Scheme for positions of radial basis functions and radius considering supports for accuracy of approximation in convolute RBF

Masao Arakawa, Kagawa University, JAPAN, arakawa@eng.kagawa-u.ac.jp
Satoshi Kitayama, Kanazawa University, JAPAN

There are a lot of studies on surrogate optimization, and most of them use Gaussian distribution type of kernels as basis function. In these cases, there are a few keys to make success. No matter which type of unification, such as Kriking, SVR and/or RBF, this parameter optimization is the most important parts for better approximation. "Better" means not only giving accurate approximation for given data, but also for good approximation for less density area of given data. Without any doubt, if you give kernels on the given data, and give a small radius, local accuracy will rise, but there are no generality. Thus, it is very questionable to use approximate function without giving additional data around it. As for recommendation of teaching data, which is the second important key in surrogate optimization, important points are giving data for approximate optimum point and some additional points in less density area. These ideas are used in so called EGO, recommendation function by authors, and so on.

In this study, we propose a scheme to give kernel position together with setting radius for convolute RBF. In convolute RBF, we gave radius of each convolution priori to learning. And we give radius from large radius to small radius in a few steps. We assumed generality of each kernel can be determined from the number of given data with in radius of the kernel. We called the number of data for that kernel as supports. And we automatically adjust the given radius, so that the smallest radius kernel must have at least two supports to keep generality. For that purpose, we calculate distance between given data and find maximum of minimum distance and calculate ratio between given smallest radius, and adjust all radius before we give kernels. Next, we calculate average of function values of given data, and shift all data so that average of given data will come as 0. In each convolution, we give kernels position as follows. First, we calculate the maximum error position. Then calculate gravity position that is given by the data included with in the given radius from maximum error position. Next, we do the same thing for the minimum error position. And we learn from RBF and calculate error again. We do the same scheme up to the given number of kernels for each convolution. After learning, we optimize the approximate functions. We give additional data, not only to de optimum of full convolution, but also to some lower number of convolutions. In this study, we have examined the effectiveness of the proposed method with some benchmarking tests, and showed the effectiveness of the method.