

Bootcamp Tutorial

Bootstat 2021 29th May 2021

Marten Reehorst

Goal Tutorial

- Introduce concepts involved in practical numerics.
- Letting you compute your first bounds on CFT data.
- Introducing the tools used in cutting edge bootstrap computations.

Numerical Bootstrap

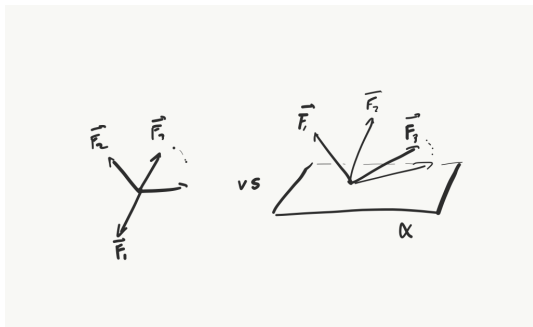
- Crossing equation and unitarity:

$$\sum_{\mathcal{O}} f_{\phi\phi\mathcal{O}}^2 \underbrace{\left(v^{\Delta_\phi} g_{\Delta,\ell}(u, v) - u^{\Delta_\phi} g_{\Delta,\ell}(v, u) \right)}_{F_{\Delta,\ell}^{\Delta_\phi}(u,v)} = 0.$$

- We can think of $F_{\Delta,\ell}^{\Delta_\phi}(u, v)$ as vectors in the infinite dimensional vector space of functions of u and v .
- The bootstrap consists of a search for a positive functional / a separating plane to exclude an assumed spectrum.

Numerical Bootstrap

- The bootstrap consists of a search for a positive functional / a separating plane to exclude an assumed spectrum.



Discretization of $F(z, zb)$

- Numerics require that we instead limit our search to some finite vector space.
- To do this we can evaluate $F(z, zb)$ at some fixed grid of points or take derivatives (usually at the self-dual point $z = zb = 1/2$)
- Construction of these basis vectors involves computing conformal blocks or derivatives of conformal blocks
- There are dedicated tools such as `scalar_blocks` to perform this task efficiently.

Numerical Bootstrap

- Similarly the spectrum (Δ, L) has to be represented by some finite amount of information. For this there are two approaches. Firstly, we can explicitly discretize and demand positivity only on a finite set of (Δ, L) . One then needs to check that the resulting bounds on the CFT data do not have much dependence on the choice of your discretization. A more rigorous method is also available. Positivity of a polynomial for all $x > 0$ can be related to the positive semi-definiteness of a finite matrix.
- Using the first approach the search for the functional can be written as a linear program. Using the second more rigorous approach one has to instead solve a semi-definite program.

The plan: Exercises

- A first exercise set can be done using only Mathematica and only build in functions. This will enable you to find your first bounds using a simplified workflow. This will serve well to illustrate the basic concepts involved. For this part we will consider $d = 2$ because the Conformal Block is in that case easy to compute from a closed form expression.
- In a second part we will show a more 'usual' workflow for a real bootstrap computation with the example of the computation of the famous $d = 3$ Ising model kink. Here we will use the arbitrary precision semi-definite programming solver SDPB that has become indispensable in the numerical bootstrap community. We will explain the in- and output involved and show an example interface to set up and solve the required semi-definite program.

The plan: Exercises

- Besides these questions of feasibility for a given spectrum there are other questions that can be asked numerically?
- For example we can compute bounds on OPE coefficient and bound for example the central charge c .
- We will also discuss how to set up the numerics for such a problem.

The plan: Exercises

- Recently we (MR, Slava Rychkov, David Simmons-Duffin, Benoit Sirois, Ning Su, Balt van Rees) introduced the concept of a *Navigator* function which instead offers a continuous measure of success. It tells you not only whether a point is allowed or disallowed but also how far you are from the boundary between the allowed and disallowed region and which direction you should move in to reach this boundary. This greatly improves the efficiency of tasks such as finding an isolated allowed point and finding the boundaries of an allowed region once an allowed point has been found.
- The computation of the navigator is very similar to the computation of bounds on OPE coefficients and we will end with an example computation of this *Navigator*.

Support on Slack

- Everyone is encouraged to ask questions on numerics on the bootcamp slack channel at any time.
- This can concern the exercises, the installation and the use of bootstrap tools such as `scalar_blocks` and SDPB or anything else related to bootstrap numerics.
- The planned live sessions are meant for you to work on the problems in the notebook, to ask questions, and/or discussing the exercises.
- If there is anything additional that you would like to learn or something difficult that you would like me to expand on please let me know.

Additional tools

Many other useful and interesting tools have been developed over the past years. Some examples of interesting ones are:

- Autoboot (by Mocho Go), this can automatically give you the crossing equations appropriately decomposed by irrep for many groups and representations.
- SimpleBoot (by Ning Su). This is an interface that can set up any PMP's arising in the conformal bootstrap given the crossing equations in the format outputted by Autoboot.
- For computations involving spinning external operators the set up is much more complicated. Some of this has recently been simplified for $d = 3$ by the availability of `blocks_3d` to automatically compute spinning blocks.