Evaluating $L$-functions with few known coefficients

David W. Farmer $^1$ Nathan C. Ryan $^2$

$^1$American Institute of Mathematics

$^2$Bucknell University

January 11th, 2013
Why compute $L$-functions?

- Riemann hypothesis
- Conjecture of Birch and Swinnerton-Dyer
- Random Matrix Theory
What is an $L$-function?

- **Dirichlet series:** $L(s) = \sum_{n=1}^{\infty} \frac{b_n}{n^s}$ where $b_1 = 1$.

- **Functional equation:** There exist $Q, \kappa_1, \ldots, \kappa_n \in \mathbb{R}_{>0}$, $\mu_1, \ldots \mu_n \in \mathbb{C}$ ($\text{Re}(\mu_i) \geq 0$), $\varepsilon \in \mathbb{C}$ ($|\varepsilon| = 1$), so that

  $$\Lambda(s) := Q^s \prod \Gamma(\kappa_j s + \mu_j) \cdot L(s) = \varepsilon \overline{\Lambda(1 - \bar{s})}$$

- **Euler product:** $L(s) = \prod_p L_p(p^{-s})^{-1}$ where $L_p$ is a polynomial with $L_p(0) = 1$.

- **Ramanujan bound:** $b_n = O(n^\varepsilon)$ for any $\varepsilon > 0$. 

David W. Farmer, Nathan C. Ryan

Evaluating $L$-functions with few known coefficients
What do I mean by computing an $L$-function?

- **An active area of research:** computing the coefficients $b_n$.
- **A hard, technical problem:** computing the data in the functional equation (the $\varepsilon$, the $Q$, the $\kappa$’s and the $\mu$’s).
- **A way to get evidence for conjectures:** Once you have coefficients and the functional equation you can evaluate $L(s)$ for particular $s \in \mathbb{C}$. 

Our interest is in computing high degree $L$-functions.

- If the $L$-function $L(s)$ has degree $d$, evaluating $L\left(\frac{1}{2} + it\right)$ using the approximate functional equation requires $\gg (1 + |t|)^{d/4}$ Dirichlet series coefficients. Here the implied constant depends on the $L$-function and the desired precision in the answer.

- Current methods are incapable of producing a large number of Dirichlet coefficients of, for example, the degree 5 and degree 10 $L$-functions of a Siegel modular form.
Statement of problem

▶ **Question:** Suppose you know the Dirichlet series coefficients $b_1, \ldots, b_N$. To what precision can you evaluate the $L$-function?

▶ **Answer:** Surprisingly well (to us, anyway).

▶ **Method:** We achieve a surprisingly high level of precision by averaging the standard computational approach over a collection of test functions, carefully selected so as to minimize the test functions’ contribution to the error.
Approximate Functional Equation

**Theorem**

Let $g : \mathbb{C} \to \mathbb{C}$ be given by $g(s) = e^{ibs + cs^2}$ and let $L(s)$ be as above and entire. Then

$$\Lambda(s)g(s) = Q^s \sum_{n=1}^{\infty} \frac{b_n}{n^s} f_1(s, n) + \varepsilon Q^{1-s} \sum_{n=1}^{\infty} \frac{\overline{b_n}}{n^{1-s}} f_2(1-s, n)$$

where

$$f_1(s, n) := \frac{1}{2\pi i} \int_{\nu-i\infty}^{\nu+i\infty} \prod_{j=1}^{a} \Gamma(\kappa_j(z + s) + \lambda_j)z^{-1}g(s + z)(Q/n)^z dz$$

$$f_2(1-s, n) := \frac{1}{2\pi i} \int_{\nu-i\infty}^{\nu+i\infty} \prod_{j=1}^{a} \Gamma(\kappa_j(z + 1-s) + \overline{\lambda_j})z^{-1}g(s - z)(Q/n)^z dz$$

with $\nu > \max \{ 0, -\text{Re}(\lambda_1/\kappa_1 + s), \ldots, -\text{Re}(\lambda_a/\kappa_a + s) \}$. 
Computation details

- Compute the integrals via Riemann sums
- For the unknown coefficients we bound them according to the Ramanujan bound.
- $Z$-function vs $L$-function.
Figure: The solid line is the calculated value and the dashed line is the error estimate in computing $Z(\frac{1}{2} + 10i)$ for the degree $5$ $L$-function using the available Dirichlet coefficients with the weight function $g(s) = e^{-i\beta s}$ where $\beta$ is given along the horizontal axis. The vertical axis is $\log_{10}$ of (the absolute value of) the actual value.
Figure: Same as previous figure but for $Z(\frac{1}{2} + 5i)$ for the degree 10 $L$-function with the weight function $g(s) = e^{-i\beta s + \frac{1}{500}(s-5i)^2}$ where $\beta$ is given along the horizontal axis.
Consider the degree 5 $L$-function and compute

$$Z(s) = \sum_{j=1}^{J} c_{\beta_j} Z(s, \beta_j)$$

where $\sum c_{\beta_j} = 1$ and where $Z(s, \beta_j)$ is the approximation we get using the test function $g(s) = e^{-i\beta_j s}$. We make the specific choices

$$s = \frac{1}{2} + 10i$$

$$(\beta_1, \beta_2, \beta_3, \beta_4, \beta_5) = \left(\frac{1}{10}, \frac{1}{5}, \frac{3}{10}, \frac{2}{5}, \frac{1}{2}\right)$$

$$(c_{\beta_1}, c_{\beta_2}, c_{\beta_3}, c_{\beta_4}, c_{\beta_5}) = (0.03150, 0.18061, 0.36563, 0.31421, 0.10801).$$
With these choices, after substituting the known Dirichlet coefficients and then using the Ramanujan bound, we find

\[ Z(s) = 3.039307086489527827801 + 2.688 \cdot 10^{-19} b_{83} + \cdots \\
- 5.291 \cdot 10^{-18} b_{137} + \cdots \\
= 3.039307086489527827827 \pm 4.737 \cdot 10^{-15}. \]

Thus, by averaging only 5 evaluations of the \( L \)-function, the error decreased by a factor of almost \( 10^{-5} \).
Figure: The error obtained from averaging $n$ evaluations of $Z(\frac{1}{2} + 10i)$ for the degree 5 $L$-function using the weight functions $g(s) = e^{-i\beta s}$ with $\beta = j/10$. The horizontal axis is $n$ and the vertical axis is $\log_{10}$ of the resulting error estimate.
Degree 10

Choose

\[ s = \frac{1}{2} + 5i \]

\[ (\beta_1, \beta_2, \beta_3, \beta_4, \beta_5) = \left( \frac{3}{5}, \frac{6}{5}, \frac{9}{5}, \frac{12}{5}, 3 \right) \]

\[ (c_{\beta_1}, c_{\beta_2}, c_{\beta_3}, c_{\beta_4}, c_{\beta_5}) = (0.035863, 0.33504, 0.47934, 0.13827, 0.01146). \]

The result is

\[ Z\left(\frac{1}{2} + 5i\right) = 0.01556 \pm 0.0049. \]

Using this method, the best result we were able to obtain, by averaging 11 evaluations, is

\[ Z\left(\frac{1}{2} + 5i\right) = 0.01558768 \pm 0.00016. \]
More computation details

- If $Z(s, \beta_j) = \text{calculated value}(s) \pm \text{error estimate}(s)$, then write

$$\text{error estimate}(s) = \sum_{n: b_n \text{ unknown}} \delta_n(\beta_j, s) b_n.$$  

- The product of $\delta_n(\beta_j, s)$ and Ram($b_n$), the Ramanujan bound for $b_n$, gives an upper bound for the contribution of the coefficient $b_n$ to the error.

- The coefficients $c_{\beta_j}$ are determined by finding the least-squares fit to

$$\sum_{n: b_n \text{ unknown}} \text{Ram}(b_n)^2 \left( \sum_j c_{\beta_j} \delta(\beta_j, s) \right)^2 = 0.$$  

- Implemented independently in Mathematica and PARI/GP.
Open questions

- Can the method of calculating an $L$-function by evaluating the approximate functional equation and then averaging to minimize the contributions of the unknown coefficients, determine numerical values of the $L$-function to arbitrary accuracy?

- Can one devise a method of determining an optimal weight function in the approximate functional equation without first calculating a large number of terms which do not actually contribute to the final answer.

- Is there any meaning to the weights determined by the least-squares method?