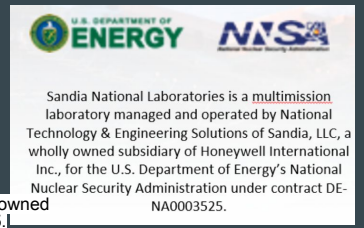
This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2021-9589C

AGD: Autonomous Gaussian Decomposition for Spectral Line Identification

P. Cho

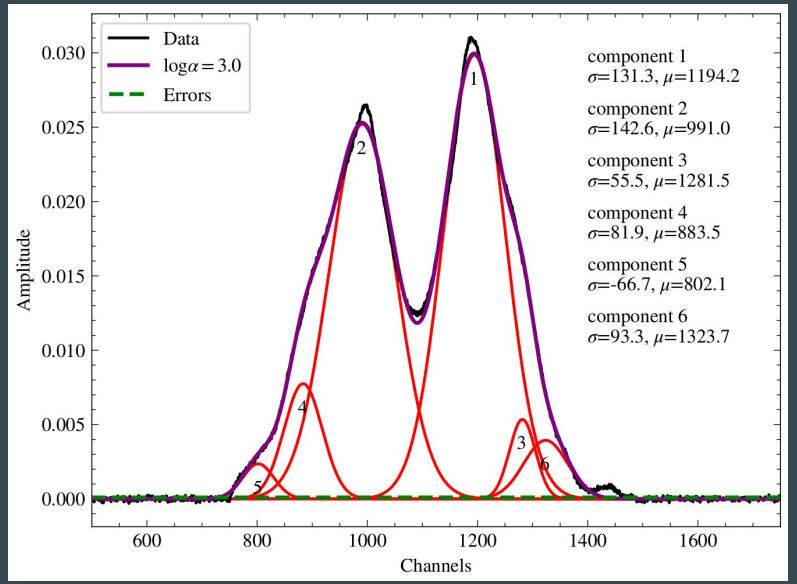
08.09.21 | 2021 ZFSW Astrophysics Breakout



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

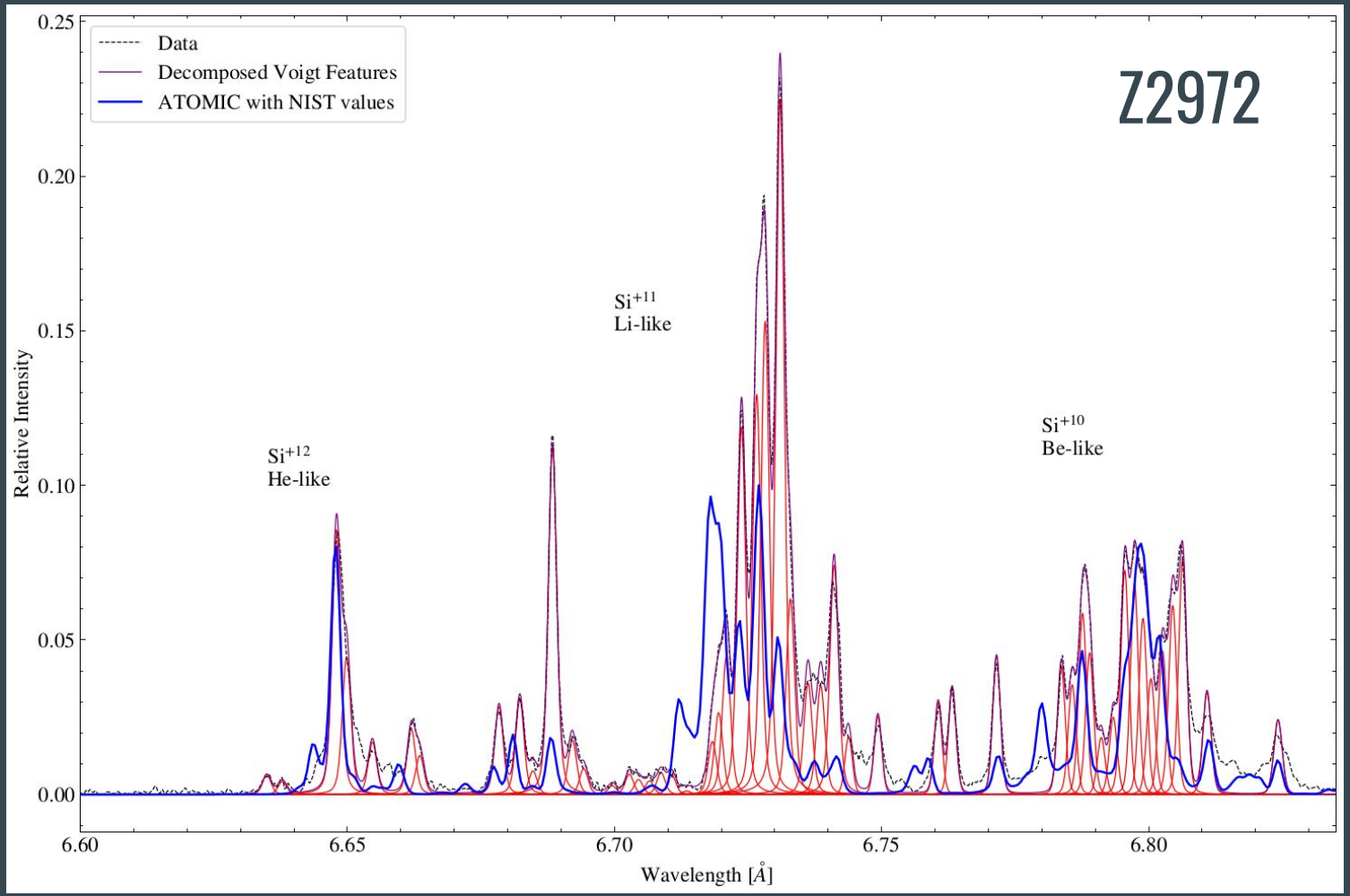
Motivation

- High resolution Si emission spectra obtained on Z gives us the opportunity to identify many lines/transitions never before observed in laboratory setting
- Line identifications → precise wavelength determinations
  - Beneficial for both laboratory and astrophysical atomic databases
- Identifying the transitions responsible for the features requires fitting their component distributions
  - Broadening dominated by instrumental sources → broadening profile is primarily Gaussian
  - Fitting can be done using Gaussian components
  - Various degrees of blending and overlap makes this non-trivial.
- We can use Autonomous Gaussian Decomposition (AGD) implemented in a Python package called GAUSSPY to find optimal fits for constituent components of spectral features.



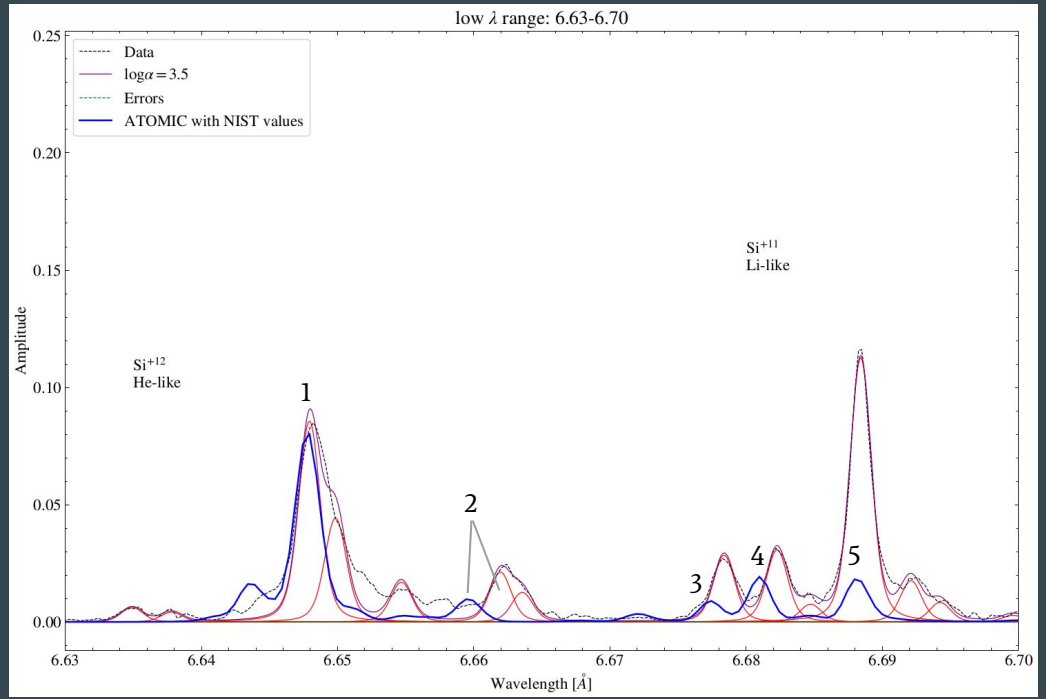
AGD Algorithm - Overview

- Automatically decomposes spectra into individual components
- Uses derivative spectroscopy and machine learning → optimized guesses for the number of components and their parameters (σ_L, σ_G, amplitude)
- Autonomous and more computationally efficient
  - Doesn't require a set of manually determined initial guesses
- I modified the source code to use frequency dependent Voigt profiles
  - Gaussian: Source size, detector, doppler
  - Crystal (Lorentzian)


Data
Decomposed Voigt Features
ATOMIC with NIST values
Z2972
Si+12
He-like
Si+11
Li-like
Si+10
Be-like
Relative Intensity
0.25
0.20
0.15
0.10
0.05
0.00
6.60
6.65
6.70
6.75
6.80
Wavelength [Å]

Z2972 low λ range: 6.63-6.70 Å

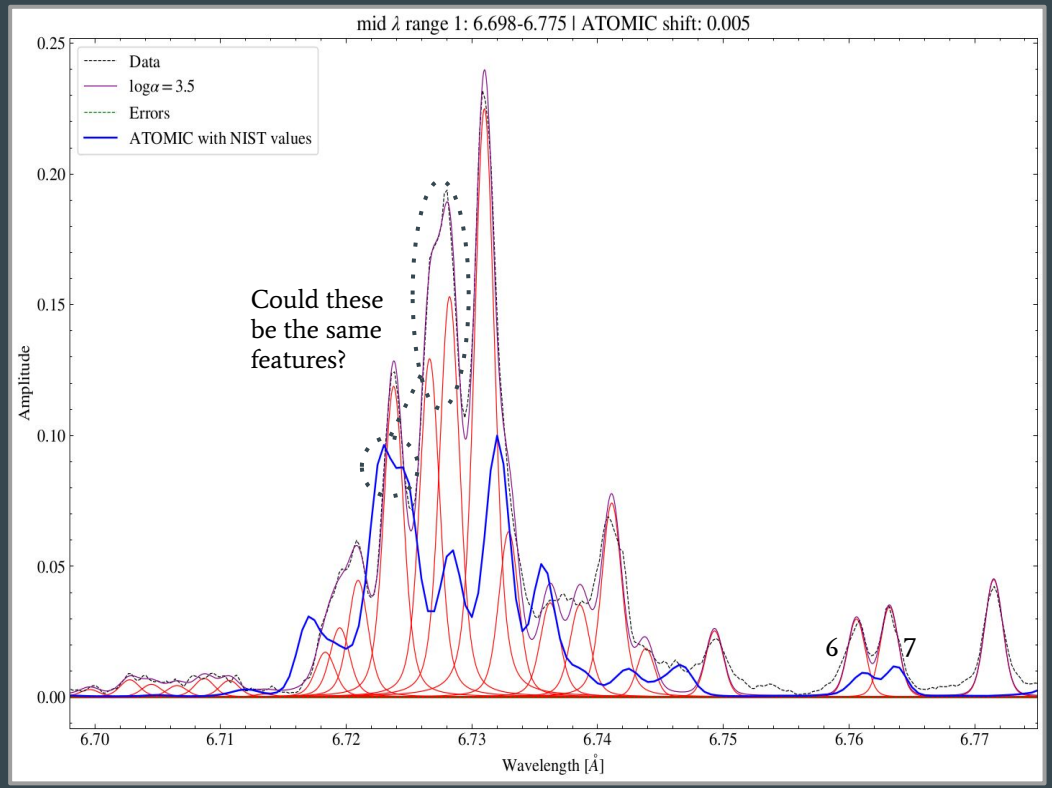
label	Charge	Upper	J	Lower	J
1	He-like	1s1 2p1	1.0	1s2	0.0
2	Li-like Li-like Li-like	1s1 2p1 3p1 1s1 2p1 3p*1 1s1 2p1 3p*1	1.5 0.5 1.5	1s2 3p1 1s2 3p*1 1s2 3p*1	1.5 0.5 0.5
3	Li-like	1s1 2p1	0.5	1s2 2p*1	0.5
4	Li-like	1s1 2p2	0.5	1s2 2p*1	0.5
5	He-like Li-like	1s1 2p*1 1s1 2s1 2p1	1.0 0.5	1s2 1s2 2s1	0.0 0.5



Z2972 low/mid λ range: 6.698-6.775 Å

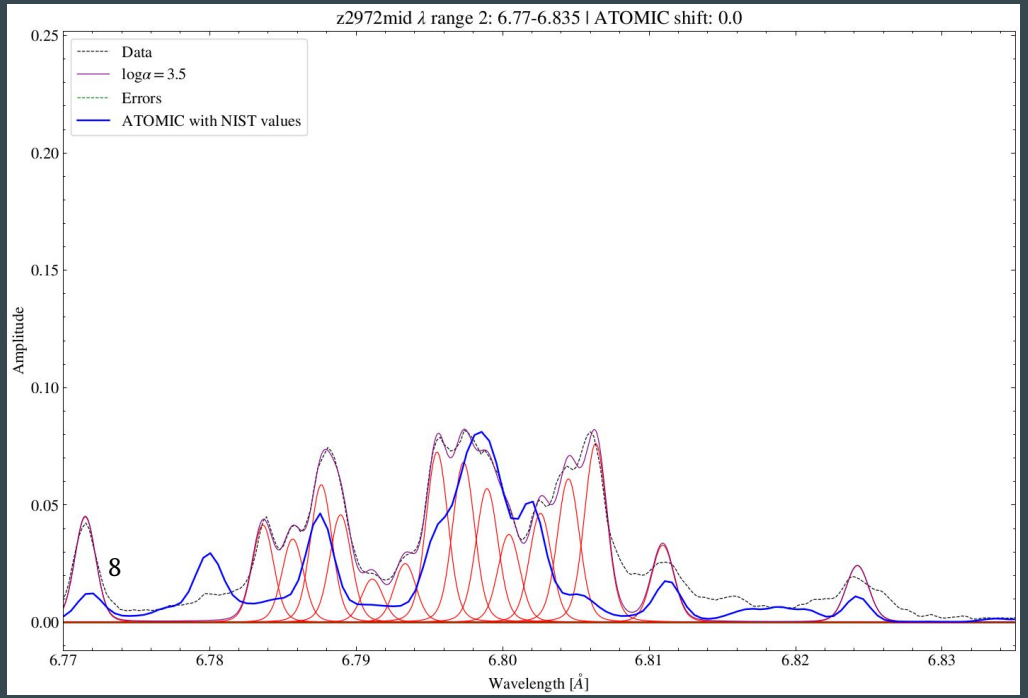
We would probably benefit from higher resolution calculations from here to higher wavelengths.

6 & 7 seem like clear matches but I don't have the label yet from Chris



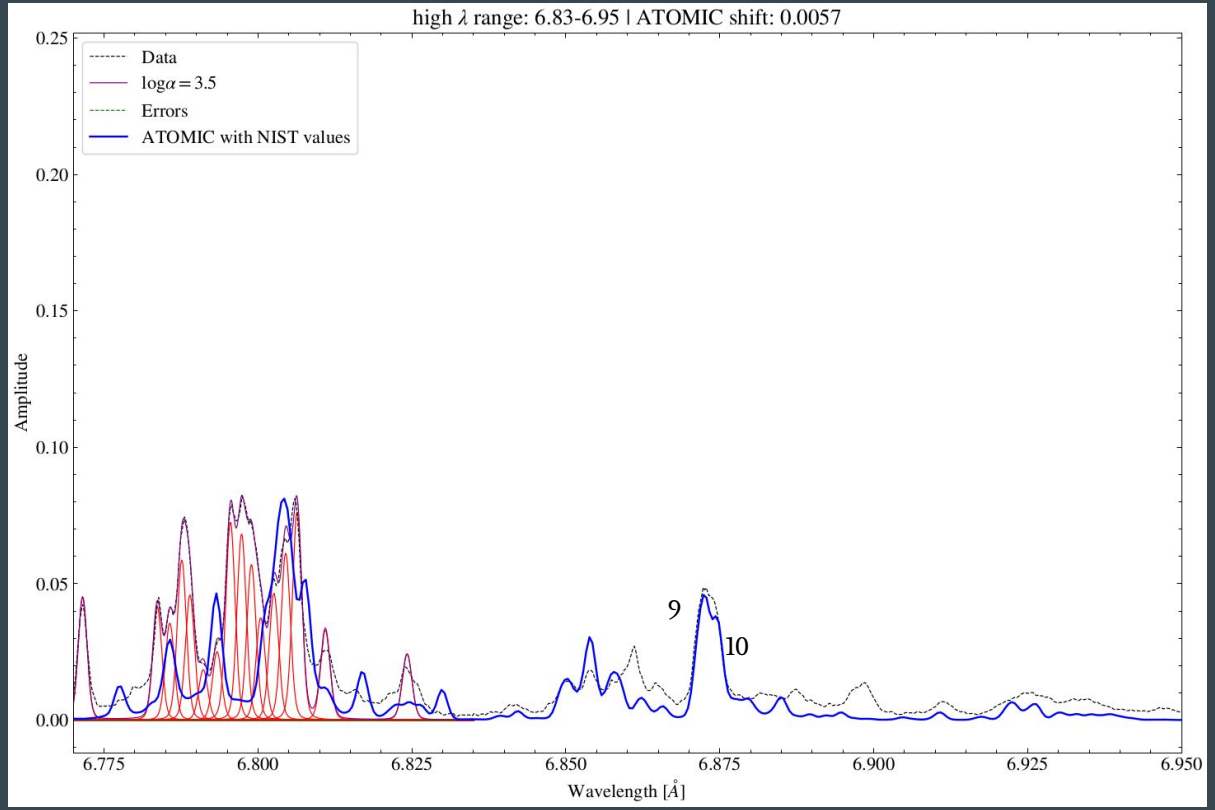
Z2972 high/mid λ range: 6.77-6.835 Å

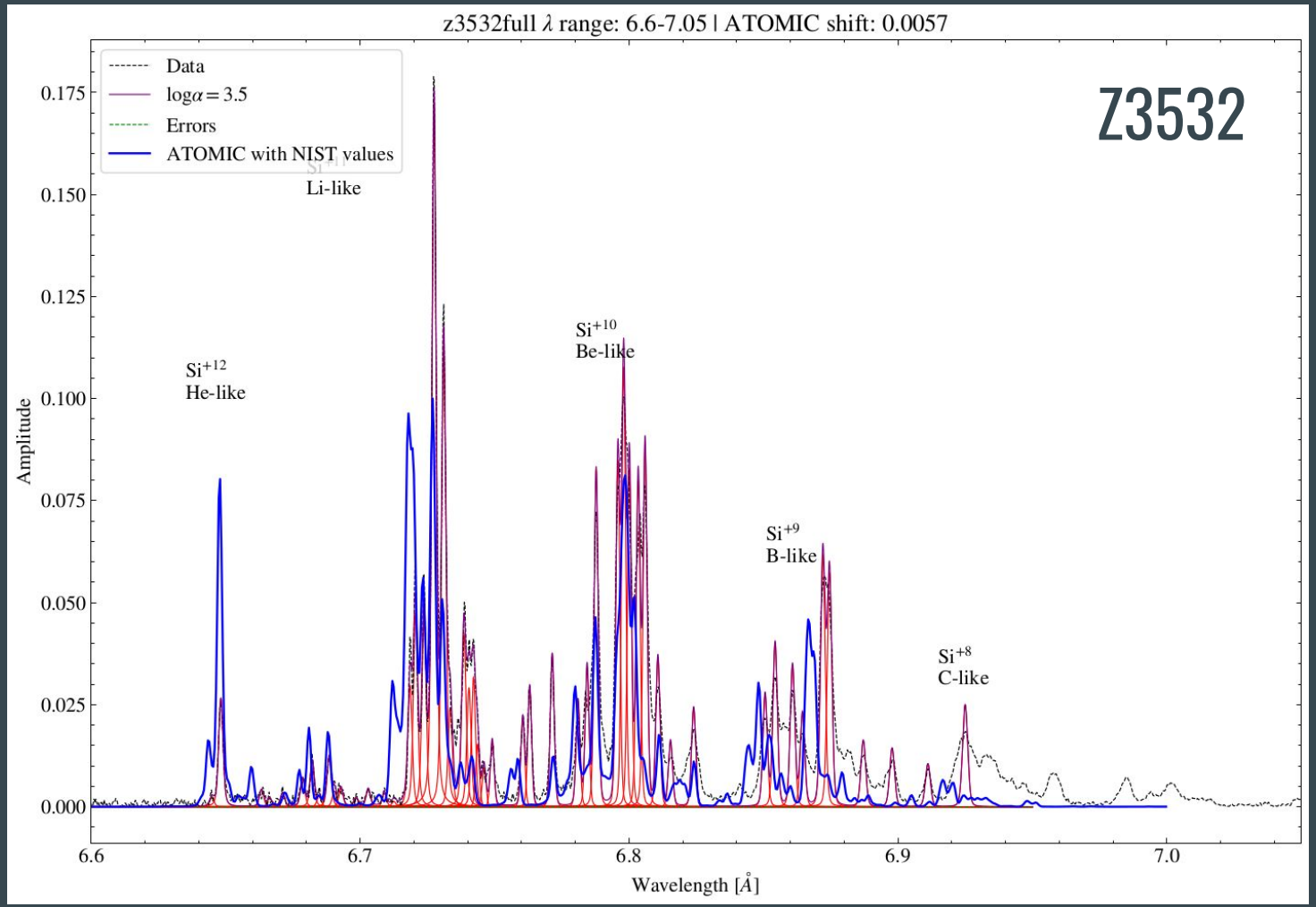
8 seems like another clear match but I don't have the label yet from Chris



Z2972 high λ range: 6.83-6.95 Å

9 & 10 seem like two more clear matches. There are also some other seemingly clear matches among the B-like features.




z3532full λ range: 6.6-7.05 | ATOMIC shift: 0.0057
Z3532
Data
logα = 3.5
Errors
ATOMIC with NIST values
Si+13
Li-like
Si+12
He-like
Si+10
Be-like
Si+9
B-like
Si+8
C-like
Amplitude
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
6.6
6.7
6.8
6.9
7.0
Wavelength [Å]

Extra Slides

How it Works: Make Initial Guesses for μ⟶ σ⟶ amplitude

- Choose high quality initial guesses for number of components and their parameters using a set of mathematical criteria.
- Derivative Spectroscopy - technique of analyzing a spectrum's derivatives
- Start by identifying means, those then give you estimates of stdev and amp.

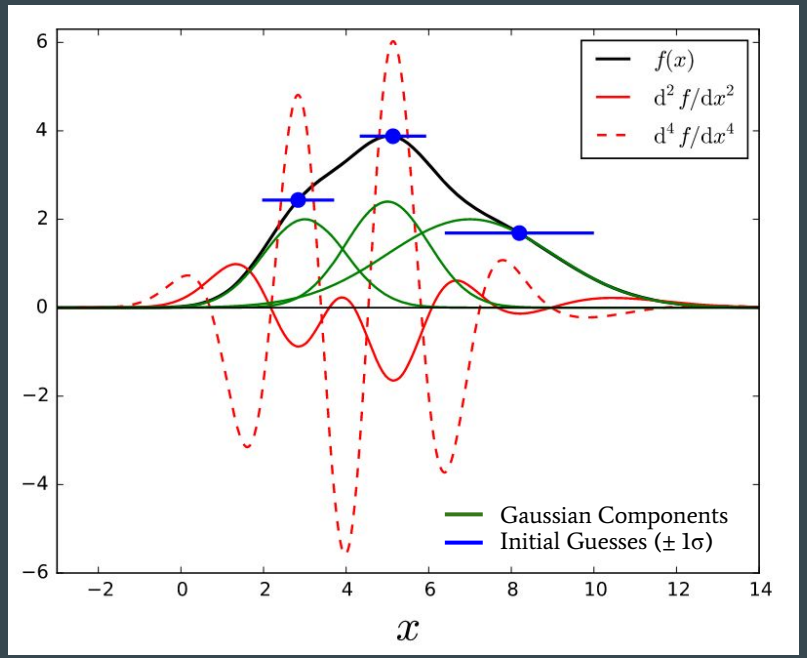
1. intensity exceeds noise
2. Curvature is negative
3. & 4. ensure it's a local min

$$f > \epsilon_0$$

$$\mathrm{d}^2 f/\mathrm{d}x^2 < 0$$

$$\mathrm{d}^3 f/\mathrm{d}x^3 = 0$$

$$\mathrm{d}^4 f/\mathrm{d}x^4 > 0.$$



How it Works: Regularized Differentiation

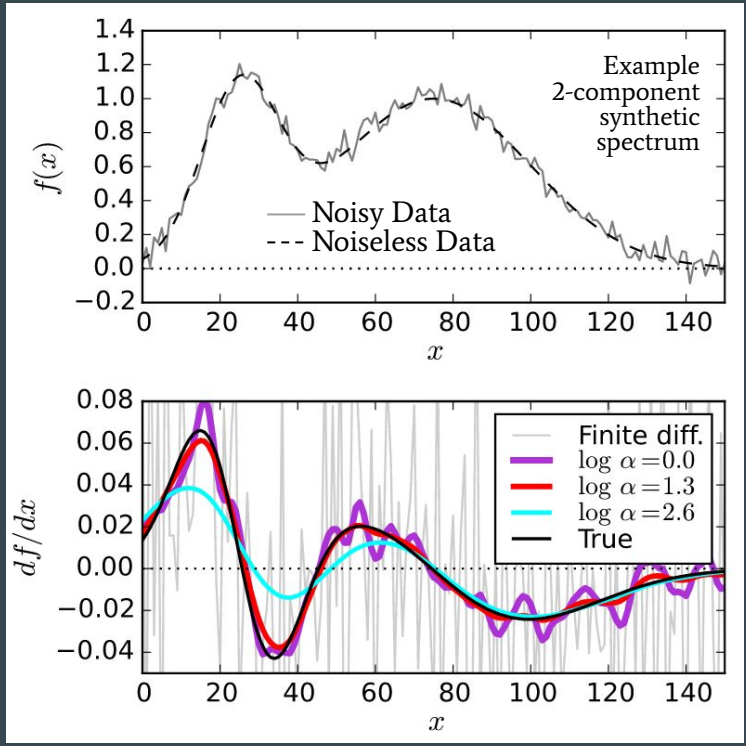
- Derivatives need to be accurate and smoothly varying to identify components.
  - Noise will produce spurious component guesses
  - Finite difference techniques greatly amplify noise in the data
- Regularization done via Tikhonov regularization
  - Derivative is fit to the data under the constraint that it remains smooth

How it Works: Regularized Differentiation

Regularized Derivative: min(R[u])

$$R[u] = \alpha \int \sqrt{(D_x u)^2 + \beta^2} + \int |A_x u - f|^2,$$

- Smooths the derivative (first term) while maintaining data fidelity (second term).
- Magnitude of α defines relative balance between the two.
- α = 0 → finite difference derivative
- $\log_{10}(\alpha)$ often used
- Returns higher accuracy derivatives compared to Gaussian convolution.



Choice of α is optimized using Machine Learning Techniques

Supervised Machine Learning - Computer is “trained” to generate predictions

- Given a training set: collection of input/output pairs
  - Input: spectral emission data,
  - Output: number and parameters of Gaussian components
- “Learns” a general rule for mapping inputs to outputs
  - Iterative process involves assigning a “grade” or accuracy to the machine’s guess for α
  - Guess is updated via least squares minimization - Levenberg-Marquardt algorithm

	Manually decompose a subset of the data	**Generate new synthetic data based on science data**
Input	Actual science data	Synthetic data
Output	Manually determined components	Components based on science data
Pro	Training data are same as science data	Decompositions are guaranteed to be “correct”
Con	Decompositions not guaranteed to be “correct”	Risk that training data are different from science data

How it Works: Make Initial Guesses for μ

- Choose high quality initial guesses
  - x and f(x) are frequency and units of flux density
- Derivative Spectroscopy - technique of analyzing a spectrum's derivatives
  - Places one guess at the location of every local minimum of negative curvature (defined as second derivative) in the data
  - Mathematical criteria:

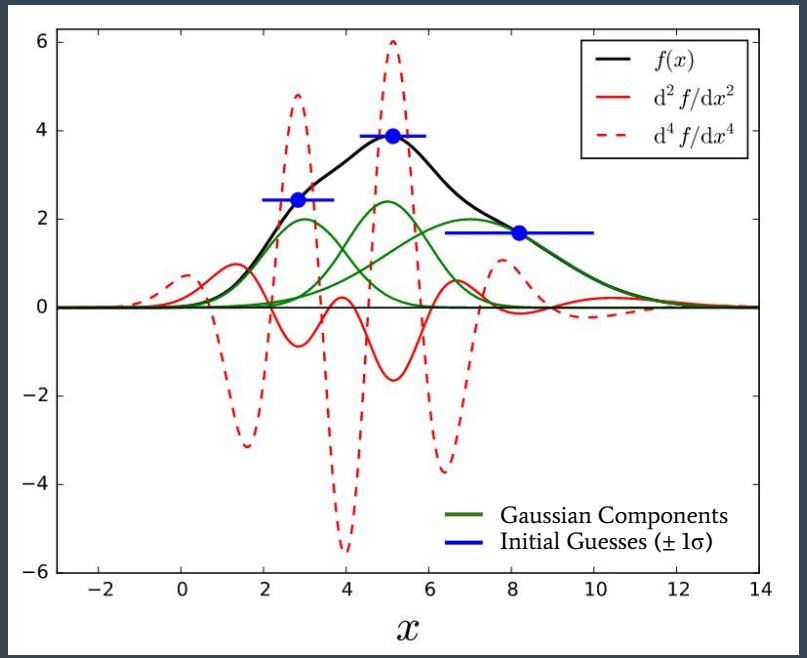
1. intensity exceeds noise
2. Curvature is negative
3. 4. ensure it's a local min

$$f > \epsilon_0$$

$$\mathrm{d}^2 f/\mathrm{d}x^2 < 0$$

$$\mathrm{d}^3 f/\mathrm{d}x^3 = 0$$

$$\mathrm{d}^4 f/\mathrm{d}x^4 > 0.$$



How it Works: Make Initial Guesses for σ

- Guess component's width by exploiting the relation between width and maximum of second derivative.
- Approximate a≈f(x)
- Rearrange to get estimate for σ

$$G(x;\, a,\, \mu,\, \sigma) = a\, e^{-(x-\mu)^2/2\sigma^2}.$$

$$\left.\frac{\mathrm{d}^2}{\mathrm{d}x^2} G(x;\, a,\, \mu,\, \sigma)\right|_{x=\mu} = -\frac{a}{\sigma^2}.$$

$$\sigma_n^2 = f(x)\left(\frac{\mathrm{d}^2 f(x)}{\mathrm{d}x^2}\right)^{-1}\Big|_{x=\mu_n}.$$

How it Works: Make Initial Guesses for Amplitude

- Naive estimates are values of the original data evaluated at the component positions.
- But with significant blending, this can significantly overestimate the guesses
- AGD attempts to “de-blend” the amplitude guesses using the information in the already-produced position and width guesses.
- I haven’t yet dug into the details of the de-blending process

How it Works: Regularized Differentiation

$$R[u] = \alpha \int \sqrt{(D_x u)^2 + \beta^2} + \int |A_x u - f|^2,$$

Regularization Term
Constrains u to be smoothly varying

How it Works: Regularized Differentiation

$$R[u] = \alpha \int \sqrt{(D_x u)^2 + \beta^2} + \int |A_x u - f|^2,$$

Regularization Term
Constrains u to be smoothly varying

Data Fidelity Term
Enforces that the integral of u closely follows the data (f)

How it Works: Regularized Differentiation

$$R[u] = \alpha \int \sqrt{(D_x u)^2 + \beta^2} + \int |A_x u - f|^2,$$

Alpha
Controls relative balance between smoothness and data fidelity

Regularization Term
Constrains u to be smoothly varying

Data Fidelity Term
Enforces that the integral of u closely follows the data (f)