Shattering Coefficient

Alban de Crevoisier

The objective of this lab is to get an estimation of the shattering coefficient of the following deep-learning architecture:

$$Input(7) \rightarrow Hidden(9) \rightarrow Hidden(5) \rightarrow Output(3).$$

Each layer being fully connected to each other.

Computing an exact shattering coefficient for such an architecture would prove far too expensive computationally-wise, so instead we resort to an estimate. To do so, we generate a random sample of points and then random hyperplanes to try and generate all the possible shattering ways. This is but a gross upper bound, but it has the merit of being tractable.

To do this, I used the provided code to compute the Shattering coefficient for 20 points by a single hyperplane, and then took its power by the number of shattering hyperplanes. After fitting a polynomial curve to it, we can get a pretty false idea of what the shattering coefficient should look like: I'm afraid the lack of power of my machine is felt in these computation-heavy tasks.

The curve fitting code is in the python file, it can be run on the provided data that was computed with the provided R file with the following equation:

$$shattering(8)^9 + shattering(10)^5 + shattering(6)^3$$