

A Bayesian method for inference of fundamental stellar parameters

November 18, 2013

Ian Czekala¹ et al.

iczekala@cfa.harvard.edu

1. Introduction

The fundamental parameters of stars, such as mass, luminosity, effective photospheric temperature, surface gravity, and metallicity are of supreme interest in astrophysics, for a wide variety of reasons. Exoplanet properties are found relative to the properties of the host star, and have important implications for models of planet formation. The properties are also important to the study of disks, for the incident radiation field sets photoevaporation, influences disk dispersal. Many methods exist for determination of stellar parameters: SPC (Buchhave et al. 2012), MOOG, SME (Valenti & Piskunov 1996), and these are robust for FKG stars, where there is a well-defined continuum and the stars are “normal.” However, for many young T-Tauri stars, the spectrum is contaminated by accretion, veiling, and fractional star spot coverage, which can be a nuisance and cause systematic errors if traditional techniques are used.

2. The model

To summarize our model, we use Bayes rule to design a posterior probability function

$$\overbrace{p(\vec{\theta}|\vec{D})}^{\text{posterior}} \propto \underbrace{p(\vec{D}|\vec{\theta})}_{\text{likelihood}} \overbrace{p(\vec{\theta})}^{\text{prior}} \quad (1)$$

where the parameter vector is comprised of both stellar parameters and calibration parameters $\vec{\theta} = \{\vec{\theta}_\star, \vec{\theta}_N\}$. Stellar parameters include $\vec{\theta}_\star = \{T_{\text{eff}}, \log(g), [\text{Fe}/\text{H}], v \sin i, v_z, A_v, R_\star^2/d^2\}$ while the data vector $\vec{D} = f_D(\lambda)$ represents a high resolution spectrum. For demonstration purposes in this paper, we use a 51-order echelle spectrum from the *TRES* spectrograph at

¹Harvard-Smithsonian Center for Astrophysics, 60 Garden Street MS 10, Cambridge, MA 02138

a resolution of $R = 48,000$ (6.8 km/s), which spans the full optical range (3800 – 9000Å), though our technique is applicable to any high resolution optical/near-IR spectrum.

Although per-pixel photon counting errors are Poisson, we can safely assume that we have enough counts that we are in the Gaussian limit, although see the discussion in §2.6. Then our likelihood function becomes a pixel-by-pixel χ^2 comparison between the data spectrum f_D , and the model spectrum f_M , summed over the wavelength axis.

$$p(\vec{D}|\vec{\theta}) = \mathcal{L} \propto \exp\left(-\frac{\chi^2}{2}\right) \quad (2)$$

$$\chi^2 = \sum_{\lambda} \left[\frac{f_D(\lambda) - k(\lambda|\vec{\theta}_N)f_M(\lambda|\vec{\theta}_*)}{\sigma(\lambda)} \right]^2 \quad (3)$$

With flat priors on $\vec{\theta}_*$ and $\vec{\theta}_N$ (an assumption we will later relax) the logarithm of our posterior probability function is then

$$\ln[p(\vec{\theta}|\vec{D})] \propto -\frac{1}{2} \sum_{\lambda} \left[\frac{f_D(\lambda) - k(\lambda|\vec{\theta}_N)f_M(\lambda|\vec{\theta}_*)}{\sigma(\lambda)} \right]^2 \quad (4)$$

To generate a model spectrum given a set of parameters $f_M(\vec{\theta})$, a model spectrum is created by linearly interpolating in $T_{\text{eff}}, \log(g), [\text{Fe}/\text{H}]$ space between grid points of instrumentally broadened, oversampled PHOENIX model stellar spectra, rotationally broadened using Fourier techniques, Doppler shifted, extinction corrected, and downsampled using spline interpolation to the exact pixels of the TRES spectrum.

$\sigma(\lambda)$ is the Poisson photon counting errors. For some data reduction pipelines, a per-pixel “sigma spectrum” that accounts for spectral extraction errors due to night sky line contamination or low signal to noise is available, and should be used instead. For more discussion of the noise sources beyond statistical Poisson errors, see §2.6.

The prefactor $k(\lambda|\vec{\theta}_N)$ is a calibration function that aims to account for errors in the blaze-correction or flux-calibration by multiplying the model by a polynomial. The parameters describing this function are the coefficients of the polynomial. This approach of accounting for large-scale calibration errors in the spectrum is more justified than the traditional approach of normalizing to the stellar continuum. From a flux-calibration experiment where we observed spectrophotometric standards BD+28 and BD+25 multiple times, we calibrated the dispersion in (Figure?) flux-calibration function, or IRAF’s “sensitivity function.” These tests revealed that the largest source of error in flux-calibration was actually in the overall magnitude of individual order, this could be due to slit losses,

poor correction due to parallactic angle, or scattered light within the instrument. These corrections are smaller than 10%. But more importantly, the corrections in the higher order terms (ie, the shape of the bandpass calibration for each order) are small ($\lesssim 3\%$). As described in §2.2, we can set priors on these coefficients. This will be the optimal mode of operation for our model. However, there exists a wealth of archival data that is un-fluxcalibrated, since its primary purpose was for radial velocity determination of FGK stars, to which it is easy to normalize to a continuum. In this case, the scatter amongst the calibration can be as large as a factor of 2x. This prior information on the calibration parameters motivates our choice of functional parameterization for the nuisance parameters in §2.1.

2.1. Nuisance parameters

In our case, $\vec{\theta}_N = \{c_0, c_1, \dots, c_N\}$ are a set of 4 Chebyshev polynomial coefficients for each order. This amounts to a “re-fluxing” of the model following the techniques of Eisenstein et al. (2006). We apply the re-fluxing to the model rather than attempt to re-flux the data since this operation would also need to re-scale the noise, making χ^2 a complicated function of k .

In order to understand the effects of the $k(\lambda|\vec{\theta}_N)$ term, we now consider its effects on just one order of the 51-order echelle spectrum. The first four Chebyshev polynomials are

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_2(x) &= 2x^2 - 1 \\ T_3(x) &= 4x^3 - 3x \end{aligned}$$

In our implementation, we map the full pixel range of a specific order $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ to $x \in [-1, 1]$. By mapping pixel number in a given order, rather than wavelength, we are insensitive to wavelength shifts of the model. Then, for a given set of Chebyshev coefficients, $\vec{\theta}_N = \{c_0, c_1, c_2, \dots, c_N\}$, we have

$$k(\lambda|\vec{\theta}_N) = c_0 T_0(\lambda) [1 + c_1 T_1(\lambda) + \dots + c_N T_N(\lambda)] \quad (5)$$

$$k(\lambda|\vec{\theta}_N) = c_0 \left[1 + \sum_{i=1}^N c_i T_i(\lambda) \right] \quad (6)$$

When applied to scale the model flux f_M , this becomes

$$k(\lambda|\vec{\theta}_N)f_M = \left[1 + \sum_{i=1}^N c_i T_i(\lambda)\right] (c_0 f_M) \quad (7)$$

$$k(\lambda|\vec{\theta}_N)f_M = c_0 f_M + c_1 T_1(c_0 f_M) + \dots + c_N T_N(c_0 f_M) \quad (8)$$

This functional form of k allows c_0 to scale by large amounts while the higher order terms (denoted by c_n) are perturbations on top of the *scaled* model ($c_0 f_M$). However, see §A for an alternative model when perturbations are small. We can rewrite Equation 4 as

$$\ln[p(\vec{\theta}|\vec{D})] \propto -\frac{1}{2} \sum_{\lambda} \left[\frac{f_D(\lambda) - [1 + \sum_{i=1}^N c_i T_i(\lambda)] c_0 f_M(\lambda|\vec{\theta}_*)}{\sigma(\lambda)} \right]^2 \quad (9)$$

To simplify the following discussion, we consider a single wavelength λ_i and then generalize this to a sum over λ later. If we expand out the square in Equation 9, then for a given λ_i we have

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2\sigma^2} \left[f_D^2 - 2f_D c_0 f_M \left[1 + \sum_{i=1}^N c_i T_i\right] + c_0^2 f_M^2 \left[1 + \sum_{i=1}^N c_i T_i\right]^2 \right] \quad (10)$$

where σ , f_D , f_M , and $T_{n,m}$ are all evaluated at λ_i . We can further expand this to

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2\sigma^2} \left[f_D^2 - 2f_D c_0 f_M \left[1 + \sum_{i=1}^N c_i T_i\right] + c_0^2 f_M^2 \left[1 + 2 \sum_{i=1}^N c_i T_i + \sum_{i=1}^N \sum_{j=1}^N c_i T_i c_j T_j\right] \right] \quad (11)$$

To simplify this math, we can rewrite the Chebyshev coefficients and polynomials as column vectors \vec{c} and $\vec{T}(\lambda)$, respectively

$$\vec{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix} \quad \vec{T}(\lambda) = \begin{bmatrix} T_1(\lambda) \\ T_2(\lambda) \\ \vdots \\ T_N(\lambda) \end{bmatrix} \quad (12)$$

then

$$k(\lambda|\vec{\theta}_N) = c_0 \left[1 + \sum_{i=0}^N c_i T_i\right] = c_0 \left[1 + \vec{T}^T \cdot \vec{c}\right] \quad (13)$$

If we let

$$\mathbf{W}(\lambda) = \vec{T} \cdot \vec{T}^\top = \begin{bmatrix} T_1 T_1 & T_1 T_2 & \dots & T_1 T_N \\ T_2 T_1 & T_2 T_2 & \dots & T_2 T_N \\ \vdots & \vdots & \ddots & \vdots \\ T_N T_1 & T_N T_2 & \dots & T_N T_N \end{bmatrix} \quad (14)$$

then we have

$$\sum_{i=1}^N \sum_{j=1}^N c_i T_i c_j T_j = \vec{c}^\top \cdot \mathbf{W} \cdot \vec{c} \quad (15)$$

We can rewrite Equation 11 quadratically in \vec{c} as

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2\sigma^2} \left[f_D^2 - 2f_D c_0 f_M \left[1 + \sum_{i=1}^N c_i T_i \right] + c_0^2 f_M^2 \left[1 + 2 \sum_{i=1}^N c_i T_i + \sum_{i=1}^N \sum_{j=1}^N c_i T_i c_j T_j \right] \right] \quad (16)$$

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2} \frac{c_0^2 f_M^2}{\sigma^2} \sum_{i=1}^N \sum_{j=1}^N c_i T_i c_j T_j + \left(\frac{-c_0^2 f_M^2 + c_0 f_D f_M}{\sigma^2} \right) \sum_{i=1}^N c_i T_i - \left(\frac{c_0^2 f_M^2 - 2c_0 f_D f_M + f_D^2}{2\sigma^2} \right) \quad (17)$$

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2} \frac{c_0^2 f_M^2}{\sigma^2} \vec{c}^\top \mathbf{W} \vec{c} + \left(\frac{-c_0^2 f_M^2 + c_0 f_D f_M}{\sigma^2} \right) \vec{T}^\top \vec{c} - \left(\frac{c_0^2 f_M^2 - 2c_0 f_D f_M + f_D^2}{2\sigma^2} \right) \quad (18)$$

Because matrix multiplication is associative, we can sum $\mathbf{W}(\lambda)$ and $\vec{T}^\top(\lambda)$ across all λ , and define

$$\mathbf{A} = c_0^2 \sum_{\lambda} \frac{f_M^2(\lambda)}{\sigma^2(\lambda)} \mathbf{W}(\lambda) \quad (19)$$

$$\vec{B} = \sum_{\lambda} \frac{-f_M^2(\lambda) c_0^2 + f_D(\lambda) f_M(\lambda) c_0}{\sigma^2(\lambda)} \vec{T}(\lambda) \quad (20)$$

$$g = -\frac{1}{2} \sum_{\lambda} \frac{c_0^2 f_M^2(\lambda) - 2c_0 f_D(\lambda) f_M(\lambda) + f_D^2(\lambda)}{\sigma^2(\lambda)} \quad (21)$$

Where \mathbf{A} , \vec{B} , and g are each a function of $\vec{\theta}_*$, $\vec{\theta}_N$, and \vec{D} . Now we can rewrite Equation 9 as

$$\ln[p(\vec{\theta}_*, \vec{\theta}_N | \vec{D})] \propto -\frac{1}{2} \vec{c}^\top \mathbf{A} \vec{c} + \vec{B}^\top \vec{c} + g \quad (22)$$

$$p(\vec{\theta} | \vec{D}) = p(\vec{\theta}_*, \vec{\theta}_N | \vec{D}) \propto \exp \left(-\frac{1}{2} \vec{c}^\top \mathbf{A} \vec{c} + \vec{B}^\top \vec{c} + g \right) \quad (23)$$

Here we have the full posterior probability distribution defined in Equation 1, assuming flat priors. This posterior is a function of both the stellar parameters $\vec{\theta}_\star$ and the nuisance parameters $\vec{\theta}_N$. Generally, we are most interested in the stellar parameters after they have been marginalized over the nuisance parameters. This marginalization encapsulates the uncertainty due to the inference on the nuisance parameters. Because this probability function is a multi-dimensional Gaussian in the higher order Chebyshev coefficients $\vec{c} = \{c_1, c_2, \dots, c_N\}$, we can do this marginalization analytically, and then sample from the marginalized distribution.

$$p(\vec{\theta}_\star, c_0 | \vec{D}) = \int p(\vec{\theta}_\star, c_0, \vec{c} | \vec{D}) d\vec{c} \quad (24)$$

The analytic multi-dimensional Gaussian integral with linear term (Stoof et al. 2009) yields

$$p(\vec{\theta}_\star, c_0 | \vec{D}) \propto \int \exp \left(-\frac{1}{2} \vec{c}^\top \mathbf{A} \vec{c} + \vec{B}^\top \vec{c} + g \right) d\vec{c} \quad (25)$$

$$p(\vec{\theta}_\star, c_0 | \vec{D}) \propto \sqrt{\frac{(2\pi)^N}{\det|\mathbf{A}|}} \exp \left(\frac{1}{2} \vec{B}^\top \mathbf{A}^{-1} \vec{B} + g \right) \quad (26)$$

Where N is the number of Chebyshev coefficients marginalized over, in our case $N = 3$. By doing this marginalization we have reduced the dimensionality of our posterior space from 200+ parameters (~ 7 stellar parameters plus 3 Chebyshev coefficients for each of 51 echelle orders) to ~ 60 (~ 7 stellar parameters and one c_0 for each of the 51 echelle orders). This enables a dramatic speedup in the MCMC sampling. For flux-calibrated spectra where the variation in c_0 is expected to be small (less than XX%), we can approximate k in a different manner which allows marginalization over the c_0 coefficients as well, reducing the dimensionality of the space to only the ~ 7 stellar parameters. This approach is detailed in §A.

2.2. Including Gaussian priors on nuisance parameters

Using this formalism, it is possible to incorporate prior knowledge about the degree of necessary “re-fluxing” through priors on the nuisance parameters. From our flux-calibration experiment, we determined that the overall scaling could range by 10%, while the higher order corrections were less than 3%. Because c_0 is a scale factor, it should be equally probable to scale up by a factor of two as it is to scale down by a factor of two. This means

we should be using a log-normal prior on c_0

$$p(c_0) = \frac{1}{\sqrt{2\pi}\sigma_{c_0}c_0} \exp\left(-\frac{(\ln c_0)^2}{2\sigma_{c_0}^2}\right) \quad (27)$$

$$\ln p(c_0) = \ln\left(\frac{1}{\sqrt{2\pi}\sigma_{c_0}c_0}\right) - \frac{(\ln c_0)^2}{2\sigma_{c_0}^2} \quad (28)$$

centered on $c_0 = 1$. Because the higher order polynomial terms are perturbations on the scaled model ($c_0 f$), we assume Gaussian priors centered about 0

$$p(c_n) = \frac{1}{\sqrt{2\pi}\sigma_{c_n}} \exp\left(-\frac{c_n^2}{2\sigma_{c_n}^2}\right) \quad (29)$$

which we can express in vector form as

$$p(\vec{c}) \propto \exp\left(-\frac{1}{2}(\vec{c} - \vec{\mu})^\top \mathbf{D}(\vec{c} - \vec{\mu})\right) \quad (30)$$

where

$$\mathbf{D} = \begin{bmatrix} \sigma_1^{-2} & 0 & \dots & 0 \\ 0 & \sigma_2^{-2} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & \sigma_N^{-2} \end{bmatrix} \quad (31)$$

and σ_i represents the width of the Gaussian prior on the i -th Chebyshev coefficient, in our case $\sigma_i = \sigma_{c_n} \approx 0.03$. $\vec{\mu}$ is the mean of the Gaussian prior, in nearly all cases we expect to set this to $\vec{\mu} = \{0, 0, \dots, 0\}$, unless we have prior knowledge that the flux-calibration is systematically warped in one direction. Now, we can write Equation 1 with priors as

$$p(\vec{\theta}_\star, \vec{\theta}_N | \vec{D}) = p(\vec{\theta}_\star, c_0, \vec{c} | \vec{D}) \propto p(\vec{D} | \vec{\theta}_\star, c_0, \vec{c}) p(\vec{c}) p(\vec{\theta}_\star) p(c_0) \quad (32)$$

$$p(\vec{\theta}_\star, \vec{\theta}_N | \vec{D}) \propto \exp\left(-\frac{1}{2}\vec{c}^\top \mathbf{A}\vec{c} + \vec{B}^\top(\vec{\theta}_\star)\vec{c} + g(\vec{\theta}_\star)\right) \exp\left(-\frac{1}{2}(\vec{c} - \vec{\mu})^\top \mathbf{D}(\vec{c} - \vec{\mu})\right) p(\vec{\theta}_\star) p(c_0) \quad (33)$$

we can expand and rearrange the argument of Equation 30 to a similar form that is quadratic in \vec{c}

$$-\frac{1}{2}(\vec{c} - \vec{\mu})^\top \mathbf{D}(\vec{c} - \vec{\mu}) = \frac{1}{2}\left(-\vec{c}^\top \mathbf{D}\vec{c} + \vec{c}^\top \mathbf{D}\vec{\mu} + \vec{\mu}^\top \mathbf{D}\vec{c} - \vec{\mu}^\top \mathbf{D}\vec{\mu}\right) \quad (34)$$

$$= -\frac{1}{2}\vec{c}^\top \mathbf{D}\vec{c} + (\mathbf{D}\vec{\mu})^\top \vec{c} - \frac{1}{2}\vec{\mu}^\top \mathbf{D}\vec{\mu} \quad (35)$$

then we can rewrite

$$\mathbf{A}' = \mathbf{A} + \mathbf{D} \quad (36)$$

$$\vec{B}' = \vec{B} + (\mathbf{D}\vec{\mu})^\top \quad (37)$$

$$g' = g - \frac{1}{2}\vec{\mu}^\top \mathbf{D}\vec{\mu} \quad (38)$$

in the case that $\vec{\mu} = \{0, 0, \dots, 0\}$, there is only a non-zero correction to \mathbf{A} . The full posterior probability distribution is then

$$p(\vec{\theta}_\star, \vec{\theta}_N | \vec{D}) = p(\vec{\theta}_\star, c_0, \vec{c}) \propto \exp\left(-\frac{1}{2}\vec{c}^\top \mathbf{A}' \vec{c} + \vec{B}'^\top \vec{c} + g'\right) p(\vec{\theta}_\star) p(c_0) p(\vec{c}) \quad (39)$$

This probability function appears in the code as `lnprob_lognormal` and has ~ 7 stellar dimensions and ~ 200 nuisance dimensions.

Using the previous multi-dimensional integral, we can marginalize over the \vec{c} including the Gaussian priors

$$p(\vec{\theta}_\star, c_0 | \vec{D}) \propto \sqrt{\frac{(2\pi)^N}{\det|\mathbf{A}'|}} \exp\left(\frac{1}{2}\vec{B}'^\top \mathbf{A}'^{-1} \vec{B}' + g'\right) p(\vec{\theta}_\star) p(c_0) \quad (40)$$

where N is the number of Chebyshev coefficients marginalized over, in our case $N = 3$. This probability function appears in the code as `lnprob_lognormal_marg` and has ~ 7 stellar dimensions and ~ 50 nuisance dimensions. If we increase the permissible degree of re-fluxing by increasing σ_i , this will have the effect of smearing out or widening the posterior on a given $\vec{\theta}_\star$.

2.3. Sampling a un-flux calibrated model

This invalidates the previous section on normalization. However, for corrections of less than XX%, this approximation is better than XX% accurate.

For the flux-calibration exercise, the overall spread in c_0 across all orders should not be more than $\sigma_{c_0} \leq 2$, or the one-sigma deviation of c_0 is between $0.125 \leq c_0 \leq 8.0$.

The plot in `plots/priortests` shows that the log-normal prior is the correct choice. Since we care most about the inference on F , we want to know that the choice on prior of c_0 is not biasing the final outcome, while a Gaussian prior might if σ_{c_0} was large. How to

quantify this error and create a dividing line? For 5% error? In peak? Do these plots need to be in the paper?

Compare distributions by doing a K-S test on the cumulative distributions between Gaussian and Log-normal?

2.4. Hierarchical modelling

Can have a c_1 , etc, that are marginalized over but we sample c_0 in a hierarchical manner which informs the overall flux-level.

2.5. Sampling with emcee

Citation to DFM (Foreman-Mackey et al. 2012), GW (Goodman & Weare 2010), and CosmoHAMMER (Akeret et al. 2013).

2.6. Noise, Masking, Residuals, and Alternative likelihood functions

In the future, we could modify our likelihood function to use a Student-t statistic instead of a Gaussian, such that it would be more robust in the presence of outlier pixels, however this does not seem necessary at the moment.

We must necessarily mask certain wavelength regions of the T Tauri spectra that are contaminated (one might also say “made more interesting”) by the astrophysical realities of accretion or stellar spots (such as the Balmer lines, which are in emission). We also accept the assumption that rotation or accretion onto the star does not fundamentally alter the structure of the model atmosphere that generates the model spectrum and that a spectral comparison of the “clean” regions of the spectrum is valid.

Section on how to choose masked regions, testing the PHOENIX grid to see where the derivative is highest and thus gives the best leverage on stellar parameters.

Section on interpolation errors.

3. Validation tests

Comparison to main sequence stars of HAT-P-9, and WASP-14, from the TRES archive (Torres et al. 2012).

Comparisons using HIRES data?

4. Processing TRES Data

We flux calibrate TRES data, but because the reduction pipeline does not produce a “sigma spectrum,” we instead use the Poisson errors on each pixel. By taking the raw, still-blazed spectrum, in units of “counts,” we can find the Signal-to-Noise ratio by,

$$S/N = \sqrt{\text{cts}} \quad (41)$$

then the Noise-to-Signal ratio is the inverse of this

$$N/S = \frac{1}{\sqrt{\text{cts}}} \quad (42)$$

then the errors on the flux-calibrated spectrum (in ergs/s/Å/cm²) is simply the flux spectrum multiplied by N/S.

IC would like to graciously acknowledge Gregory Green, Doug Finkbeiner, and Daniel Eisenstein for many fruitful discussions about statistics and spectroscopy. This research made use of Astropy, a community-developed core Python package for Astronomy (Astropy Collaboration et al. 2013).

A. Simplification for small c_0 perturbations

If the correction in c_0 is small, we can use an alternate formulation of k and put Gaussian priors on all nuisance parameters. This allows analytic marginalization over all nuisance parameters (including c_0), and reduces the dimensionality of the posterior to only the ~ 7 stellar parameters, $\vec{\theta}_*$.

$$k(\lambda|\vec{\theta}_N) = c_0 T_0(\lambda) + c_1 T_1(\lambda) + \dots + c_N T_N(\lambda) \quad (\text{A1})$$

$$k(\lambda|\vec{\theta}_N) = \sum_{i=0}^N c_i T_i(\lambda) \quad (\text{A2})$$

Following §2.1

$$\ln[p(\vec{\theta}|\vec{D})] \propto -\frac{1}{2} \sum_{\lambda} \left[\frac{f_D(\lambda) - [\sum_{i=0}^N c_i T_i(\lambda)] f_M(\lambda|\vec{\theta}_*)}{\sigma(\lambda)} \right]^2 \quad (\text{A3})$$

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2\sigma^2} [f_D^2 - 2f_D f_M \sum_n c_n T_n + f_M^2 \sum_n \sum_m c_n T_n c_m T_m] \quad (\text{A4})$$

Now the column vectors \vec{c} and $\vec{T}(\lambda)$ include c_0 as well

$$\vec{c} = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix} \quad \vec{T}(\lambda) = \begin{bmatrix} T_0(\lambda) \\ T_1(\lambda) \\ \vdots \\ T_N(\lambda) \end{bmatrix} \quad (\text{A5})$$

$$k(\lambda|\vec{\theta}_N) = \sum_{i=0}^N c_i T_i = \vec{T}^\top \cdot \vec{c} \quad (\text{A6})$$

$$\mathbf{W}(\lambda) = \vec{T} \cdot \vec{T}^\top = \begin{bmatrix} T_0 T_0 & T_0 T_1 & \dots & T_0 T_N \\ T_1 T_0 & T_1 T_1 & \dots & T_1 T_N \\ \vdots & \vdots & \ddots & \vdots \\ T_N T_0 & T_N T_1 & \dots & T_N T_N \end{bmatrix} \quad (\text{A7})$$

$$\sum_n \sum_m c_n T_n c_m T_m = \vec{c}^\top \cdot \mathbf{W} \cdot \vec{c} \quad (\text{A8})$$

$$\ln[p(\vec{\theta}|\lambda_i)] \propto -\frac{1}{2\sigma^2} [f_D^2 - 2f_D f_M \vec{T}^\top \vec{c} + f_M^2 \vec{c}^\top \mathbf{W} \vec{c}] \quad (\text{A9})$$

$$\mathbf{A} = \sum_{\lambda} \frac{f_M^2(\lambda)}{\sigma^2(\lambda)} \mathbf{W}(\lambda) \quad (\text{A10})$$

$$\vec{B}^\top = \sum_{\lambda} \frac{f_D(\lambda) f_M(\lambda)}{\sigma^2(\lambda)} \vec{T}^\top(\lambda) \quad (\text{A11})$$

$$g = -\frac{1}{2} \sum_{\lambda} \frac{f_D(\lambda)}{\sigma^2(\lambda)} \quad (\text{A12})$$

$$(\text{A13})$$

Where \mathbf{A} , \vec{B} , and g are each a function of $\vec{\theta}_\star$, $\vec{\theta}_N$, and \vec{D} . If we include Gaussian priors, then we have

$$\mathbf{A}' = \mathbf{A} + \mathbf{D} \quad (\text{A14})$$

$$\vec{B}' = \vec{B} + (\mathbf{D}\vec{\mu})^\top \quad (\text{A15})$$

$$g' = g - \frac{1}{2}\vec{\mu}^\top \mathbf{D}\vec{\mu} \quad (\text{A16})$$

where now \mathbf{D} and $\vec{\mu}$ include c_0 as well. The main difference from before is that now the prior on c_0 is now Gaussian instead of log-normal. This is acceptable as long as σ_{c_0} is small ($< XX\%$), otherwise this will bias the correction factor. This also means that we will usually have $\vec{\mu} = \{1, 0, \dots, 0\}$. The full posterior probability function becomes

$$p(\vec{\theta}_\star, \vec{\theta}_N | \vec{D}) = p(\vec{\theta}_\star, \vec{c}) \propto \exp\left(-\frac{1}{2}\vec{c}^\top \mathbf{A}' \vec{c} + \vec{B}'^\top \vec{c} + g'\right) p(\vec{\theta}_\star) p(\vec{c}) \quad (\text{A17})$$

Using the previous multi-dimensional integral, we can marginalize over all \vec{c} including the Gaussian priors

$$p(\vec{\theta}_\star | \vec{D}) \propto \sqrt{\frac{(2\pi)^N}{\det|\mathbf{A}'|}} \exp\left(\frac{1}{2}\vec{B}'^\top \mathbf{A}'^{-1} \vec{B}' + g'\right) p(\vec{\theta}_\star) \quad (\text{A18})$$

To gain insight, we can re-express this new version of k in the original form as

$$k(\lambda | \vec{\theta}_N) = c_0 T_0 + c_1 T_1 + \dots + c_N T_N \quad (\text{A19})$$

where $c_1 = c_0 c_1' T_0$ and $c_N = c_0 c_N' T_0$. Thus, if there needs to be a large perturbation in c_0 , then there must also be a large perturbation in c_n . The prior on c_1 is now a product of a log-normal prior on c_0 and a Gaussian prior on c_1' , which is not analytically tractable. Therefore, in the case of large perturbations, it is better to use Equation and use a log-normal prior on c_0 . When we use the original framework, we are saying that perturbations in c_0 are small enough that we can approximate the log-normal prior on c_0 as Gaussian and that the priors on c_n can also be correctly approximated by a Gaussian. From tests, the original framework gives acceptable accuracy (using $\sigma_{c_n} = 0.05$) if the corrections in c_0 are $\sigma_{c_0} \leq 0.35$, or the one-sigma deviation of c_0 is between $0.7 \leq c_0 \leq 1.42$. Beyond this, one should switch to this framework, for example to correct for un-flux calibrated data.

REFERENCES

- Akeret, J., Seehars, S., Amara, A., Refregier, A., & Csillaghy, A. 2013, CosmoHammer: Cosmological parameter estimation with the MCMC Hammer, 1212.1721, astro-physics Source Code Library
- Astropy Collaboration et al. 2013, A&A, 558, A33
- Buchhave, L. A. et al. 2012, Nature, 486, 375
- Eisenstein, D. J. et al. 2006, ApJS, 167, 40, arXiv:astro-ph/0606700
- Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2012, ArXiv e-prints, 1202.3665
- Goodman, J., & Weare, J. 2010, Communications in Applied Mathematics and Computational Science, 5, 65
- Stoof, H. T., Gubbels, K. B., & Dickerscheid, D. B. 2009, Ultracold quantum fields (Springer)
- Torres, G., Fischer, D. A., Sozzetti, A., Buchhave, L. A., Winn, J. N., Holman, M. J., & Carter, J. A. 2012, ApJ, 757, 161, 1208.1268
- Valenti, J. A., & Piskunov, N. 1996, A&AS, 118, 595