

Methods for the spectroscopic inference of fundamental stellar parameters, continued.

February 21, 2014

Ian Czekala¹ et al.

`iczekala@cfa.harvard.edu`

1. Correlated noise

- systematic errors don't necessarily invalidate chi-squared if they can be treated as a noise with a non-trivial covariance matrix; they just make chi-squared not an independent sum of residuals weighted by inverse variances but because a dot-product through a non-trivial inverse covariance tensor.
- "more drastic than adding in noise" – what about "adding in highly covariant and correlated noise"? That takes no more equipment (everything is still Gaussian), but can handle far worse situations.
- Where you mention χ_Q^2 , you are treating the pixels as independent. But they aren't if there is model mismatch. That is, there are correlations among the residuals.

Thank you for pointing this out. It spurred a bit of math on my part and now I've arrived at an interesting toy model that I would consider a nice proof-of-concept, although it has some major scaling problems. Maybe you could point to some good matrices routines.

Describe the χ^2 for a non-trivial covariance matrix.

Describe how I determined the covariance function for a Gaussian.

Show a plot of how using this covariance function one could generate noise that follows any range of values (a la ipython notebook).

Show the Bayesian model that uses the covariance function parameters, sparse matrices, and χ^2 . Show how it is able to correctly identify the parameters b , m , and the proper Gaussian noise values.

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

Ask about how to speed up this computation, or at least make some approximations. Note how similar it is to the “add a line” approach, though much more expensive. How to efficiently store matrix inversions? Is it necessary to actually compute the inverse?

Point out that different covariance functions could be useful for different spectral problems, such as M dwarf continuum (show reference, or lifted plot).

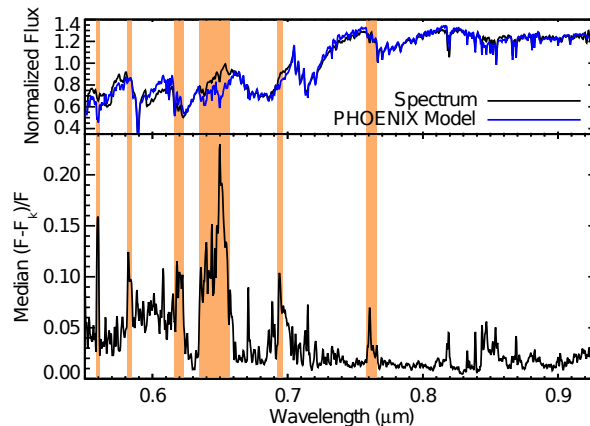


Fig. 1.— From Mann et al. (2013)

good on the stuff about there being important information where the models and data disagree.

in your four classes of line behavior, there is none which is dominated by continuum issues? Or is there? That is, continuum fitting can cause problems, no? I would think that yes, there should be areas where continuum fitting can be a problem. However, in the stars I’m looking at (over the narrow echelle range), the Chebyshev formalism essentially assumes that any continuum mis-match is actually a flux-calibration (instrumental) issue, and accounts for it. This means that if we were fitting hotter stars with few lines, any information contained in the continuum shape would be lost. However, this approach does prevent against introducing bad features *into* the continuum.

Depending on the size of the continuum problem, yes, this may also be a problem. I didn’t include it in the error classification because I didn’t think it was as dire a problem as the others, but when it comes to M spectra, this may actually be a problem.

trigger went off with “making these regions useless for determining...”. Useless? I

shouldn't have been so hasty!

Section 5: You might even be able to learn changes to the models. This is the holy grail. I see where this is going. In the original approach I could plot the sparsity matrix, and the areas of larger variance/off-diagonal terms would pin-point where the model and data diverge. Also, the form of the covariance function.

2. Downsampling

section 2: why "linearly interpolate"? It would be good to see more specificity here; there are lots of ways to do this smooth and downsample slightly wrong.

For the linear interpolation from the grid, basically it is because I wanted implement my lnprob easily and to be memory efficient. It would be better to use a more sophisticated interpolator. For example, Husser et al. (2013) use a spline interpolator (I think), where they pre-compute and store the derivatives of the spectra. This is currently low priority because we expect the actual error to be due to these systematic errors in the model.

I went through a few of the wrong approaches. Allow me to elaborate on the current method. Keep Fourier in touch. Spectra from the PHOENIX grid are given in units of flux density. We flux calibrate our spectra, so that they are also in flux density.

3. Chebyshev

Can you really marginalize analytically over the Chebyshev polynomials? And doesn't that make a huge, correlated uncertainty tensor? And what priors do you use? I'm uncertain what you mean about the huge, correlated uncertainty tensor. Which parameters are in this tensor?

In re +/- 0.01 dex: You have tons of pixels so of course you get amazing constraints! It would be surprising if you didn't. But do you get such amazing constraints if you marginalize analytically over thousands of Chebyshevs? Show plot of run output, things still work fine, and in fact you get the same answer sampled vs. marginalized?

When you do the drop and interpolate over testing, can you build an empirical interpolation error model and then use it? I am planning on building a model, but it has been lower priority at the moment.

REFERENCES

- Husser, T.-O., Wende-von Berg, S., Dreizler, S., Homeier, D., Reiners, A., Barman, T., & Hauschildt, P. H. 2013, *A&A*, 553, A6, 1303.5632
- Mann, A. W., Gaidos, E., & Ansdell, M. 2013, *ApJ*, 779, 188, 1311.0003