A Dimension Adaptive Combination Technique Using Localised Adaptation Criteria

Jochen Garcke

Technische Universität Berlin, Institut für Mathematik, MA 3-3,
Straße des 17. Juni 136, 10623 Berlin, Germany garcke@math.tu-berlin.de

Summary. We present a dimension adaptive sparse grid combination technique for the machine learning problem of regression. A function over a d-dimensional space, which assumedly describes the relationship between the features and the response variable, is reconstructed using a linear combination of partial functions; these may depend only on a subset of all features. The partial functions, which are piecewise multilinear, are adaptively chosen during the computational procedure. This approach (approximately) identifies the ANOVA-decomposition of the underlying problem. We introduce two new localized criteria, one inspired by residual estimators based on a hierarchical subspace decomposition, for the dimension adaptive grid choice and investigate their performance on real data.

1 Introduction

Sparse grids are an approach for efficient high dimensional function approximation. They were introduced under this name for the numerical solution of partial differential equations, although the underlying idea was used first for numerical integration. These approaches are based on a multiscale tensor product basis where basis functions of small importance are omitted. In the form of the combination technique, sparse grids have successfully been applied to the machine learning problems of classification and regression using a regularization network approach [2, 3]. Here the problem is discretized and solved on an a priori chosen sequence of anisotropic grids with uniform mesh sizes in each coordinate direction. The sparse grid solution is then obtained from the solutions on these different grids by linear combination. This results in a non-linear function, while the computational complexity scales only linear in the number of data. The main difference in comparison to many other machine learning approaches is the choice of basis functions whose anchoring position is independent of the locations of the data.

Although sparse grids cope with the curse of dimensionality to some extent the approach still has high dependence on d, the number of dimensions. But

typically the importance of and variance within a dimension vary in real machine learning applications. This can exploited by different mesh resolutions for each feature. The degree of interaction between different dimensions also varies; the usage of all dimensions in each partial grid might be unnecessary.

In this spirit a so-called *dimension adaptive* algorithm [5, 7] to construct a generalized sparse grid was recently used for regularized least squares regression [4]; the idea is to choose the grids for the combination technique during the computation instead of defining them a priori. The aim is to attain a function representation for $f(\underline{x})$, with $\underline{x} = (x_1, \ldots, x_d)$, of the ANOVA type

$$f(\underline{x}) = \sum_{\{j_1,\ldots,j_q\}\subset\{1,\ldots,d\}} c_{j_1,\ldots,j_q} f_{j_1,\ldots,j_q}(x_{j_1}, \ldots, x_{j_q}),$$

where each $f_{j_1,\ldots,j_q}(x_{j_1}, \ldots, x_{j_q})$ depends only on a subset of size q of the dimensions and may have different refinement levels for each dimension. The computational complexity now depends only on the so-called *superposition* (or *effective*) dimension q.

Originally the overall error reduction was used as an adaptation criteria [4], but the computational effort of this criteria grows quite significantly with the number of grids. In this paper we introduce and investigate two different error indicators which are *localized* in some sense since one considers the improvement using subsets of all grids employed, and whose computational effort grows less with the number of partial grids.

In the following we describe the regularized regression approach, present the dimension adaptive combination technique, introduce the two new error indicators, and show results on machine learning benchmarks.

2 Dimension adaptive combination technique for regression

We assume that the relation between predictor and response variables can be described by an (unknown) function f which belongs to some space V of functions defined over the range of the predictor variables. To get a well-posed, uniquely solvable problem we use regularization theory and impose additional smoothness constraints on the solution of the approximation problem. In our regularized least squares approach this results in the variational problem

$$\operatorname*{argmin}_{f\in V} \frac{1}{M}\sum_{i=1}^{M}(f(\underline{x}_i) - y_i)^2 + \lambda||\nabla f||^2, \tag{1}$$

for a given a dataset $S = \{(\underline{x}_i, y_i) \in [0,1]^d \times \mathbb{R}\}_{i=1}^{M}$. Note that using the following semi-definite bilinear form

$$\langle u, v\rangle_{\text{RLS}} := \frac{1}{M}\sum_{i=1}^{M} u(\underline{x}_i)v(\underline{x}_i) + \lambda\langle\nabla u, \nabla v\rangle_2$$

corresponding to (1), the Galerkin equations are

$$\langle f, g\rangle_{\mathrm{RLS}} = \frac{1}{M}\sum_{i=1}^{M} g(\underline{x}_i)y_i, \tag{2}$$

which hold for the solution f of (1) and all $g \in V$.

The discretization to a finite dimensional subspace V_N of the function space V is achieved by the sparse grid combination technique [6]. To get a solution defined on a sparse grid we discretize and solve the problem (1) on a suitable sequence of small anisotropic grids $\Omega_{\underline{l}} = \Omega_{l_1,\ldots,l_d}$, characterized by an index set I, i.e. $\underline{l} \in \mathsf{I}$. These are grids which have different but uniform mesh sizes h_t in each coordinate direction with $h_t = 2^{-l_t}, t = 1,\ldots,d$. The grid points are numbered using the multi-index $\underline{j}, j_t = 0,\ldots,2^{l_t}$ and have the coordinate $j_t \cdot h_t$ in dimension t. A finite element approach with piecewise d-linear functions $\phi_{\underline{l},\underline{j}}(\underline{x}) := \prod_{t=1}^{d}\phi_{l_t,j_t}(x_t), \quad j_t = 0,\ldots,2^{l_t}$ on each grid $\Omega_{\underline{l}}$, where the one-dimensional basis functions $\phi_{l,j}(x)$ are the *hat* functions

$$\phi_{l,j}(x) = \begin{cases} 1 - |x/h_l - j|, & x \in [(j-1)h_l, (j+1)h_l] \\ 0, & \text{otherwise,} \end{cases}$$

now gives the function space $V_{\underline{l}} := \mathrm{span}\{\phi_{\underline{l},\underline{j}}, j_t = 0,\ldots,2^{l_t}, t = 1,\ldots,d\}$.

The sparse grid combination technique solution f_{I} for a given index set I is now built via [2, 3, 6, 8]

$$f_{\mathsf{I}}(\underline{x}) := \sum_{\underline{l}\in\mathsf{I}} c_{\underline{l}} f_{\underline{l}}(\underline{x}), \tag{3}$$

where f_{I} is an element of a discrete function space defined on a sparse grid, the $f_{\underline{l}}$ are partial solutions and the $c_{\underline{l}}$ are corresponding combination coefficients.

A general choice of grids was introduced in [7]. One considers an index set I which only needs to fulfil the following *admissibility condition* [5]

$$\underline{k} \in \mathsf{I} \text{ and } \underline{j} \le \underline{k} \quad \Rightarrow \quad \underline{j} \in \mathsf{I}, \tag{4}$$

an index $\underline{k}$ can only belong to the index set I if all smaller grids $\underline{j}$ belong to it. The combination coefficients, which are related to the *inclusion/exclusion* principle from combinatorics, depend only on the index set [7, 8].

In the original combination technique one considers an a priori chosen index set I consisting of all $\underline{l}$ with $|\underline{l}|_1 := l_1 + \cdots + l_d \le n$. This results in a sparse grid of refinement level n [2, 3, 6]. The size of an $\Omega_{\underline{l}}$ is here of order $\mathcal{O}(2^d \cdot h_n^{-1})$, while the total number of points used in the combination technique is of order $\mathcal{O}(2^d \cdot h_n^{-1} \cdot \log(h_n^{-1})^{d-1})$, the same as for a sparse grid.

The solution obtained this way is the same as the sparse grid solution if the projections into partial spaces commute, or at least of the same approximation order $\mathcal{O}(h_n^2 \cdot \log(h_n^{-1})^{d-1})$ (if the function has bounded mixed second derivatives) if a series expansion of the error of the form $f - f_{\underline{l}} =$

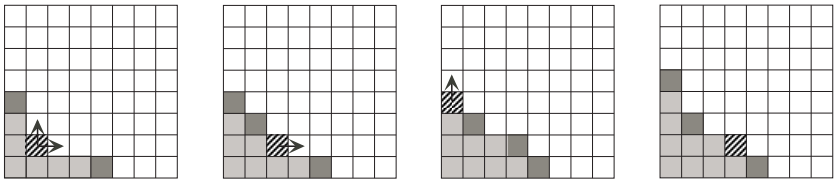

Fig. 1. A few steps of the dimension adaptive algorithm. Active indices $\underline{i} \in \mathsf{A}$ are shown in dark grey and old indices $\underline{i} \in \mathsf{O}$ in light grey. The chosen active index (with the largest error indicator) is shown striped. The arrows indicate the admissible forward neighbours which are added to A. The indexes go from -1 to 6.

$\sum_{i=1}^{d} \sum_{j_1,\ldots,j_m \subset 1,\ldots,d} c_{j_1,\ldots,j_m}(h_{j_1},\ldots,h_{j_m}) \cdot h_{j_1}^2 \cdot \ldots \cdot h_{j_m}^2$ exists [6, 8]. But for the machine learning application this does not hold [8]. Instead combination coefficients which also depend on the function to be represented are employed. They are optimal in the sense that the sum of the partial functions minimizes the error against the sparse grid solution computed directly in the joint function space [3, 8], this approach is valid for the machine learning setting as well. Note that the approximation properties of the optimized combination technique in relation to a sparse grid are currently unknown.

In any case we employ in (3) the optimal coefficients computed according to

$$\begin{bmatrix} \langle f_1, f_1\rangle_{\text{RLS}} & \cdots & \langle f_1, f_k\rangle_{\text{RLS}} \\ \langle f_2, f_1\rangle_{\text{RLS}} & \cdots & \langle f_2, f_k\rangle_{\text{RLS}} \\ \vdots & \ddots & \vdots \\ \langle f_k, f_1\rangle_{\text{RLS}} & \cdots & \langle f_k, f_k\rangle_{\text{RLS}} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix} = \begin{bmatrix} \|f_1\|_{\text{RLS}}^2 \\ \|f_2\|_{\text{RLS}}^2 \\ \vdots \\ \|f_k\|_{\text{RLS}}^2 \end{bmatrix}, \tag{5}$$

using a suitable numbering of the $f_{\underline{l}}$ [3, 8].

A generalization of the original sparse grid combination technique consists in the use of a slightly different level hierarchy. Let us formally define the one-dimensional basis functions $\widetilde{\phi}_{l,j}(x)$ as $\widetilde{\phi}_{-1,0} := 1, \widetilde{\phi}_{0,0} := \phi_{0,1}$, and $\widetilde{\phi}_{l,j} := \phi_{l,j}$ for $l \geq 1$, with $\phi_{l,j}$ as before. Note that it holds $\phi_{0,0} = \widetilde{\phi}_{-1,0} - \widetilde{\phi}_{0,0}$.

If one builds the tensor product between a constant in one dimension and a $(d-1)$-linear function the resulting d-dimensional function is still $(d-1)$-linear, one gains no additional degrees of freedom. But formally introducing a level -1, and using this as coarsest level in our adaptive procedure described in the next section, will allow us to build a combined function in the ANOVA-style, in other words each partial function might only depend on a subset of all features. The size of each grid $\Omega_{\underline{l}}$ is now of order $\mathcal{O}(2^q(|\underline{l}|_1 + (d-q))$, where $q = \#\{l_i | l_i \geq 0\}$.

2.1 Adaptive grid choice

Most important is the choice of a suitable index set I. One might be able to use external knowledge of the properties and the interactions of the dimensions

which would allow an a priori choice of the index set. In general the algorithm should choose the grids automatically in a *dimension adaptive* way during the actual computation. We therefore start with the smallest grid with index $\underline{-1} = (-1, \ldots, -1)$ (i.e., $\mathsf{I} = \{\underline{-1}\}$) which is just a constant function. Step-by-step we add additional indices such that:

(i) the new index set remains admissible;
(ii) the grid corresponding to the index provides a large reduction in (1).

During each adaptation step one has an outer layer of indices under consideration for inclusion in the sequence, the set of *active indices* denoted by A. Furthermore there is the set $\mathsf{O} = \mathsf{I} \backslash \mathsf{A}$ of *old indices* which already belong to the sequence of grids, O needs to fulfil (4). The *backward neighbourhood* of an index $\underline{k}$ is defined as the set $\mathsf{B}(\underline{k}) := \{\underline{k} - \underline{e}_t,\ 1 \leq t \leq d\}$; the set $\mathsf{F}(\underline{k}) := \{\underline{k} + \underline{e}_t,\ 1 \leq t \leq d\}$ is the *forward neighbourhood*, with $\underline{e}_t$ the unit vector in the t-th dimension. To limit the search range we restrict the active set A to only include indices whose backward neighbours are in the old index set O, in other words for $\underline{k} \in \mathsf{A}$ it holds that $\mathsf{O} \cup \underline{k}$ fulfils (4). Note that A cannot be empty since, at least, an index of the form $(-1, \ldots, k, \ldots, -1)$ for each coordinate direction is active.

In Figure 1 a few adaptation steps for the two dimensional case are presented. We assume here that the indices $(0,0), (1,0), (-1,2)$ and $(2,0)$ are chosen in succession. In each case their forward neighbours are considered: in the first step both are admissible and added to A; in the second and third step both are admissible, but one is not used since the backward neighbour is not in O; in the last step one forward neighbour is not admissible and the other is not used since the backward neighbour is not in O.

In [4] for each candidate index $\underline{k}$ from A one computes $\|f_{\mathsf{O} \cup \{\underline{k}\}} - f_{\mathsf{O}}\|_{\text{RLS}}$ to measure the contribution of this grid to the overall solution, i.e., the reduction in the functional (1) for the solution using the set $\mathsf{O} \cup \{\underline{k}\}$ in comparison to the current solution using O. Although $\|\cdot\|_{\text{RLS}}$ has to be computed in the additive space of all partial grids, its value can still be computed by using just the partial grids [3]. For the data dependent part one computes for each partial function its value on a given data point and adds these using the combination coefficients. The smoothing term of the discrete Laplacian can be expressed as a weighted sum over expressions of the form $\langle \nabla f_{\underline{i}}, \nabla f_{\underline{j}} \rangle$ which can be computed on-the-fly via a grid which includes both $\Omega_{\underline{i}}$ and $\Omega_{\underline{j}}$ [3]. This approach was shown to recover the ANOVA-decomposition for synthetic problems [4] but can be computationally expensive. This is due to the necessary recomputation of the error indicators for all remaining grids in A after a grid is added to O since the reference solution for the error indicator of the candidate grids changes in each adaptation step. If this recomputation does not take place the adaptive procedure tends to converge to less suitable solution or stops too early.

2.2 Localized adaptation criteria

In this paper we investigate two different error indicators, which are localized in some sense. First, we propose to use $\|f_{\{\underline{k}\}} - f_{\mathsf{B}(\underline{k})}\|_{\text{RLS}}$, the difference in the functional (1) between the solution of the candidate grid $\underline{k}$ added to A and the optimized combination solution of its backward neighbourhood $\mathsf{B}(\underline{k})$.

This value will not change during the computation since the backward neighbourhood $\mathsf{B}(\underline{k})$ will not change; candidate grids can only be added to A if all backward neighbours are already in I. This is a big reduction in the computational complexity in comparison to the original approach since no recomputation is necessary. But it is also has the potential for a good error indicator. The combined solution from its backward neighbours lives in the same (small) discrete function space as the candidate grid. If the solution on the new candidate grid shows no improvement in regard to the combined solution from its backward neighbours no large gain in the representation of the overall function can be expected.

Second, we propose an error indicator inspired by the residual estimator based on a hierarchical subspace decomposition used for the finite element approach, see e.g. [1]. For a candidate grid $\underline{k}$ we consider the set $\mathsf{O} \setminus \mathsf{B}(\underline{k})$, i.e. the old indices without the indices from the backward neighbourhood of $\underline{k}$. The solution $f_{\mathsf{O}\setminus\mathsf{B}(\underline{k})}$ for this index set is now computed according to (3) using the optimal coefficients (5) and considered as the reference solution. As an error indicator we new compare the difference in the solutions of the residual problems

$$\langle e_{\mathsf{I}}, f_{\underline{l}} \rangle_{\text{RLS}} = \frac{1}{M} \sum_{i=1}^{M} f_{\underline{l}}(\underline{x}_i) y_i - \langle f_{\mathsf{O}\setminus\mathsf{B}(\underline{k})}, f_{\underline{l}} \rangle_{\text{RLS}} \qquad \forall \underline{l} \in \mathsf{I}$$

for $\mathsf{I} = \mathsf{B}(\underline{k})$ and $\mathsf{I} = \mathsf{B}(\underline{k}) \cup \underline{k}$, that is we use $\|e_{\mathsf{B}(\underline{k})} - e_{\mathsf{B}(\underline{k})\cup\underline{k}}\|_{\text{RLS}}$ as an error indicator. It measures the *additional* improvement $\underline{k}$ can provide for the solution of the overall problem. This error indicator is localized, since it measures the improvement of $\underline{k}$ compared against its backward neighbourhood $\mathsf{B}(\underline{k})$, but it also takes the global problem into account since we compute the residual solutions, we measure the improvement in addition to the solution provided by $f_{\mathsf{O}\setminus\mathsf{B}(\underline{k})}$.

Solving for e_{I} amounts to nothing else as computing the optimal combination coefficients for the modified right hand side

$$L(g) := \frac{1}{M} \sum_{i=1}^{M} g(\underline{x}_i) y_i - \langle f_{\mathsf{O}\setminus\mathsf{B}(\underline{k})}, g \rangle_{\text{RLS}},$$

using a suitable numbering of I. Then $e_{\mathsf{I}} = \sum_{\underline{l}\in\mathsf{I}} c_{\underline{l}} f_{\underline{l}}(\underline{x})$ and one can compute the error indicator $\|e_{\mathsf{B}(\underline{k})} - e_{\mathsf{B}(\underline{k})\cup\underline{k}}\|_{\text{RLS}}$.

Both these error indicators can be viewed as a local criteria since only the difference between the candidate grid and its backward neighbours are

Algorithm 1: The dimension adaptive algorithm

```
compute partial problem for index −1
A := {−1}                                   ▷ active index set
O := ∅                                      ▷ old index set
while stopping criterion not fulfilled do
    choose k ∈ A with largest ε_k           ▷ largest indicator
    O := O ∪ {k}
    A := A\{k}
    for t = 1, ..., d do                    ▷ look at neighbours of k
        j := k + e_t
        if j − e_l ∈ O ∀ l = 1, ..., d then ▷ admissible
            A := A ∪ {j}
            compute partial problem for index j
            compute local error indicator ε_j
```

considered, we call them LOCAL CHANGE and LOCAL RESIDUAL, respectively. In comparison the original indicator, where all grids from O are taken into account, can be regarded as a global error indicator, we call it GLOBAL CHANGE in the following.

The overall procedure is sketched in Figure 1. Given the sets O and A the algorithm uses a greedy approach for the dimension adaptive grid choice. Depending on one of the above error indicators (and possibly other values such as the complexity of the computation for a partial solution) the grid in A which provides the highest benefit is chosen and added to the index set O. Its forward neighbourhood is searched for admissible grids to be added to A, for these the solution and error indicator are computed. Then the outer loop restarts and the procedure continues until a suitable global stopping criterion is reached; typically when the reduction of the residual falls under a given threshold. Note that for GLOBAL CHANGE additionally the error indicators for all $\underline{j} \in \mathsf{A}$ need to be recomputed after an index $\underline{k}$ is added to O.

Note that we start in the algorithm with the constant function of grid $\Omega_{\underline{-1}}$ (i.e., $\mathsf{A} := \{\underline{-1}\}$) and in the first step look at all grids which are linear in only one dimension, that is all $\Omega_{\underline{-1}+\underline{e}_j}$ with $j = 1, \ldots, d$. Once two of these one-dimensional grids were chosen in successive adaptive steps, the algorithm starts to branch out to grids which can involve two dimensions and later more. Since each partial grid is small and depends in its complexity not on d, the total number of dimensions, but q, the number of dimensions which are not treated as constant, it allows us to treat higher dimensional problems than before with the original combination technique. Furthermore, the information about which dimensions are refined and in which combinations attributes are used allows an interpretation of the combined solution, for example one can easily see which input dimensions are significant.

Table 1. Results for several real life data sets using the dimension adaptive combination technique with the three different error criteria. Given is the mean squared error (MSE) on the test data (with order of magnitude in the subscript), the used tolerance for stopping criteria, the number of grids in I, and the run times in seconds.

	GLOBAL CHANGE				LOCAL CHANGE				LOCAL RESIDUAL			
	tol	MSE	grid	time	tol	MSE	grid	time	tol	MSE	grid	time
census	5_{5}	7.85_{7}	5477	2796	2.5_{7}	6.61_{7}	5215	1413	2.5_{5}	4.36_{7}	9883	2463
cpu activity	1_{-2}	4.96	291	15	5_{-1}	5.27	2251	260	5_{-2}	5.30	1674	83
elevators	1_{-9}	6.37_{-6}	450	157	1_{-8}	8.48_{-6}	998	110	1_{-10}	5.98_{-6}	2674	1011
helicopter	5_{-9}	5.69_{-5}	1584	22565	7.5_{-5}	2.71_{-4}	909	8550	5_{-10}	3.74_{-5}	3573	6636
pole	1_{-2}	7.95_{1}	1785	7946	1_{-1}	9.88_{1}	3133	13690	5_{-3}	7.29_{1}	5016	2500

3 Numerical experiments

The following experiments were done on a machine with an Opteron (2.6 GHz) CPU. We measure the mean squared error $\text{MSE} = \frac{1}{M}\sum_{i=1}^{M}(f(\underline{x}_i) - y_i)^2$ and the normalized root mean square error $\text{NRMSE} = \sqrt{\text{MSE}}/(\max y_i)$.

Note that for GLOBAL CHANGE after each addition of an index $\underline{k}$ to O we only recompute the criteria for the 25 % of the indices $\underline{k} \in \mathsf{A}$ with the currently largest criteria, the values for all indices in A are only recomputed after every 10th add, this is to reduce the computational effort.

We consider several real life data sets:

1. Census housing[1] consists of 22,784 instances with 121 attributes.
2. Computer activity[2] consists of 8,192 instances in 21 dimensions.
3. Elevators[2] consists of 16,599 instances in 18 dimensions.
4. Helicopter flight project[3], with 13 attributes and 44,000 instances.
5. Pole[2] consists of 15,000 instances in 26 dimensions.

For the following experiments 90 % of each data set are used for training and 10 % for evaluation. We further use a 2:1 split of the training data to tune the parameters λ and the stopping criteria, i.e. learning on 2 parts and evaluating on 1 part. With the λ and tolerance resulting in the lowest MSE we then compute on all training data and evaluate on the before unseen test data. These are the results given in Table 1.

The first observation is that the two local criteria involve more grids, but the used run time does not increase as much, which was the aim. The GLOBAL CHANGE is good for data which do not need many grids for the representation. Overall the best performance achieved the LOCAL RESIDUAL criteria, while the LOCAL CHANGE criteria produced the worst results. Just using the localized

[1] Available at `http://www.cs.toronto.edu/~delve/data/census-house`
[2] Available at `http://www.liaad.up.pt/~ltorgo/Regression/`
[3] from Ng et.al., Autonomous inverted helicopter flight via reinforcement learning

Table 2. Comparison of our results using the GLOBAL CHANGE and LOCAL RESIDUAL criteria against results from [9], time is in seconds. The MSE results of the LOCAL RESIDUAL adaptive criteria scale accordingly to NRMSE.

	SVR		GLOBAL CHANGE		LOCAL RESIDUAL	
	NRMSE	time	NRMSE	time	NRMSE	time
census	> 0.015	> 400	0.017	2796	0.013	2463
cpu activity	> 0.04	> 100	0.022	15	0.023	83
elevators	> 0.08	> 200	0.043	157	0.042	1011
pole	> 0.09	> 100	0.089	7946	0.085	2500

information from the comparison of the solution from grid $\underline{k}$ against the one from its backward neighbours $\mathsf{B}(\underline{k})$ often results in the use of grids with large refinement per dimension and therefore large computational effort. The lack of information from the other grids in O also leads to early overfitting.

The run time depends on the number of grids, but also on the kind of grids which are being used. Grids which depend on a small number of dimensions but are highly refined, i.e. for a few but large entries in $\underline{k}$, are worse in this regard than grids which depend on more dimensions, but only have a small level, i.e. many, but small entries in $\underline{k}$.

For the considered data sets all dimensions were used at least in one grid, although the number of grids can depend largely for the different attributes. We observed up to 5 non-constant dimensions per grid. How often a dimension is used and the size of the error indicators for these grids are information about the importance of attributes and can be derived from the final results. If this information is worthwhile in practise needs to be investigated on real life data sets together with specialists from the application area.

Only on the helicopter data set with just 13 dimensions the non-adaptive optimized combination technique [3] could be used. It achieves a MSE of 3.26_{-5} in 18,280 seconds using level 3. Level 4 was not finished after 5 days.

Finally a comparison with results using CVR, a special form of support vector regression, is given in Table 2. For all data sets our method achieves better results, but might need more, in one case quite significant, run time. On the other hand, using a smaller tolerance a somewhat worse result could be achieved by our approach in less time. Note that for a larger synthetic data set a quite significant run time advantage of the dimension adaptive approach in comparison to CVR can be observed [3, 9].

4 Conclusions and Outlook

The dimension adaptive combination technique for regression shows good results in high dimensions and breaks the curse of dimensionality of grid based approaches. It gives a non-linear function describing the relationship between

predictor and response variables and (approximately) identifies the ANOVA-decomposition. Of the three different refinement criteria, GLOBAL CHANGE is best suited for applications using a small number of partial grids, otherwise LOCAL RESIDUAL performed best. It is known that error estimators which use the difference between two approximations of different resolution, i.e. of extrapolation type, have weaknesses. For example the error estimator can be small, although the actual error is still large [1]. Furthermore, the combination technique can also be derived as an extrapolation technique, therefore a thorough investigation of the observerd behaviour in this context is warranted.

We currently employ a simple greedy approach in the adaptive procedure. More sophisticated adaptation strategies and different error indicators, for example taking computational complexity of a grid into account, are worthwhile investigating, especially in regard to an underlying theory which could provide robustness and efficiency of the approach similar to the numerical solution of partial differential equations with adaptive finite elements [1].

The original approach scales linear in the number of data [2, 3]. In the dimension adaptive approach at least the computational effort for each partial grid scales linear in the number of data. Since the value of the adaption and stopping criteria depends on the number of data, the number of partial grids might change with a different number of data for a given stopping tolerance. Although we did not observe such unwanted behaviour in our experiments, it has to be seen if in a worst case scenario the dimension adaptive approach could result in a non-linear scaling in regard to the number of data.

References

1. Mark Ainsworth and J.Tinsley Oden. *A posteriori error estimation in finite element analysis*. Wiley, 2000.
2. J. Garcke, M. Griebel, and M. Thess. Data mining with sparse grids. *Computing*, 67(3):225–253, 2001.
3. Jochen Garcke. Regression with the optimised combination technique. In W. Cohen and A. Moore, editors, *23rd ICML '06*, pages 321–328, 2006.
4. Jochen Garcke. A dimension adaptive sparse grid combination technique for machine learning. In Wayne Read, Jay W. Larson, and A. J. Roberts, editors, *Proc. of 13th CTAC-2006*, volume 48 of *ANZIAM J.*, pages C725–C740, 2007.
5. T. Gerstner and M. Griebel. Dimension–Adaptive Tensor–Product Quadrature. *Computing*, 71(1):65–87, 2003.
6. M. Griebel, M. Schneider, and C. Zenger. A combination technique for the solution of sparse grid problems. In P. de Groen and R. Beauwens, editors, *Iterative Methods in Linear Algebra*, pages 263–281. IMACS, Elsevier, 1992.
7. M. Hegland. Adaptive sparse grids. In K. Burrage and Roger B. Sidje, editors, *Proc. of 10th CTAC-2001*, volume 44 of *ANZIAM J.*, pages C335–C353, 2003.
8. M. Hegland, J. Garcke, and V. Challis. The combination technique and some generalisations. *Linear Algebra and its Applications*, 420(2–3):249–275, 2007.
9. Ivor W. Tsang, James T. Kwok, and Kimo T. Lai. Core vector regression for very large regression problems. In Luc De Raedt and Stefan Wrobel, editors, *22nd ICML 2005*, pages 912–919. ACM, 2005.