A Superposition Model for Function Reconstruction from Data Samples in Higher Dimensions

M. Griebel[1] **and J. Braun**[2]

Abstract: In general, numerical methods underly the so called *curse of dimensionality*, a term that describes the exponential growth of the numerical costs of an algorithm with the dimensionality n of the given problem. This makes many algorithms unfeasable even for moderate dimensions $n > 3$. Here, we consider the reconstruction of an unknown function $f : \mathbb{R}^n \to \mathbb{R}$ from given data pairs $\{(\mathbf{x}_1, f(\mathbf{x}_1)), \ldots, (\mathbf{x}_P, f(\mathbf{x}_P))\}$. This is a very general formulation of a task that arises in many fields of applications like regression, classification, neural networks or data mining.
Our goal is now, to construct a method for which the numerical costs scale linearly in the amount of data P *and at worst polynomially* in the dimensionality n of the data points.

To this end, we employ a result of Kolmogorov from 1957. He showed that any multivariate function f can be represented as a superposition of one–dimensional inner and outer functions $\psi_{q,p}$, Φ_q by

$$f(x_1, \ldots, x_n) \;=\; \sum_{q=0}^{2n} \Phi_q \left(\sum_{p=1}^{n} \psi_{q,p}(x_p) \right).$$

After Hecht–Nielsen's interpretation of this representation as a feed–forward neural network in 1987, Kolmogorov's theorem also found attention in neural network computation. Amongst other things it was due to the non–constructiveness of the result that its applicability was fiercely discussed.

In this talk, we introduce a version of Kolmogorov's theorem that was shown by Sprecher in 1996, for which the inner functions are defined explicitly and independent of f. Then, based on Sprecher's and own results we define an algorithm to reconstruct f from the given data points with linear complexity in their number P. This algorithm only has to determine one single one-dimensional outer function. We therefore transformed the n–dimensional problem into an one–dimensional problem. Further investigations then show that the dependency on the dimension n is also present in this outer function. It turns out that the function is an oscillating function with high frequency numbers which depend on n. However, capturing the relevant frequencies in the right manner guarantees an algorithm with polynomial numerical costs in n.

Finally, we show numerical results for the application of our method to artificial and real data sets.

[1,2] Institute for Numerical Simulation, University of Bonn, Wegelerstr. 6, 53115 Bonn, Germany
braun@ins.uni-bonn.de