An idea of numerical conformal bootstrap

Gaoli Chen

The idea

In this note, we propose a method of numerical conformal bootstrap. The idea is similar to the deep learning algorithm: define a cost function (based on the crossing symmetry) in terms of CFT data and then train (minimize) the cost function with large amount of input data. The input data can be generated randomly in our case.

Consider the four-point function of scalar primaries

$$\langle \phi_i(x_1)\phi_j(x_2)\phi_k(x_3)\phi_l(x_4)\rangle = \sum_{\mathcal{O}} \frac{\lambda_{ij\mathcal{O}}\lambda_{kl\mathcal{O}}}{x_{12}^{\Delta_i+\Delta_j}x_{34}^{\Delta_k+\Delta_l}}\left(\frac{x_{24}}{x_{14}}\right)^{\Delta_{ij}}\left(\frac{x_{14}}{x_{13}}\right)^{\Delta_{kl}} g_{\Delta,l}^{\Delta_{ij},\Delta_{kl}}(u,v).$$

where

$$\Delta_{ij}=\Delta_i-\Delta_j,\quad u=\frac{x_{12}^2x_{34}^2}{x_{13}^2x_{24}^2},\quad v=\frac{x_{23}^2x_{14}^2}{x_{13}^2x_{24}^2}$$

and $g_{\Delta,l}^{\Delta_{ij},\Delta_{kl}}$ are conformal blocks (CBs). Applying the crossing symmetry $i\leftrightarrow k, x_1\leftrightarrow x_3$ to the four-point function gives rise to

$$f_{ijkl}(u,v)\equiv\sum_{\mathcal{O}}\left[\lambda_{ij\mathcal{O}}\lambda_{kl\mathcal{O}}v^{\frac{\Delta_k+\Delta_j}{2}}g_{\Delta,l}^{\Delta_{ij},\Delta_{kl}}(u,v)-\lambda_{kj\mathcal{O}}\lambda_{il\mathcal{O}}u^{\frac{\Delta_i+\Delta_j}{2}}g_{\Delta,l}^{\Delta_{kj}\Delta_{il}}(v,u)\right]=0. \tag{1}$$

$$f_{ijkl}(u,v)\equiv\sum_{\mathcal{O}}\left[\lambda_{ij\mathcal{O}}\lambda_{kl\mathcal{O}}u^{-\frac{\Delta_i}{2}}v^{\frac{\Delta_j}{2}}g_{\Delta,l}^{\Delta_{ij},\Delta_{kl}}(u,v)-\lambda_{kj\mathcal{O}}\lambda_{il\mathcal{O}}u^{\frac{\Delta_j}{2}}v^{-\frac{\Delta_k}{2}}g_{\Delta,l}^{\Delta_{kj}\Delta_{il}}(v,u)\right]=0$$

$$f_{ijkl}(u,v)\equiv\sum_{\mathcal{O}}\left[\lambda_{ij\mathcal{O}}\lambda_{kl\mathcal{O}}u^{-\frac{\Delta_i+\Delta_j}{2}}g_{\Delta,l}^{\Delta_{ij},\Delta_{kl}}(u,v)-\lambda_{kj\mathcal{O}}\lambda_{il\mathcal{O}}v^{-\frac{\Delta_k+\Delta_j}{2}}g_{\Delta,l}^{\Delta_{kj}\Delta_{il}}(v,u)\right]=0.$$

Eq. (1) holds only when the sum runs over all primaries $\mathcal{O}$, which is a set of infinite size. What if the sum only runs over a subset of the primaries $\{\mathcal{O}\}$ of dimension $\Delta\le\Delta_{max}$ and spin $l\le l_{max}$? It is natural to expect that the Eq. (1) holds approximately when $\Delta_{\max}$ and $l_{\max}$ are large enough.

Let us pick the following subsets of primaries:

- Lowest N scalar primaries: ϕ_1, ϕ_2, $\phi_3,\cdots,\phi_N$ with dimension $\Delta_1\le\Delta_2\le\cdots\le\Delta_N$.
- Lowest N_l symmetric traceless primaries of spin l, $2\le l\le l_{max}$: $\mathcal{O}_1^{(l)}$, $\mathcal{O}_2^{(l)},\cdots\mathcal{O}_{N_l}^{(l)}$ with dimension $\Delta_1^{(l)}\le\Delta_2^{(l)}\le\cdots\le\Delta_{N_l}^{(l)}$.

Of the N the scalar primaries, we choose the lowest $m < N$ scalars and consider all the four-point functions built out of these m scalars

$$\left\{ \langle \phi_i \phi_j \phi_k \phi_l \rangle \middle| 1 \leq i, j, k, l \leq m \right\}. \tag{2}$$

For each four-point function $\langle \phi_i \phi_j \phi_k \phi_l \rangle$, we have a function $f_{ijkl}(u, v)$ defined as (1). We then randomly pick 100 points (u, v),

$$P = \{(u_1, v_1), (u_2, v_2), \cdots, (u_{100}, v_{100})\},$$

and consider the following quantity as a function of the CFT data

$$F(\{\lambda_{ij\mathcal{O}}\}, \{\Delta_i\}) = \sum_{(u,v)\in P} \sum_{1\leq i,j,k,l\leq m} |f_{ijkl}(u, v)|^2.$$

When the primary subsets we choose are large, the function F should be very close zero, which is the minimum value of F. On the other hand, if we minimize F with respect $\{\lambda_{ij\mathcal{O}}\}$ and $\{\Delta_i\}$, the minima should be closely related to the CFT data. If the crossing symmetry is powerful enough, we could obtain approximate values of the CFT data (maybe only those of low-dimensional operators are close to the actual values); otherwise, we may obtain constraints or bounds on the CFT data.

Comments

- To handle constraints, such as $\Delta_1 \leq \Delta_2$ and unitary bounds, we can add additional terms to the F function, e.g,

  $$|f_{ijkl}(u, v)|^2 \to |f_{ijkl}(u, v)|^2 + \mathrm{ReLU}(\Delta_1 - \Delta_2) + \mathrm{ReLU}\left(\frac{d-2}{2} - \Delta_1\right),$$

  where $\mathrm{ReLU}(x)$ is defined as

  $$\mathrm{ReLU}(x) = \max(0, x)$$

- We might want to choose points in P in a region that the CBs have nice behavior, i.e, easy to compute. Probably, for some small $\delta > 0$,

  $$u = z\bar{z}, \quad \left|z - \frac{1}{2}\right| < \delta$$

  is a good choice.

- We can use Newton method to find minima of F. For the sake of performance, we need to implement the algorithm in C++. We need to find effective ways to calculate CBs and derivative of CBs with CFT data, for example $\frac{\partial}{\partial \Delta_i} g_{\Delta,l}^{\Delta_{ij},\Delta_{kl}}$. Not sure how difficult it is. Results of Refs [1, 2] seem to be very helpful.

- We consider only four-point functions of scalars because the CBs of scalars are simpler and better understood than spinning CBs. One can certainly include spinning CBs if there is no difficulty in calculation.

Recursion of Scalar Conformal Block

Let us present the recursion relation of scalar conformal block, which was discussed in ref. [2, 1].

In the radial coordinates

$$re^{i\theta} = \frac{z}{\left(1+\sqrt{1-z^2}\right)^2}, \quad re^{-i\theta} = \frac{\bar{z}}{\left(1+\sqrt{1-z^2}\right)^2},$$

the scalar CB can be presented as

$$\begin{aligned} g_{\Delta l}(r,\eta) &= (4r)^{\Delta} h_{\Delta l}(r,\eta) \\ h_{\Delta l}(r,\eta) &= h_{\infty l}(r,\eta) + \sum_A \frac{R_A}{\Delta - \Delta_A^*} (4r)^{n_A} h_{\Delta_A l_A}(r,\eta) \end{aligned}$$

where we have suppressed Δ_{12} and Δ_{34} dependence in the equations. The label A is defined as

$$A \equiv \text{Type}, n, \quad \begin{cases} \text{Type} & = \text{I, II, III} \\ n & = 1,\, 2,\, \cdots \end{cases}.$$

Δ_A and l_A are listed in the following table.

A	Δ_A^*	n_A	l_A
$\text{Type I}: \ n = 1,2,\dots\infty$	$1-l-n$	n	$l+n$
$\text{Type II}: \ n = 1,2,\dots l$	$l+2h-1-n$	n	$l-n$
$\text{Type III}: \ n = 1,2,\dots\infty$	$h-n$	$2n$	l

The residues R_A are

$$R_{\text{I},n} = \frac{-n(-2)^n}{(n!)^2} \left(\frac{\Delta_{12}+1-n}{2}\right)_n \left(\frac{\Delta_{34}+1-n}{2}\right)_n$$

$$R_{\text{II},n} = \frac{-nl!}{(-2)^n (n!)^2 (l-n)!} \frac{(2h+l-n-2)_n}{(h+l-n)_n (h+l-n-1)_n} \left(\frac{\Delta_{12}+1-n}{2}\right)_n \left(\frac{\Delta_{34}+1-n}{2}\right)_n$$

$$\begin{aligned} R_{\text{III},n} = & \frac{-n(-1)^n (h-n-1)_{2n}}{(n!)^2 (h+l-n-1)_{2n} (h+l-n)_{2n}} \left(\frac{\Delta_{12}-h-l-n+2}{2}\right)_n \\ & \times \left(\frac{\Delta_{12}+h+l-n}{2}\right)_n \left(\frac{\Delta_{34}-h-l-n+2}{2}\right)_n \left(\frac{\Delta_{34}+h+l-n}{2}\right)_n. \end{aligned}$$

Finally, $h_{\infty l}$ reads

$$h_{\infty l} = \frac{\left(1-r^2\right)^{1-h}}{\left(r^2-2\eta r+1\right)^{\frac{1-\Delta_{12}+\Delta_{34}}{2}} \left(r^2+2\eta r+1\right)^{\frac{1+\Delta_{12}-\Delta_{34}}{2}}} \frac{l!}{(-2)^l (h-1)_l} C_l^{h-1}(\eta).$$

We want to find derivative of CBs with repsect to Δ and Δ_i.

$$\partial_\Delta g_{\Delta l} = g_{\Delta l}\left(\ln(4r) + \frac{\partial_\Delta h_{\Delta l}}{h_{\Delta h}}\right) = \ln(4r)\, g_{\Delta l} + (4r)^{\Delta} \partial_\Delta h_{\Delta l}$$

$$\partial_\Delta h_{\Delta l} = -\sum_A \frac{R_A}{(\Delta - \Delta_A^*)^2} (4r)^{n_A} h_{\Delta_A l_A} (r, \eta) .$$

For Δ_i, we find

$$\frac{\partial g_{\Delta l}}{\partial \Delta_{12}} = (4r)^\Delta \frac{\partial h_{\Delta l} (r, \eta)}{\partial \Delta_{12}}$$

$$\frac{\partial h_{\Delta l} (r, \eta)}{\partial \Delta_{12}} = \frac{\partial h_{\infty l}}{\partial \Delta_{12}} + \sum_A \frac{(4r)^{n_A}}{\Delta - \Delta_A^*} \left[\frac{\partial R_A}{\partial \Delta_{12}} + \frac{\partial h_{\Delta_A l_A} (r, \eta)}{\partial \Delta_{12}} \right]$$

$$\frac{\partial R_{\mathrm{I},n}}{\partial \Delta_{12}} = \frac{1}{2} \left(\psi_0 \left(\frac{\Delta_{12} + 1 + n}{2} \right) - \psi_0 \left(\frac{\Delta_{12} + 1 - n}{2} \right) \right) R_{\mathrm{I},n}$$

$$\frac{\partial R_{\mathrm{I},n}}{\partial \Delta_{34}} = \frac{1}{2} \left(\psi_0 \left(\frac{\Delta_{34} + 1 + n}{2} \right) - \psi_0 \left(\frac{\Delta_{34} + 1 - n}{2} \right) \right) R_{\mathrm{I},n}$$

$$\frac{\partial R_{\mathrm{II},n}}{\partial \Delta_{12}} = \frac{1}{2} \left(\psi_0 \left(\frac{\Delta_{12} + 1 + n}{2} \right) - \psi_0 \left(\frac{\Delta_{12} + 1 - n}{2} \right) \right) R_{\mathrm{II},n}$$

$$\frac{\partial R_{\mathrm{II},n}}{\partial \Delta_{34}} = \frac{1}{2} \left(\psi_0 \left(\frac{\Delta_{34} + 1 + n}{2} \right) - \psi_0 \left(\frac{\Delta_{34} + 1 - n}{2} \right) \right) R_{\mathrm{II},n}$$

$$\begin{aligned} \frac{\partial R_{\mathrm{III},n}}{\partial \Delta_{12}} &= \frac{1}{2} \left[\psi_0 \left(\frac{\Delta_{12} - h - l + n + 2}{2} \right) - \psi_0 \left(\frac{\Delta_{12} - h - l - n + 2}{2} \right) \right] R_{\mathrm{III},n} \\ &+ \frac{1}{2} \left[\psi_0 \left(\frac{\Delta_{12} + h + l + n}{2} \right) - \psi_0 \left(\frac{\Delta_{12} + h + l - n}{2} \right) \right] R_{\mathrm{III},n} \end{aligned}$$

$$\begin{aligned} \frac{\partial R_{\mathrm{III},n}}{\partial \Delta_{34}} &= \frac{1}{2} \left[\psi_0 \left(\frac{\Delta_{34} - h - l + n + 2}{2} \right) - \psi_0 \left(\frac{\Delta_{34} - h - l - n + 2}{2} \right) \right] R_{\mathrm{III},n} \\ &+ \frac{1}{2} \left[\psi_0 \left(\frac{\Delta_{34} + h + l + n}{2} \right) - \psi_0 \left(\frac{\Delta_{34} + h + l - n}{2} \right) \right] R_{\mathrm{III},n} \end{aligned}$$

$$\frac{\partial h_{\infty l}}{\partial \Delta_{12}} = \frac{1}{2} h_{\infty l} \ln \left(\frac{r^2 - 2\eta r + 1}{r^2 + 2\eta r + 1} \right)$$

$$\frac{\partial h_{\infty l}}{\partial \Delta_{34}} = \frac{1}{2} h_{\infty l} \ln \left(\frac{r^2 + 2\eta r + 1}{r^2 - 2\eta r + 1} \right)$$

References

[1] Filip Kos, David Poland, and David Simmons-Duffin. Bootstrapping Mixed Correlators in the 3D Ising Model. *JHEP*, 11:109, 2014.

[2] JoÃ£o Penedones, Emilio Trevisani, and Masahito Yamazaki. Recursion Relations for Conformal Blocks. *JHEP*, 09:070, 2016.