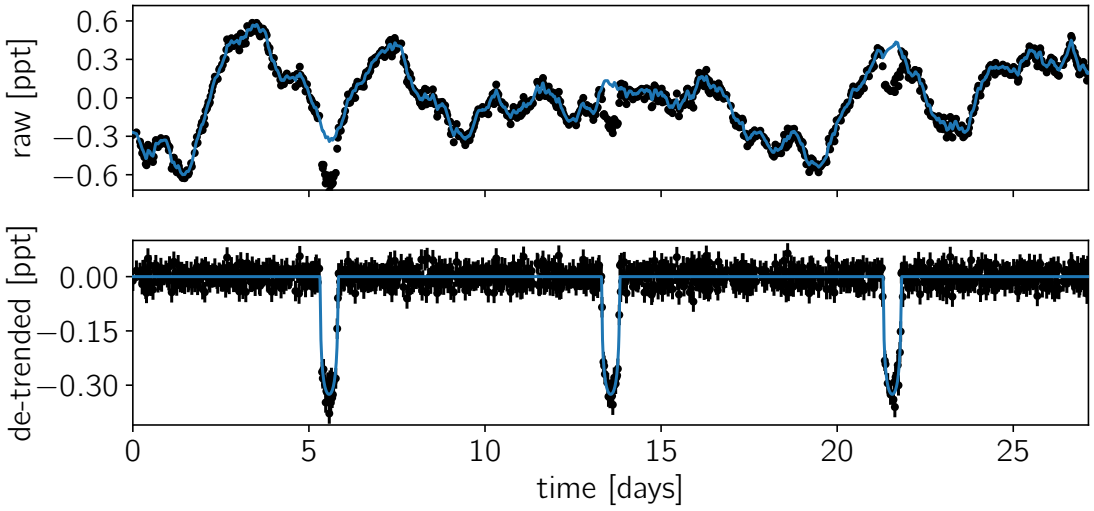

Figure 12. *(top)* A month-long segment of Kepler light curve for KIC 1430163 with a synthetic transit model injected (black points) and the maximum likelihood model for the stellar variability (blue line). *(bottom)* The maximum likelihood "de-trending" of the data in the top panel. In this panel, the maximum likelihood model for the stellar variability has been subtracted to leave only the transits. The de-trended light curve is shown by black error bars and the maximum likelihood transit model is shown as a blue line.

8. COMPARISONS TO OTHER METHODS

There are many other methods of scaling GP models to large datasets and in this section we draw comparisons between celerite and other popular methods. Scalable GP methods tend to fall into two categories: approximate and restrictive. celerite falls into the latter category because, while the method is exact, it requires a specific choice of stationary kernel function and it can only be used in one-dimension.

As discussed previously, the celerite method is based upon a method developed by Rybicki & Press twenty years ago (Rybicki & Press 1995). In the case where the kernel function has a single term of the form Equation (5), it is interesting to compare the performance of celerite and the Rybicki & Press method. Their method does not require solving the extended matrix because the matrix inverse can be calculated analytically but the method is made slightly more complicated when white noise is included. We implement their full method and find empirically that it is more computationally efficient by a factor of ≈ 2 for large $N \gg 1000$ but we note that the requirement of a single, real, exponential kernel is extremely restrictive and this model cannot describe quasi-periodic variability.

Another popular method uses the fact that, in the limit of evenly-spaced data and homoscedastic uncertainties, the covariance matrix is "Toeplitz" (for example Dillon et al. 2013). There are exact methods for solving Toeplitz matrix equations that scale as $\mathcal{O}(N \log N)$ and methods for computing determinants exactly in $\mathcal{O}(N^2)$ or approximately in $\mathcal{O}(N \log N)$ (Wilson 2014). The Toeplitz method is, in some ways, more flexible than celerite because it can be used with any stationary kernel but it

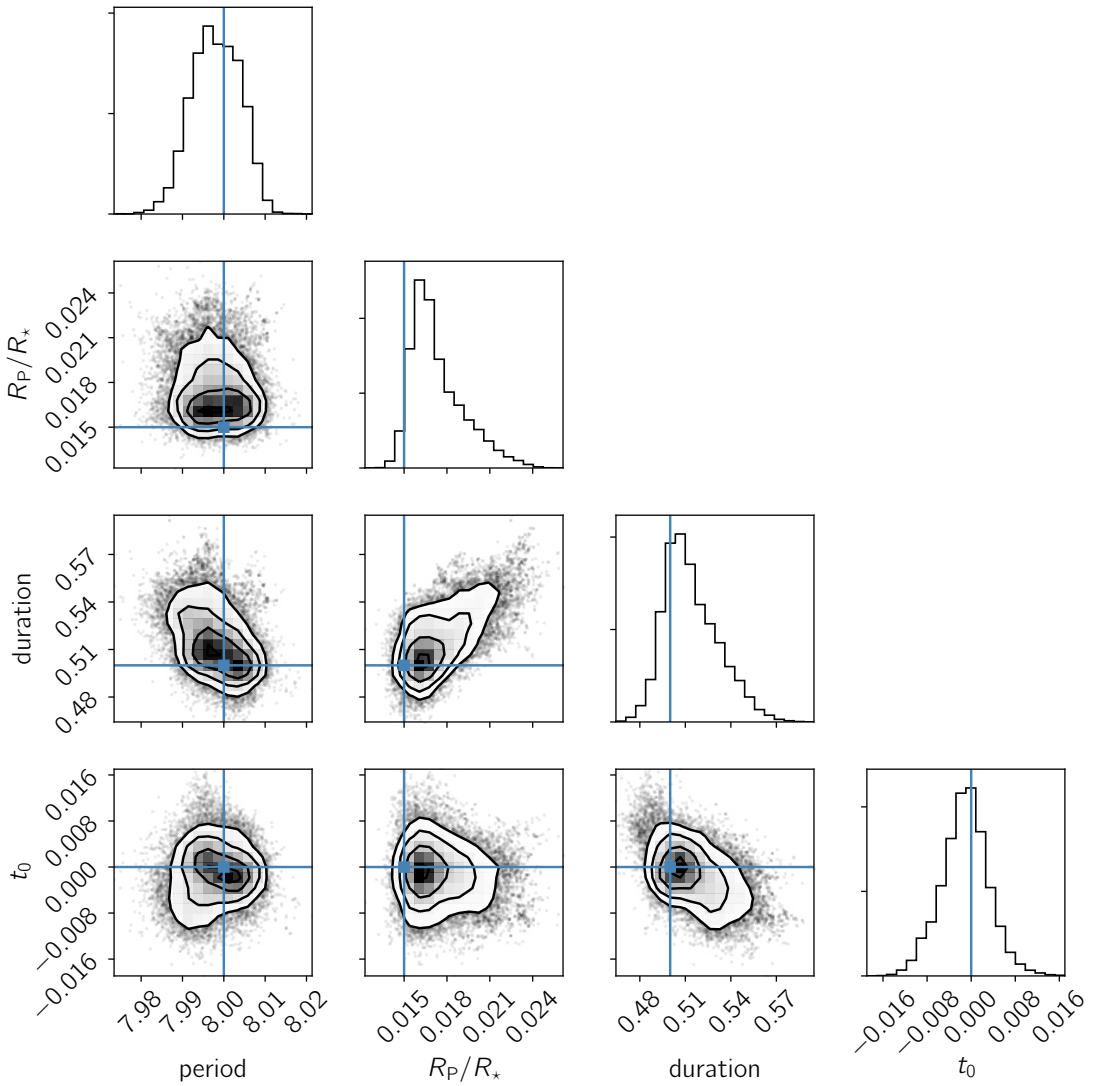

Figure 13. The marginalized posterior constraints on the physical parameters of the planet transit in the light curve shown in the top panel of Figure 12. The two-dimensional contours show the 0.5-, 1-, 1.5, and 2-sigma credible regions in the marginalized planes and the histograms along the diagonal show the marginalized posterior for each parameter. The true values used in the simulation are indicated by blue lines. For each parameter, the inference is consistent with the true value.

requires uniformly spaced data and the scaling is worse than celerite so it, in general, is less efficient when applied to large datasets.

Carter & Winn (2009) improved the scaling of Toeplitz methods by introducing a wavelet-based method for computing a GP likelihood with $\mathcal{O}(N)$ scaling. This method has been widely applied in the context of exoplanet transit characterization but it requires evenly spaced observations and the power spectrum of the process must have the form $S(\omega) \propto \omega^{-1}$ to gain the computational advantage. This wavelet method has been demonstrated to improve parameter estimation for transiting exoplanets (Carter & Winn 2009) but these strict requirements make this method applicable for only a limited set of use cases.

The continuous autoregressive moving average (CARMA) models introduced into the astrophysics literature by Kelly et al. (2014) share many features with celerite. Like celerite, the likelihood function for a CARMA model conditioned on a sorted one-dimensional dataset can be solved in $\mathcal{O}(N)$ using a recursive method. The kernel function for CARMA(J, K) model is (Kelly et al. 2014)

$$k_{\rm CARMA}(\tau) = \sum_{j=1}^{J} A_j \, \exp\left(r_j \, \tau\right) \tag{64}$$

where

$$A_j = \sigma^2 \, \frac{\left[\sum_{k=0}^{K} \beta_k \left(r_j\right)^k\right] \, \left[\sum_{k=0}^{K} \beta_k \left(-r_j\right)^k\right]}{-2\,\mathrm{Re}(r_j) \, \prod_{k=1,\,k\neq j}^{J} (r_k - r_j)(r_k{}^* + r_j)} \tag{65}$$

and σ, $\{r_j\}_{j=1}^{J}$ and $\{\beta_k\}_{k=1}^{K}$ are parameters of the model. Comparing Equation (64) to Equation (7), we can see that every CARMA model corresponds to an equivalent celerite model and the parameters a_j, b_j, c_j, and d_j can be easily computed analytically. The inverse statement is not true however, for any $J > 1$. This means celerite could be used to compute any CARMA model but using CARMA to compute the likelihood of a celerite model would require solving Equation (65) for a given set of $\{A_j\}_{j=1}^{J}$ numerically. This is an interesting distinction because, while the computational scaling of CARMA models is also $\mathcal{O}(N\,J^2)$, CARMA solvers are, in practice, somewhat faster and the memory requirements are smaller. However, as discussed in Section 5, celerite models have a physical interpretation as a mixture of stochastically-driven damped harmonic oscillators. This enables, for example, the use of celerite as a model for asteroseismic oscillations. A similar example using CARMA would necessarily be qualitative.

Another GP method that has been used extensively in astronomy is the hierarchical off-diagonal low rank (HODLR, Ambikasaran et al. 2016) solver. This method exploits the fact that many commonly used kernel functions produce "smooth" matrices to approximately compute the GP likelihood with the scaling $\mathcal{O}(N\,\log^2 N)$. This method has the advantage that, unlike celerite, it can be used with any kernel function but, in practice, the cost can still prove to be prohibitively high for multi-dimensional

inputs. The proportionality constant in the $N \log^2 N$ scaling of the HODLR method is a function of the specific kernel and we find – using the george software package (Foreman-Mackey et al. 2014; Ambikasaran et al. 2016) – that this scales approximately linearly with J but with substantial overhead for small models when compared to celerite. This means that the HODLR solver can *approximately* evaluate celerite models more efficiently than celerite for large models $J \gg 10$ and small datasets where N is smaller than a few thousand.

Many other approximate methods for scaling GP inference exist (see, for example, Wilson et al. 2015, and references therein) and we make no attempt to make our discussion exhaustive. The key takeaway here is that celerite provides an *exact* method for GP inference for a specific but flexible class of one-dimensional kernel functions. The closest cousin is the CARMA method with similar strengths and limitations but, since there is a simpler relationship between the parameters of a celerite model and its behavior, celerite can be more easily integrated with physical or physically interpretable models.

9. SUMMARY

Gaussian Process models have been fruitfully applied to many problems in astronomical data analysis but the fact that the computational cost scales as the cube of the number of data points has limited their use to relatively small datasets. With the linear scaling of celerite we envision the application of Gaussian processes may be expanded to much larger datasets. Despite the restrictive form of the celerite kernel, with a sufficient number of components it is flexible enough to describe a wide range of astrophysical variability. In fact, the relation of the celerite kernel to the damped, stochastically-driven harmonic oscillator matches simple models of astrophysical variability, and makes the parameterization interpretable in terms of resonant frequency, amplitude, and quality factor. A drawback of this method is its quadratic scaling with the number of terms J but, in many cases, small values of J are sufficient.

Our background is in studying transiting exoplanets, a field which has only recently begun to adopt full covariance matrices in analyzing the noise in stellar light curves when detecting or characterizing transiting planets (for example, Carter & Winn 2009; Gibson et al. 2012; Barclay et al. 2015; Evans et al. 2015; Aigrain et al. 2016; Foreman-Mackey et al. 2016; Grunblatt et al. 2016; Luger et al. 2016). All of these analyses have been limited to small datasets or restrictive kernel choices. celerite weakens these requirements by providing a scalable method for computing the likelihood and a physical motivation for the choice of kernel. celerite can be used to accurately model stellar oscillations using the relation to the mixture of stochastically-driven, damped simple harmonic oscillators. As higher signal-to-noise observations of transiting exoplanet systems are obtained, the effects of stellar variability will more dramatically impact the correct inference of planetary transit parameters. We expect that celerite

will be important for transit detection (Pope et al. 2016; Foreman-Mackey et al. 2016), transit timing (Agol et al. 2005; Holman 2005), transit spectroscopy (Brown 2001), Doppler beaming (Loeb & Gaudi 2003; Zucker et al. 2007), tidal distortion (Zucker et al. 2007), phase functions (Knutson et al. 2007; Zucker et al. 2007), and more.

Beyond these applications to model stellar variability, the method is generally applicable to other one-dimensional GP models. Accreting black holes show time series which may be modeled using a GP (Kelly et al. 2014); indeed, this was the motivation for the original technique developed by Rybicki & Press (Rybicki & Press 1992, 1995). This approach may be broadly used for characterizing quasar variability (MacLeod et al. 2010), measuring time lags with reverberation mapping (Zu et al. 2011; Pancoast et al. 2014), modeling time delays in multiply-imaged gravitationally-lensed systems (Press & Rybicki 1998), characterizing quasi-periodic variability in a high-energy source (McAllister et al. 2016), or classification of variable objects (Zinn et al. 2016). We expect that there are also applications beyond astronomy.

The celerite formalism can also be used for power spectrum estimation and quantification of its uncertainties. In principle, a large number of celerite terms could be used to perform non-parametric probabilistic inference of the power spectrum despite unevenly-spaced data with heteroscedastic noise (for example, Wilson & Prescott Adams 2013; Kelly et al. 2014). This type of analysis will be limited by the quadratic scaling of celerite with the number of terms J but this limits existing methods as well (CARMA models, Kelly et al. 2014). In general, we encourage the use of physically motivated models for parameter estimation instead of qualitative modeling of the power spectrum itself.

There are many data analysis problems where celerite will not be applicable. In particular, the restriction to one-dimensional problems is significant. There are many examples of multidimensional GP modeling the astrophysics literature (recent examples from the field of exoplanet characterization include Haywood et al. 2014; Rajpaul et al. 2015; Aigrain et al. 2016), where celerite cannot be used to accelerate any of these analyses. It is plausible that an extension could be derived to tackle some multidimensional problems with the right structure – simultaneous parallel time series, for example – and we hope to revisit this possibility in future work.

Alongside this paper, we have released a well-tested and documented open source software package that implements the method and all of the examples discussed in these pages. This software is available at `https://github.com/dfm/celerite`[7] and it is made available under the MIT license.

It is a pleasure to thank Megan Bedell, Will Farr, Sam Grunblatt, David W. Hogg, Dan Huber, and Jake VanderPlas for helpful discussions informing the ideas and code presented here.

[7] This version of the paper was generated with git commit `e40c771` (2017-03-24).

This work was performed in part under contract with the Jet Propulsion Laboratory (JPL) funded by NASA through the Sagan Fellowship Program executed by the NASA Exoplanet Science Institute.

EA acknowledges support from NASA grants NNX13AF20G, NNX13A124G, NNX13AF62G, from National Science Foundation (NSF) grant AST-1615315, and from NASA Astrobiology Institute's Virtual Planetary Laboratory, supported by NASA under cooperative agreement NNH05ZDA001C.

This research made use of the NASA Astrophysics Data System and the NASA Exoplanet Archive. The Exoplanet Archive is operated by the California Institute of Technology, under contract with NASA under the Exoplanet Exploration Program.

This paper includes data collected by the Kepler mission. Funding for the Kepler mission is provided by the NASA Science Mission directorate. We are grateful to the entire Kepler team, past and present.

These data 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 is provided by the NASA Office of Space Science via grant NNX13AC07G and by other grants and contracts.

Facility: Kepler

Software: corner.py (Foreman-Mackey 2016), Eigen (Guennebaud et al. 2010), emcee (Foreman-Mackey et al. 2013), george (Ambikasaran et al. 2016), LAPACK (Anderson et al. 1999), matplotlib (Hunter et al. 2007), numpy (Van Der Walt et al. 2011), transit (Foreman-Mackey & Morton 2016), scipy (Jones et al. 2001).

APPENDIX

A. ENSURING POSITIVE DEFINITENESS

For a GP kernel to be valid, it must produce a positive definite covariance matrix for all input coordinates. For stationary kernels, this is equivalent – by Bochner's theorem (see Section 4.2.1 in Rasmussen & Williams 2006) – to requiring that the kernel be the Fourier transform of a positive finite measure. This means that the power spectrum of a positive definite kernel must be positive for all frequencies. This result is also intuitive because, since the power spectrum of a process is defined as expected the squared amplitude of the Fourier transform of the time series, it must be non-negative.

Using Equation (9), we find that for a single celerite term, this requirement is met when

$$\frac{(a_j\,c_j + b_j\,d_j)\,({c_j}^2 + {d_j}^2) + (a_j\,c_j - b_j\,d_j)\,\omega^2}{\omega^4 + 2\,({c_j}^2 - {d_j}^2)\,\omega^2 + ({c_j}^2 + {d_j}^2)^2} > 0 \quad . \tag{A1}$$

The denominator is positive for all $c_j \neq 0$ and it can be shown that, when $c_j = 0$, Equation (A1) is satisfied for all $\omega \neq d_j$ where the power is identically zero. Therefore,

when $c_j \neq 0$, we require that the numerator is positive for all ω. This requirement can also be written as

$$a_j\, c_j > -b_j\, d_j \tag{A2}$$

$$a_j\, c_j > b_j\, d_j \quad . \tag{A3}$$

Furthermore, we can see that a_j must be positive since $k(0) = a_j$ should should be positive and, similarly, by requiring the covariance to be finite at infinite lag, we obtain the constraint $c_j \geq 0$. Combining these results, we find the constraint

$$|b_j\, d_j| < a_j\, c_j \quad . \tag{A4}$$

In the case of J celerite terms, we can check for negative values of the PSD by solving for the roots of the power spectrum; if there are any real, positive roots, then the power-spectrum goes negative (or zero), and thus does not represent a valid kernel. We rewrite the power spectrum, Equation 9), abbreviating with $z = \omega^2$:

$$S(\omega) = \sum_{j=1}^{J} \frac{q_j\, z + r_j}{z^2 + s_j\, z + t_j} = 0 \tag{A5}$$

where

$$q_j = a_j\, c_j - b_j\, d_j \tag{A6}$$

$$r_j = (d_j^2 + c_j^2)(b_j\, d_j + a_j\, c_j) \tag{A7}$$

$$s_j = 2(c_j^2 - d_j^2) \tag{A8}$$

$$t_j = (c_j^2 + d_j^2)^2. \tag{A9}$$

The denominators of each term are positive, so we can multiply through by $\prod_j (z^2 + s_j z + t_j)$ to find

$$Q_0(z) = \sum_{j=1}^{J} (q_j z + r_j) \prod_{k \neq j} \left(z^2 + s_k z + t_k\right) = 0 \quad , \tag{A10}$$

which is a polynomial with order $2\,(J-1)+1$. With $J = 2$, this yields a cubic equation whose roots can be obtained exactly.

For arbitrary J, a procedure based upon Sturm's theorem (Dörrie 1965) allows one to determine whether there are any real roots within the range $(0, \infty]$. We first construct $Q_0(z)$ and its derivative $Q_1(z) = Q_0{}'(z)$, and then loop from $k = 2$ to $k = 2\,(J-1)+1$, computing

$$Q_k(z) = -\mathrm{rem}(Q_{k-2},\, Q_{k-1}) \tag{A11}$$

where the function $\mathrm{rem}(p,\, q)$ is the remainder polynomial after dividing $p(z)$ by $q(z)$.

We evaluate the coefficients of each of the polynomial in the series by evaluating $f_0 = \{Q_0(0),\, \ldots,\, Q_{2\,(J-1)+1}(0)\}$ to give us the signs of these polynomials evaluated at $z = 0$. Likewise, we evaluate the coefficients of the largest order term in each polynomial that gives the sign of the polynomial as $z \to \infty$, f_∞. With the sequence

of coefficients f_0 and f_∞, we then determine how many times the sign changes in each of these, where $\sigma(0)$ is the number of sign changes at $z = 0$, and $\sigma(\infty)$ is the number of sign changes at $z \to \infty$. The total number of real roots in the range $(0, \infty]$ is given by $N_+ = \sigma(0) - \sigma(\infty)$.

We have checked that this procedure works for a wide range of parameters, and we find that it robustly matches the number of positive real roots which we evaluated numerically. The advantage of this procedure is that it does not require computing the roots, but only carrying out algebraic manipulation of polynomials to determine the number of positive real roots. If a non-zero real root is found, the likelihood may be set to zero.

REFERENCES

Agol, E., Steffen, J., Sari, R., & Clarkson, W. 2005, Monthly Notices of the Royal Astronomical Society, 359, 567

Aigrain, S., Parviainen, H., & Pope, B. J. S. 2016, Monthly Notices of the Royal Astronomical Society, 706

Aigrain, S., Llama, J., Ceillier, T., et al. 2015, Monthly Notices of the Royal Astronomical Society, 450, 3211

Ambikasaran, S. 2015, Numer. Linear Algebra Appl., 22, 1102

Ambikasaran, S., Foreman-Mackey, D., Greengard, L., Hogg, D. W., & O'Neil, M. 2016, IEEE Transactions on Pattern Analysis and Machine Intelligence, 38, 252

Anderson, E., Bai, Z., Bischof, C., et al. 1999, LAPACK Users' Guide, 3rd edn. (Philadelphia, PA: Society for Industrial and Applied Mathematics)

Anderson, E. R., Duvall, Jr., T. L., & Jefferies, S. M. 1990, ApJ, 364, 699

Barclay, T., Endl, M., Huber, D., et al. 2015, ApJ, 800, 46

Bond, J. R., Crittenden, R. G., Jaffe, A. H., & Knox, L. 1999, Comput. Sci. Eng., Vol. 1, No. 2, p. 21 - 35, 1, 21

Bond, J. R., & Efstathiou, G. 1987, Monthly Notices of the Royal Astronomical Society, 226, 655

Brewer, B. J., & Stello, D. 2009, Monthly Notices of the Royal Astronomical Society, 395, 2226

Brown, T. M. 2001, The Astrophysical Journal, 553, 1006

Campante, T. L., Schofield, M., Kuszlewicz, J. S., et al. 2016, The Astrophysical Journal, 830, 138

Carter, J. A., & Winn, J. N. 2009, The Astrophysical Journal, 704, 51

Chaplin, W. J., Kjeldsen, H., Christensen-Dalsgaard, J., et al. 2011, Science, 332, 213

Chaplin, W. J., Basu, S., Huber, D., et al. 2013, The Astrophysical Journal Supplement Series, 210, 1

Claret, A., & Bloemen, S. 2011, Astronomy & Astrophysics, 529, A75

Corsaro, E., & Ridder, J. D. 2014, Astronomy & Astrophysics, 571, A71

Czekala, I., Mandel, K. S., Andrews, S. M., et al. 2017, ArXiv e-prints, arXiv:1702.05652

Demmel, J. W., Eisenstat, S. C., Gilbert, J. R., Li, X. S., & Liu, J. W. H. 1999, SIAM J. Matrix Analysis and Applications, 20, 720

Dillon, J. S., Liu, A., & Tegmark, M. 2013, PhRvD, 87, 043005

Dörrie, H. 1965, 100 Great Problems of Elementary Mathematics: Their History and Solution, Dover Books on Mathematics Series, §24, (Dover Publications), 112–116

Dumusque, X., Boisse, I., & Santos, N. C. 2014, The Astrophysical Journal, 796, 132

Evans, T. M., Aigrain, S., Gibson, N., et al. 2015, Monthly Notices of the Royal Astronomical Society, 451, 680

Foreman-Mackey, D. 2016, The Journal of Open Source Software, 24, doi:10.21105/joss.00024

Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, PASP, 125, 306

Foreman-Mackey, D., Hoyer, S., Bernhard, J., & Angus, R. 2014, george: George (v0.2.0), , , doi:10.5281/zenodo.11989

Foreman-Mackey, D., & Morton, T. 2016, dfm/transit: v0.3.0, , , doi:10.5281/zenodo.159478

Foreman-Mackey, D., Morton, T. D., Hogg, D. W., Agol, E., & Schölkopf, B. 2016, AJ, 152, 206

Gibson, N. P., Aigrain, S., Roberts, S., et al. 2012, MNRAS, 419, 2683

Gilliland, R. L., Brown, T. M., Christensen-Dalsgaard, J., et al. 2010, Publications of the Astronomical Society of the Pacific, 122, 131
Gould, A., Huber, D., Penny, M., & Stello, D. 2015, Journal of The Korean Astronomical Society, 48, 93
Grunblatt, S. K., Huber, D., Gaidos, E. J., et al. 2016, AJ, 152, 185
Guennebaud, G., Jacob, B., et al. 2010, Eigen v3, http://eigen.tuxfamily.org, ,
Harvey, J. 1985, in ESA Special Publication, Vol. 235, Future Missions in Solar, Heliospheric & Space Plasma Physics, ed. E. Rolfe & B. Battrick
Haywood, R. D., Cameron, A. C., Queloz, D., et al. 2014, Monthly Notices of the Royal Astronomical Society, 443, 2517
Holman, M. J. 2005, Science, 307, 1288
Huber, D., Stello, D., Bedding, T. R., et al. 2009, Communications in Asteroseismology, 160, 74
Huber, D., Bedding, T. R., Stello, D., et al. 2011, ApJ, 743, 143
Hunter, J. D., et al. 2007, Computing in science and engineering, 9, 90
Jones, E., Oliphant, T., Peterson, P., et al. 2001, SciPy: Open source scientific tools for Python, ,
Kallinger, T., De Ridder, J., Hekker, S., et al. 2014, A&A, 570, A41
Kelly, B. C., Becker, A. C., Sobolewska, M., Siemiginowska, A., & Uttley, P. 2014, ApJ, 788, 33
Knutson, H. A., Charbonneau, D., Allen, L. E., et al. 2007, Nature, 447, 183
Loeb, A., & Gaudi, B. S. 2003, The Astrophysical Journal, 588, L117
Luger, R., Agol, E., Kruse, E., et al. 2016, AJ, 152, 100
MacLeod, C. L., Ivezić, Ž., Kochanek, C. S., et al. 2010, ApJ, 721, 1014
Mandel, K., & Agol, E. 2002, The Astrophysical Journal, 580, L171
Mathur, S., García, R. A., Ballot, J., et al. 2014, A&A, 562, A124
McAllister, M. J., Littlefair, S. P., Dhillon, V. S., et al. 2016, Monthly Notices of the Royal Astronomical Society, 464, 1353
McQuillan, A., Aigrain, S., & Mazeh, T. 2013, MNRAS, 432, 1203
McQuillan, A., Mazeh, T., & Aigrain, S. 2014, ApJS, 211, 24
Michel, E., Samadi, R., Baudin, F., et al. 2009, A&A, 495, 979
Nocedal, J., & Wright, S. J. 2006, Numerical Optimization (Springer)
Pancoast, A., Brewer, B. J., Treu, T., et al. 2014, Monthly Notices of the Royal Astronomical Society, 445, 3073
Pinsonneault, M. H., Elsworth, Y., Epstein, C., et al. 2014, ApJS, 215, 19
Pope, B. J. S., Parviainen, H., & Aigrain, S. 2016, MNRAS, 461, 3399
Press, W. H., & Rybicki, G. B. 1998, ApJ, 507, 108
Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 1992, Numerical recipes in FORTRAN. The art of scientific computing (Cambridge University Press)
Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 2007, Numerical Recipes 3rd Edition: The Art of Scientific Computing (Cambridge University Press)
Rajpaul, V., Aigrain, S., Osborne, M. A., Reece, S., & Roberts, S. 2015, Monthly Notices of the Royal Astronomical Society, 452, 2269
Rasmussen, C. E., & Williams, K. I. 2006, Gaussian Processes for Machine Learning (MIT Press)
Rauer, H., Catala, C., Aerts, C., et al. 2014, Experimental Astronomy, 38, 249
Ricker, G. R., Winn, J. N., Vanderspek, R., et al. 2014, in Space Telescopes and Instrumentation 2014: Optical, Infrared, and Millimeter Wave, ed. J. M. Oschmann, M. Clampin, G. G. Fazio, & H. A. MacEwen (SPIE-Intl Soc Optical Eng)
Rybicki, G. B., & Press, W. H. 1992, ApJ, 398, 169
—. 1995, Physical Review Letters, 74, 1060
Stello, D., Huber, D., Bedding, T. R., et al. 2013, The Astrophysical Journal, 765, L41
Uttley, P., McHardy, I. M., & Vaughan, S. 2005, Monthly Notices of the Royal Astronomical Society, 359, 345
Van Der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, Computing in Science & Engineering, 13, 22
Wandelt, B. D., & Hansen, F. K. 2003, PhRvD, 67, 023001
Wilson, A. G. 2014, PhD thesis, University of Cambridge
Wilson, A. G., Dann, C., & Nickisch, H. 2015, arXiv preprint arXiv:1511.01870, http://arxiv.org/abs/1511.01870
Wilson, A. G., & Prescott Adams, R. 2013, ArXiv e-prints, arXiv:1302.4245
Zinn, J. C., Kochanek, C. S., Kozłowski, S., et al. 2016, ArXiv e-prints, arXiv:1612.04834
Zu, Y., Kochanek, C. S., & Peterson, B. M. 2011, ApJ, 735, 80
Zucker, S., Mazeh, T., & Alexander, T. 2007, The Astrophysical Journal, 670, 1326