Implementation of the Sparse Modeling Method for Analytical Continuation

Tianran Chen[a,*], Emanuel Gull[b]

[a] *Department of Physics, West Chester University, PA 19380, USA*
[b] *Department of Physics, University of Michigan, Ann Arbor, MI 48109, USA*

Abstract

We present SpM, a tool for performing analytic continuation of spectral functions using the maximum entropy method. The code operates on discrete imaginary axis datasets (values with uncertainties) and transforms this input to the real axis. The code works for imaginary time and Matsubara frequency data and implements the 'Legendre' representation of finite temperature Green's functions. It implements a variety of kernels, default models, and grids for continuing bosonic, fermionic, anomalous, and other data. Our implementation is licensed under GPLv3 and extensively documented. This paper shows the use of the programs in detail.

Keywords: Sparse Modeling Method, Analytic Continuation

PROGRAM SUMMARY

Manuscript Title: Implementation of the Sparse Modeling Method for Analytic Continuation
Authors: Tianran Chen and Emanuel Gull
Program Title: SpM
Journal Reference:
Catalogue identifier:
Licensing provisions: GPLv3
Programming language: C++
Operating system: Tested on Linux and Mac OS X
RAM: 10 MB – 200 MB
Keywords: Maximum Entropy Method, Analytic Continuation
Classification: 4.9
External routines/libraries: ALPSCore [**?**][**?**], GSL, HDF5
Nature of problem: The analytic continuation of imaginary axis correlation functions to real frequency/time variables is an ill-posed problem which has an infinite number of solutions.

Solution method: The sparse modeling method obtains a possible solution that maximizes entropy, enforces sum rules, and otherwise produces 'smooth' curves. Our implementation allows for input in Matsubara frequencies, imaginary time, or a Legendre expansion. It implements a range of bosonic, fermionic and generalized kernels for normal and anomalous Green's functions, self-energies, and two-particle response functions.

Running time: 10s - 2h per solution

[1] B. Bauer, et al., The ALPS project release 2.0: open source software for strongly correlated systems, J. Stat. Mech. Theor. Exp. 2011 (05) (2011) P05001. arXiv:1101.2646, doi:10.1088/1742-5468/2011/ 05/P05001.

[2] A. Gaenko, E. Gull, A. E. Antipov, L. Gamper, G. Carcassi, J. Paki, R. Levy, M. Dolfi, J. Greitemann, J.P.F. LeBlanc, Alpscore: Version 0.5.4doi: 10.5281/zenodo.50203.

1. Introduction

Analytical continuation has been a long-standing problem in quantum many-body physics. In numerical simulations of many-body systems such as quantum Monte Carlo lattice and impurity solvers, correlation functions are usually calculated in the imaginary time formalism; in experiments, however, it's the real frequency response

*Corresponding author.
E-mail address: tchen@wcupa.edu

or spectral functions that are measured. To compare with experimental results, one needs to use analytical continuation to convert the numerical data from imaginary to real axis.

Analytical continuation, however, is very sensitive to noise, especially for quantum Monte Carlo (QMC) simulations. Any small fluctuations in the input data, such as Monte Carlo statistical noise or finite precision, may lead to large fluctuations of the output data. For this reason, the standard Pade approximation often generates unphysical solutions. To reduce sensitivity to noise, a variety of alternatives have been proposed, including the maximum entropy method [ref], a constrained optimization procedure [ref], and a number of stochastic methods [ref].

In this paper, we will discuss a more recent algorithm: sparse modeling (SpM), which utilizes some of the new techniques developed in data science and information theories. Compared to other methods, SpM proves to have the advantage of obtaining stable and physical real-frequency spectra, by eliminating the redundant variables that may cause overfitting. In the remainder of the paper, we will briefly review the formalism of SpM method and introduce its implementation with examples. Our implementation is part of the ALPS applications and makes use of the core ALPS libraries.

2. Sparse Modeling Approach

2.1. Analytical Continuation

In diagrammatic perturbation theory and quantum Monte Carlo simulations, Green's function is computed in the imaginary time as $G(\tau)$, with $0 < \tau < \beta$, or in the imaginary frequency as $G(i\omega_n)$ through a Fourier transform:

$$G(i\omega_n) = \int_0^\beta e^{i\omega_n \tau} G(\tau), \tag{1}$$

where $i\omega_n$ are the Matsubara frequencies: $i\omega_n = 2\pi(n + \frac{1}{2})/\beta$ for fermionic and $i\omega_n = 2\pi n/\beta$ for bosonic operators. In the case of fermions, $G(\tau)$ and $G(i\omega_n)$ are related to a real frequency Green's function $G(\omega)$ via

$$G(\tau) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{d\omega\, \mathrm{Im}\,[G(\omega)]\, e^{-\tau\omega}}{1 + e^{-\beta\omega}}, \tag{2}$$

$$G(i\omega_n) = \frac{-1}{\pi}\int_{-\infty}^{\infty} \frac{d\omega\, \mathrm{Im}\,[G(\omega)]}{i\omega_n - \omega}, \tag{3}$$

Defining the spectral function

$$\mathcal{A}(\omega) = -\frac{1}{\pi}\mathrm{Im}\,[G(\omega)]\,, \tag{4}$$

one can formulate Eq. 2 as

$$G(\tau) = \int_{-\infty}^{\infty} d\omega\, K(\tau,\omega)\mathcal{A}(\omega), \tag{5}$$

where $0 \le \tau \le \beta \equiv 1/T$, and the kernel $K(\tau,\omega)$ is given by

$$K(\tau,\omega) = -\frac{e^{-\tau\omega}}{1 + e^{-\omega\beta}}. \tag{6}$$

In analytical continuation, the imaginary time Green's function $G(\tau)$ produced from QMC simulations is the input data, while $\mathcal{A}(\omega)$, the desired real frequency spectral function, is the output data. Thus, analytical continuation is essentially an inversion problem: for a given input $G(\tau)$, one can infer $\mathcal{A}(\omega)$ from the inverse of the integral.

However, due to the bad conditioning of kernel $K(\tau,\omega)$, Eq. 5 becomes very sensitive to noise: even tiny fluctuations in $G(\tau)$ may lead to large variations of $\mathcal{A}(\omega)$, making it difficult to select the reasonable solution among a large number of possible spectra.

Thus, instead of solving the spectral function exactly, an alternative way is to minimize the deviation between an approximate $\mathcal{A}$ and the original input G. For simplicity, we write Eq. 5 as a linear vector equation:

$$\boldsymbol{G} = K\boldsymbol{\mathcal{A}} \tag{7}$$

where $G_i \equiv G(\tau_i)$, $K_{ij} \equiv K(\tau_i,\omega_j)$, and $\mathcal{A}_j \equiv \mathcal{A}(\omega_j)\Delta\omega$. Here, τ is discretized to M points in the range of $[0 : \beta]$, while ω is discretized to N points in the range of $[-\omega_{max} : \omega_{max}]$. The discretization can be in linear or non-linear manner.

One can characterize the deviation between the approximated $\mathcal{A}$ and the original input data as

$$\chi^2 = \frac{1}{2}||\boldsymbol{G} - K\boldsymbol{\mathcal{A}}||_2^2. \quad (8)$$

Here $||\cdot||_2$ denotes the L_2 norm, defined as $||\boldsymbol{\mathcal{A}}||_2 \equiv (\sum_j \mathcal{A}_j^2)^{1/2}$. To find the solution, one needs to optimize Eq. 8.

Because $\boldsymbol{\mathcal{A}}$ is a spectral function, the optimization of χ^2 is subject to two physical constraints: $\boldsymbol{\mathcal{A}}$ is exclusively positive and integrates to one:

$$\mathcal{A}_j \geq 0 \ \forall j, \quad (9)$$

$$\sum_j \mathcal{A}_j = 1. \quad (10)$$

2.2. SVD

Since matrix K is ill-conditioned and difficult to invert, one can use the singular value decomposition (SVD) to decompose K into:

$$K = USV^t, \quad (11)$$

where S is an $M \times N$ diagonal matrix, and U and V are orthogonal matrices of size $M \times M$ and $N \times N$, respectively. [This way the matrix S can be compressed suitably by removing eigenvalues that are zero within numerical precision. Note: I don't think this is what Hiroshi's paper does.]. With SVD, Eq. 8 becomes

$$\chi^2 := \frac{1}{2}||U^T\boldsymbol{G} - SV^T\boldsymbol{\mathcal{A}}||_2^2 =: \frac{1}{2}||y' - Sx'||_2^2, \quad (12)$$

where $y' = U^T\boldsymbol{G}$ and $x' = V^T\boldsymbol{\mathcal{A}}$.

2.3. L_1 regularization

Because the singular values $s_l (l = 0, 1, 2, \cdots)$ decay exponentially, the corresponding elements of x' in Eq. 12 are negligibly small, causing the optimization of χ^2 to be very sensitive to noise. In order to find a stable solution that is independent of noise, the authors of [ref] propose adding an additional term, the so-called L_1 norm, to Eq. 12:

$$|x'|_1 = \sum_j |x'_j|. \quad (13)$$

Therefore, one should minimize

$$L_\lambda(x') = \frac{1}{2}||y' - Sx'||_2^2 + \lambda|x'|_1, \quad (14)$$

where λ is a positive (a priori arbitrary) parameter used to penalize large $|x'|_1$. The L_1 norm plays the role of imposing sparseness on x', and therefore helps select the solution among a large number of possible spectra that all satisfy Eq. 7 within a given accuracy. For more detailed justification of L_1 regularization see [ref]. The optimization of Eq. 14 is of the form of a LASSO (Least Absolute Shrinkage and Selection Operators) problem with two physical constraints: a convex constraint for the positivity (Eq. 9), and a constraint for the norm (Eq. 10).

2.4. ADMM

The optimization of Eq. 14 without the two physical constraints can be carried out using ADMM as illustrated in [ref]. To minimize Eq. 14, it is equivalent to do the following:

$$\text{minimize } f(x') + g(z') \quad (15)$$

$$\text{subject to } x' - z' = 0, \quad (16)$$

where we define

$$f(x') = \frac{1}{2}||y' - Sx'||_2^2 \quad (17)$$

$$g(z') = \lambda|z'|_1. \quad (18)$$

The ADMM algorithm solves the optimization problem of the above form by forming an augmented Lagrangian with parameter ρ

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}||Ax + Bz - c'||_2^2. \quad (19)$$

The first two terms are the Lagrangian without the "augmented" constraint. The remaining part would be zero if the constraint is satisfied.

3. Examples

3.1. Exact $G(\tau)$
3.1.1. Triple Gaussian
3.1.2. Semicircle
3.1.3. Semi-sector
3.2. $G(\tau)$ with noise

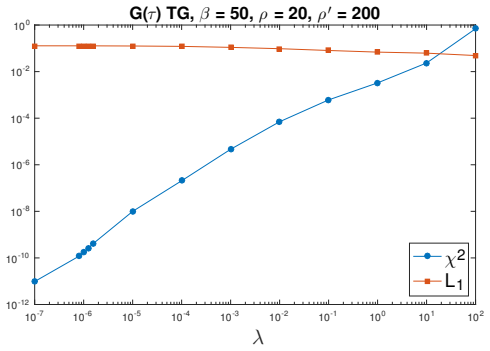

Figure 1: Exact triple Gaussian curve as input $G(\tau)$: dependence of L_2 and L_1 norm on parameter λ.

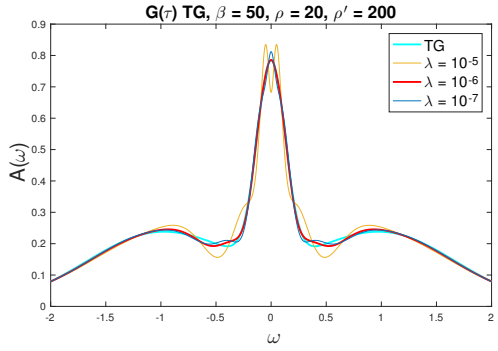

Figure 2: Exact triple Gaussian curve as input $G(\tau)$: spectral functions for chosen values of λ.

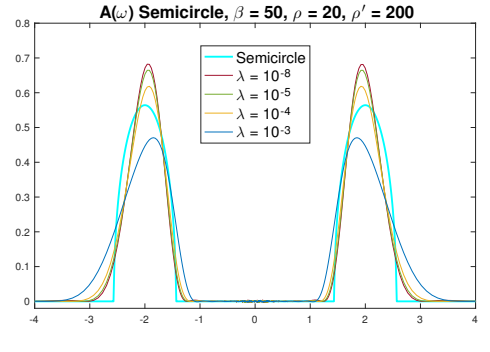

Figure 3: Exact triple Gaussian curve as input $G(\tau)$: spectral functions for chosen values of λ.

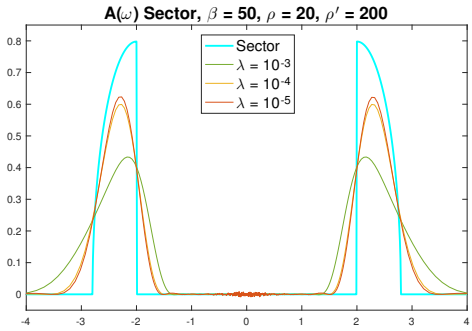

Figure 4: Exact triple Gaussian curve as input $G(\tau)$: spectral functions for chosen values of λ.

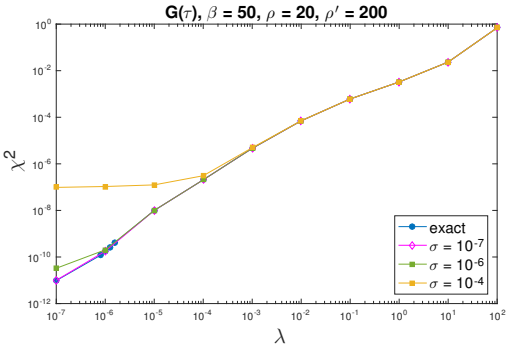

Figure 5: Triple Gaussian curve as input $G(\tau)$ with noise: dependence of χ^2 on λ.

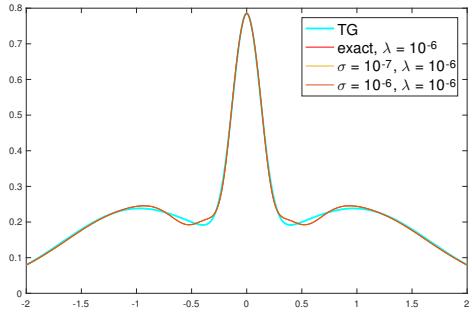

Figure 6: Triple Gaussian curve as input $G(\tau)$ with noise: spectral functions for chosen values of λ.