Detecting and identifying ambiguities in regression problems: an approach using a modified mountain method

A.K. Nandi[A1], F. Klawonn[A2]

Department of Computer Science, University of Applied Sciences BS/WF

Salzdahlumer Str. 46/48, 38302, Wolfenbuettel, Germany

[A1] (Corresponding author) Email: nandiarup@yahoo.com

[A2]Email: f.klawonn@fh-wolfenbuettel.de

Abstract- Regression problems occur in many data analysis applications. The aim of regression is to approximate a function from which measurements were taken. When considering a regression problem, we have to take a number of aspects into account: How noisy the data are, whether they cover the domain sufficiently in which we want to find the regression function and what kind of regression function we should choose. However, the underlying assumption is always that the data actually are (noisy) samples of a single function. In some cases, this might not be true. For instance, when we consider data from a technical process that is controlled by human operators, these operators might use different strategies to reach a particular goal. Even a single operator might not stick to the same strategy all the time. Thus, the dataset containing a mixture of samples from different strategies, do not represent (noisy) samples from a single function. Therefore, there exists an ambiguity of selecting data from a large dataset for regression problems to fit a single model.

In this paper, we suggest an approach using a modified mountain method (MMM) to select data from a jumble of large data samples that come from different functions, in order to cope with the ambiguities in the underlying regression problem. The proposed method may also serve to identify the best local (approximation) function(s). These are determined using a weighted regression analysis method. The proposed methodology is explained with a one-dimensional problem, a single input single output system, and later performance of the proposed approach is analysed with artificial data of a two-dimensional case study.

Keywords: Liner regression; ambiguities; function approximation; modified mountain method.

I. Introduction

Regression refers to the problem of predicting the value of a continuous variable based on one or a number of other (independent) variables. In a regression problem (RP), a sample dataset consists of a number of measured (multiple) input (single) output pairs. There are a number of aspects that have to be taken into account to solve a regression problem. How should the regressions function be chosen in order to be flexible enough to fit to the data, but restricted enough to avoid overfitting? What is the appropriate error function? In most cases, the mean squared error is taken. However, there are a variety of other measures like the mean absolute error or the maximum absolute error (used for uniform approximation). Do the data cover the input domain sufficiently in the region where we want to carry out the prediction? Otherwise, we are in the area of extrapolation, which can lead to random guessing, if we have no further information about the underlying model that generates the data. How noisy are the data? Can we make sufficiently precise predictions in spite of the noise? Does the dataset contain outliers? How can we identify and remove them? However, one aspect is very often neglected or even ignored in the context of regression problems: The data might not represent (noisy) measurements of a single function, but (noisy) measurements from more than one function. This can happen for different reasons. When the data are taken from observing human operators controlling a process, there might be different strategies used by the same or different operators to achieve the same goal. As a simple example, consider steering a vehicle including obstacle avoidance. The easy and common method is, an obstacle can be avoided by either steering to the right or to the left. Data from both cases might be present in the data set. The best thing a regression function aiming at minimizing the mean (squared) error can do, is to interpolate between the two strategies which leads to bumping straight into the obstacle.

Another reason for the presence of such ambiguities in data sets might occur in the case, when there are attributes that contain important information that were not observed or not observable. Therefore, for the same values of the observed attributes we might have completely different values for the dependent variable due to different values in the non-observed attributes.

Another example of ambiguity lies in the fact of selecting the optimal values of the machining parameters in manufacturing processes to achieve a desired output such as surface roughness of a machined component.

For instance, in grinding processes (a manufacturing process to generate a smooth surface on a workpiece), the selection of optimal values of wheel speed to achieve a desired surface finish with different values of wheel dress lead is ambiguous in the operator's point of view for a given work speed and feed rate. Practically, different operators have used their own (expert) strategy to consider the optimal values of machining/process control parameters to perform a particular task. Again, there is also the possibility of real outliers due to inherent human or measurement errors, which cannot be avoided. Therefore, the existence of imperfect data in the whole data set as well as data collected from various sources; based on which the regression analysis is carried out to develop a single model, affect the model's performance. This is obvious because every data point involved in a particular dataset has an effect in determining the values of the regression function coefficients that are estimated using a regression analysis method.

2. Outline of the problem and its solution

Our principle aim is to approximate the whole (possibly ambiguous) data set by local models or local functions. In order to define a suitable local function, one should consider the following three primary aspects: determination of the coefficients of the function of a given structure, identification of the region where the function is applicable and the goodness of fit for a significant number of data samples. Due to the ambiguity inherent in the data, the local function might not fit to all the data in its assigned local region. However, those aspects are interrelated with each other. Let us consider a physical system and we have a set of data that are observed/measured during operation of the system with different strategies applied to obtain the same system target/goal. Thus, these data do not follow a single function/model rather follow different functions of the same system corresponding to the respective strategies. The main aim of this work is first to formulate the local function(s) of the respective strategies based on the available sample data in a specified region and finally to identify as well measure the ambiguity (number of strategies associated with the system) involved in the regression procedure used to construct the function(s). A simple one-dimensional case is illustrated in Fig. 1, where x is the input variable and y is the output variable. In this problem, the data samples as depicted in Fig. 1 are originally taken from three different local functions. Now, to define a local function using a regression

analysis based on the entire data samples or some of the arbitrary data samples may not be able to satisfy the above-mentioned criteria of a good (local) function. This incident is obvious because, first, each used data sample has a contribution in determining the coefficients of the chosen regression function, secondly, the distance or the physical location of a data sample from the derived local function has also a great influence, and these factors finally affect the computation of the optimal values of the function coefficients. However, there must exist a suitable approximate function that fulfils the aspects of a best local function with a significant number of data samples. This phenomenon particularly may occur when the data samples belong to more than one underlying function. Thus, our primary objective is to develop an algorithm that will be able to find a (near) best function through an extensive local search over the data samples. This function should cover as many data as possible in a local region that is as large as possible. One possible method is to proceed in finding the local function(s) step by step in a typical manner such that, once a function is derived, the associated data samples are removed from the whole data set, and finally to measure the ambiguities in terms of the number of underlying functions involved in the whole conglomeration of data samples.

In the authors' previous work [11], an algorithm was proposed to detect ambiguities in regression problems using two TSK models. TSK models are fuzzy rule-based descriptions of functions. Each rule defines a local (fuzzy) region and assigns an output function to that region. In that work, two TSK fuzzy models are considered to work in parallel based upon the idea of sharing or competing for the data with each other in every step of the iteration process. This idea of data competing or sharing is implemented following the concept of alternating optimization in clustering. However, the underlying TSK models are quite static and cannot be fitted well to the data in all cases, since the underlying fuzzy partitions (defining the local regions) are assumed to be fixed. In the present work, we are interested in detecting as well as obtaining the approximate function(s) in the mixture of data samples that are considered to be present in the whole input space or a single (fuzzy) partition of the input space by means of a locally extensive search method such as the (modified) mountain method. The mountain method is a clustering technique that is suitable for finding (approximate) cluster centres in a data set step by step. After a cluster centre is found (in a region with high data density), data associated with this cluster centre are (partly) subtracted from the data set. We face a

similar, but more complicated problem here. Instead of finding simple cluster centres, we are looking for appropriate local functions. In addition, a local function does not necessarily have to approximate all the data in its assigned region, because of the inherent ambiguities in our data.

The rest of the paper is arranged as follows: Section 3 briefly reviews the general mountain method. The Modified Mountain Method (MMM) and its enrichment with linear regression based on distance weights is discussed in Section 4.The proposed algorithm based on the modified mountain method to detect and identify ambiguities in regression problems is explained in Section 5. Section 6 investigates the performance of the proposed MMM-based approach with a two-dimensional case study. The concluding remarks of this work are presented in Section 7.

3. The general mountain method

The clustering concept of a mountain method was first introduced by Barone et. al. in 1993 [1]. Later, Yager had extended its utilization in different applications such as approximate estimation of cluster centres [13], fuzzy clustering [2] and fuzzy rule generation [14]. The basic idea of the mountain approach is do identify regions of high data density step by step, cover the data in that region by a cluster centre and (partly) remove the covered data – depending on how well they are covered – from the data set. In order to identify regions of high data density, the mountain method defines a kind of potential function. Then the mountain method searches for a candidate for a cluster centre, which has a higher potential than the others. The (potential) surface generated by the potential values of each candidate forms a mountain-like range and the highest peak of this mountain range is the highest potential candidate. In the scheme of Yager's mountain method applied for data clustering, the algorithm starts with the restriction of an s dimensional data space Rs to an s dimensional hypercube I_1 x I_2 x…x I_s, where the length of each dimension of the dataset I_j, j=1, 2,…, s is defined by the ranges of the corresponding coordinates x_{kj} (the j_{th} coordinate of the k_{th} data point). Then, each interval (I_j) is discretized into multiple equidistant points in order to define a number of grid nodes in the data space restricted by the hypercube. To each grid node (N_i) a value is assigned, the so-called mountain potential

that is obtained based on a function, called mountain function. The mountain function is defined by the following empirical expression.

$$M\left(N_i\right) = \sum_{k=1}^{n} e^{-\alpha d\left(x_k, N_i\right)}, \tag{1}$$

where α is a positive constant and $d\left(x_k, N_i\right)$ is the Euclidean distance (or any other suitable distance measure) between the data point x_k and grid node N_i. The grid node that has the maximum value of the mountain function is considered to be the peak of the mountain range and chosen as the cluster centre in clustering of data i.e.,

$$M_1^* = Max\left[M\left(N_i\right)\right], \tag{2}$$

After the identification of this grid node, say N_j, as the first cluster centre, the mountain potentials of all the grid nodes are re-evaluated by eliminating the effects of the cluster centre that has just been identified previously. In order to accomplish this, a certain amount from the prior mountain potential at each grid node is subtracted, depending on the grid node's distance to N_j. Besides this, the amount to be deducted from the mountain potential also depends on the current maximal mountain potential value, M_1*. Thus, the modified values, $M_2(N_i)$ of node N_i after the k_{th} iteration are defined by

$$M^k\left(N_i\right) = M^{k-1}\left(N_i\right) - M_{k-1}^* \sum_{k=1}^{n} e^{-\beta d\left(N_{k-1}^*, N_i\right)} \tag{3}$$

where β is another positive constant. Moreover, it should be mentioned that Chiu [4] has suggested a method called subtractive clustering method (SCM), which proceeds in a way similar to that of Yager's mountain method. The only difference between SCM and the mountain method is that, in SCM each data point is considered as a potential cluster centre instead of grid points as considered in Yager's mountain method. Kelly and Yager proposed a modified mountain method in 1995 for determining clusters centres [10]. Furthermore, Pal and Chakraborty have suggested a generalization of the mountain method (called mountain circular shell (MCS) method) by extending its application to detect special types of clusters such as circular, elliptical and rectangular shells [12]. The MCS method includes a modification the mountain function by incorporating the

radius and centre of circular shell. However, the applications of the mountain method as found in the past literature were mainly used to search and find approximate cluster centres of general data samples. The great advantage of using the mountain method is that the number of clusters is determined automatically in contrast to strategies such as the fuzzy C-means clustering algorithm [3, 8].

The mountain method as described above is an extremely useful technique to find a single local best fit (a cluster centre) for data using an extensive search strategy. However, as we are going to detect the number of underlying functions of a mixture of data samples as well as to find approximations of those functions simultaneously, the above (general) mountain method is not suited to solve this kind of problem in a straightforward manner. For example, the general mountain method does not consider deviation(s) of output value(s) for different sets of input(s) in determining the mountain potential. Thus, in order to tackle such kind of problems, a modification of the original mountain method, called the modified mountain method (MMM) is suggested here, which incorporates some additional features related to linear regression to find the best approximate local function.

4. Modified mountain method and its enrichment with weighted linear regression

4.1 Modified mountain method

The main difference of the modified mountain method (MMM) to the general mountain method (as described in Section 3) is that, at each iteration step, unlike the re-evaluation of the peaks of the mountain range by diminishing the effect of the highest peak that is just identified, here the mountain peaks are newly constructed followed by fully destructing the mountain range. Furthermore, in MMM, the mountain function is represented in a different way, which includes some important features of generalized linear regression such as the number of selected data samples and the error distance measure of each selected data point. The selection of the highest mountain peak is made based on the maximum mountain function value (mountain potential). The MMM that is suggested here, is used to identify the data samples from a large number of mixed data that are collected from different sources (strategies or functions), in order to detect and cope with ambiguities in the regression problem. In order to detect the ambiguities, MMM helps to formulate the best local function (or to fit a local function to (selected) data samples) in a large amount of mixed data that are present in a specified local region.

The local function is constructed using a distance weighted linear regression method around a grid node based on an inner product norm of the selected data points. In order to accomplish this we suggest an algorithm for data selection in a local sense.

In principle, the modified mountain method is works as follows.

Let us consider a set of n input-output data samples $\{(x_1^{1,\ldots,s}, y_1), (x_2^{1,\ldots,s}, y_2), \ldots\ldots, (x_n^{1,\ldots,s}, y_n)\}$ originating from various functions, sources or strategies having s input variables (x^s) and a single output variable (y). Thus we have a set of input data points $\{x_1, x_2, \ldots\ldots, x_n\}$ in the s dimensional space R^s. The value x_{kj} is defined by the j^{th} coordinate of the k^{th} input data point where k=1, 2, …., n and j=1, 2,…, s. We assume that the input data are located in a j dimensional input hyperbox I_1 x I_2 x..x I_s of a specified region where the intervals I_j (j=1, ……., s) are defined by the ranges of the coordinates of the inputs x_{kj}, which are defined by the (nearest rounded) values of [$\min_k(x_{kj})$, (and) $\max_k(x_{kj})$]. Now, let us consider s dimensional grid nodes $N(i_1, i_2, \ldots\ldots, i_S)$ by discretetizing each of the intervals I_j into r_j (not necessarily) equidistant (grid) points, where the indices i_1, ….., i_s take values from the sets [1, …, r_1], ……, [1, …, r_s]. The space restricted by the hyperbox is discretized by forming the grid nodes, $N(i_1, i_2, \ldots\ldots, i_S)$, which are theoretically defined by the every point in the hyperbox. The primary function of these grid nodes is that they are considered as possible mountains peaks based upon which the local functions are to be identified.

Now, we consider a single grid node that might be a mountain peak and evaluate the mountain function value that will be discussed in the following. We will use the simpler notation of N_i to indicate a node with the implicit fact that i equals some tuple of the form (i_1, ….., i_s). To determine the mountain peak value on the grid node, we propose an expression, which we call modified mountain function. The proposed modified mountain function is defined as follows. We incorporate three basic terms in the modified mountain function. First of all, we have to choose those data points, which the corresponding local function should approximate. This number of data should be as large as possible. Secondly, since the local approximation should take place around our considered grid node, the chosen data should be close to this grid node. Finally, the associated output values

for these data should be approximated as good as possible by the considered local function. Therefore, we define the modified mountain function by

$$Mv(N_i) = c_1 D_p + c_2 \sum_{i=1}^{D_p} e^{-\sigma d(N_k, x_i)} - c_3 \sum_{i=1}^{D_p} Ep_i e^{-\sigma d(N_k, x_i)}, \quad (4)$$

where c_1, c_2 and c_3 are positive parameters that are used to balance the values of the corresponding associated terms, the positive constant σ is used to influence the value of the Euclidian distances $d(N_k, x_i)$ between the node N_k and the data points x_i, D_p is the number of selected data points that are to be approximated by the local function and E_{pi} is the absolute error of the local function at the i^{th} data point. In order to determine an optimal value of M_v, the selected values of the positive parameters (c_1, c_2 and c_3) should be chosen properly. We can look at the value of the function $M_v(N_i)$ as the height of a mountain range. The higher the mountain function value the larger is this potential ability to approximate a considerable number of points well enough. Thus, the mountain function value can be used as an indicator of the best local function approximation for a (large) subset of the data.

In contrast to the standard mountain method, we face additional problems here. The standard mountain method evaluates the grid nodes and then decides which grid node yields the best local fit for the data. In our case, we have to specify more parameters than only the grid node for the evaluation. For each grid node, we have to define a local regression function and we have to choose the subset of the data for which the corresponding local model is responsible. In the following subsection, we first tackle the problem of finding a suitable local approximation based on standard regression.

4. 2 Linear regression based on distance weights

In this section, we consider a method to determine the function coefficients using linear regression based on distance weights on the data samples. There is a great advantage of using distance weight-based regression over the general regression method. In the general regression method, all data points have the same importance for the function coefficients, while in distance weight-based regression the data points with a lower distance value have a higher importance in finding the coefficients of the functions around the respective nodes. In

order to derive a set of linear equations for evaluation of the function coefficients, let us assume we have a data set containing s input-output tuples of sample data where the output $y^{(i)}$ is assigned to the input $\left(x_1^{(i)}, x_2^{(i)}, \ldots\ldots\ldots, x_n^{(i)}\right)$

$$D = \left\{\left(x_1^{(1)}, \ldots\ldots, x_n^{(1)}, y^{(1)}\right), \left(x_1^{(2)}, \ldots\ldots, x_n^{(2)}, y^{(2)}\right) \ldots\ldots\ldots \left(x_1^{(s)}, \ldots\ldots, x_n^{(s)}, y^{(s)}\right)\right\}$$

Now, assume we have a polynomial function in the following form:

$$y = f\left(x_1^l, x_2^l, \ldots\ldots\ldots, x_n^l\right)$$

$$= \sum_{j=1}^{K} a_j f_j\left(x_1^l, x_2^l, \ldots\ldots x_n^l\right)$$

Let us assume, the total quadratic error that is caused by the function with respect to the given dataset is:

$$E\left(a_1, a_2, \ldots\ldots\ldots, a_k\right)$$

$$= \sum_{l=1}^{s} w_l \left(f\left(x_1^{(l)}, x_2^{(l)}, \ldots . x_n^{(l)}\right) - y^{(l)}\right)^2$$

$$= \sum_{l=1}^{S} w_l \left(a_1 f_l\left(x_1^l, x_2^l, \ldots, x_n^l\right) + \ldots + a_k f_1\left(x_1^l, x_2^l, \ldots, x_n^l\right) - y^l\right)^2, \qquad (5)$$

where w_l is a positive parameter called distance weight factor, which is defined by $e^{-\delta d\left(N, x_l\right)}$ and δ is a positive constant. The term $d\left(N, x_l\right)$ is defined as the distance (more specifically Euclidian distance or any other suitable distance measure) of data point x_l to node N. To determine the suitable values of the parameters $\left(a_1, a_2, \ldots\ldots, a_k\right)$ by minimizing E, the partial derivatives of E with respect to each parameter are set to zero and finally we obtain Equation (3) in the following form:

$$\sum_{j=1}^{K} a_j \sum_{l=1}^{S} w_l f_j\left(x_1^l, x_2^l, \ldots, x_n^l\right) \cdot f_t\left(x_1^l, x_2^l, \ldots, x_n^l\right) = \sum_{l=1}^{S} w_l y^l f_t\left(x_1^l, x_2^l, \ldots, x_n^l\right) \qquad (6)$$

Thus, Equation (6) provides a set of linear equations from which we can compute the coefficients $(a_1, a_2, \ldots, a_k)$. The solution of Equation (6) in order to find the values of the coefficients can be solved using any conventional numerical method for systems of linear equations.

5. Detecting and identifying ambiguities in RP using MMM

In this section, we show how MMM can be used in a procedure for detecting ambiguities in regression problems. In order to detect the ambiguities in RP by selecting the (near) optimal local functions and the associated data samples in the mixture of data samples using the modified mountain method, we are systematically looking for global maxima of the mountain function value (mountain potential) at any point in the s-dimensional input space I_j (j=1, ……., s) i.e., the highest peak of the mountain range and the associated derived local function. Since, it is computationally impossible to consider all points as possible mountain peak in the input space, we restrict ourselves – as in the usual mountain method - to a finite number of grid points (nodes) spread over the input space. Let us first consider any of the grid nodes of the input hyperbox. Now the aim is to construct a local function/model around the grid node. There are three primary criteria to be considered in order to construct a local function/model in the input hyperbox of muddled data samples: function coefficient(s), locality of the model by means of its centre point (i.e. the grid node) and an suitable subset of data with significant number of data points. The coefficient(s) of the function are determined using the distance-weighted linear regression method as discussed in sub-section 4.2 based on the data samples in the suitable subset of muddled data. The locality of the model is characterised/sensed by the grid node around which the model is to be extended. To select the appropriate data samples from the mixture of data, there are well-known search methods such as genetic algorithm (GA) [7], simulated annealing [9], tabu search [5, 6]. However due to reasonably high computational complexity as well as processing time required for those method to obtain a sufficiently good solution, in this present work, an intuitive algorithm is proposed which is explained below. In this algorithm, the local function/model derived at its centre point (grid node) based on the input-output data samples those are well suited to the local function according to a permitted (absolute) error

limit in an iteration procedure. This algorithm is supposed to have less computational complexity to solve this specific problem in an efficient manner.

In the first step of the algorithm, a local function is constructed using Equation (6) based on all (initial) input-output data samples that might originate from different underlying strategies/functions. Then, for each input of the total data samples the output of the (tentatively) derived local function is calculated. Now, the deviation of the output value of the derived function from that of the desired one is considered as an error. Then the absolute (or any other error measuring function) error value to achieve a good approximation, is compared with a given desired error limit. Those data yielding a smaller error than the prescribed error limit are assigned to the corresponding local function and called them "selected data" (SD). In the next step, these selected data are used for linear regression (named as DR) to derive a new local function. Now we check for which data from the whole data set this function satisfies the prescribed error limit. If this consists of the same data samples as in the previous step, we terminate the computation of the local function at the corresponding grid node. Otherwise, we continue the procedure with the data samples assigned to new local function.

Once the local function is constructed, the mountain function value of the grid node is calculated using the modified mountain function (as expressed in Equation (4)). Now, the selection of the grid node (mountain peak) and the associated function are made based on the maximum modified mountain function value. If there is more than one grid node having the same maximum modified mountain function values, we select randomly any one of them as the highest peak of the mountain range. Let us denote the maximum value of the modified mountain function as Mv_1^* thus $Mv_1^*=\text{Max}_i[M(N_i)]$.

Let N_1^* indicate the grid node in the input-space where this maximal score of the mountain function value is attained, it is the highest peak of the mountain range formed by the data set available in the specified input hyperbox (region). We select the data points associated with the corresponding local function associated to this node. The function associated to the node N_1^* is considered as one of the local models from which the data samples are actually collected.

Once the first local function is identified, the data associated with this function or based on which the function is derived are separated from the total data set. It is essentially important to mention that, in case, if

two local functions yield exactly the same output for a (data) input value, the algorithm assigns the corresponding data sample randomly to any of the functions. Sometimes, the function that is first selected will be preferred to accommodate the common data sample between multiple underlying functions. Unlike to the destruction of the mountain peak after choosing the first cluster centre that is done in the general mountain method proposed by Yager and Filev, in the modified mountain method the mountain peaks are completely destroyed and we newly rebuild the mountains based on the rest of the data samples in the next iteration stage. The above procedure as is performed above in deriving/selecting the first function (strategy) is executed in the next iteration to find the second function that is generated using the remaining data samples. In this way, this iteration procedure is allowed to continue to find the different underlying functions until the number of remaining data sample(s) reaches a predefined minimum number (or zero when all the data samples are perfectly suited to the different functions according to the specified error level). The minimum number of data samples is prescribed in the sense that a very small number of data sample(s) is not sufficient in order to construct a function using the regression analysis method. There is also another possibility in terminating the iteration process, i.e. when no data are assigned to a local function based on the predefined error limit. The number of iterations is equal to the number of underlying local functions from where the data are actually generated. The data samples that are not fitted by any of the underlying functions (due to the high noise) are identified and we may call them outliers. It is important to mention that this proposed algorithm determines the function coefficients based on the type of the structure of the function assumed by the designer.

We demonstrate the working principle of the proposed approach, which is based on the modified mountain method to detect ambiguities in RP in the following example with artificial data. For reasons of simplicity, we consider a one-dimensional problem. Later in Section 6, we discuss the performance of the MMM-based approach with a two-dimensional case study. We analyse some of the important aspects of the algorithm that are responsible for the detection of ambiguities in RP and the estimation of function coefficients of the derived local function using the distance-weighted linear regression method.

Example

In this example, we consider a one-dimensional problem, which consists of one input and one output variable. Furthermore, in order to demonstrate the working procedure of the algorithm, we have considered a dataset, which is the mixture of data samples that are taken from three (pre-defined) different functions as follows.

$$y_1 = 14 + 0.5x \tag{7}$$

$$y_2 = -2 + x \tag{8}$$

$$y_3 = 50 - 1.5x \tag{9}$$

For clarity as well as simplicity in describing the working procedure of the algorithm, only 31 data samples (15 from Equation (7); 10 from Equation (8) and 6 from Equation (9)) are considered. These functions are plotted in Fig 1. Now, if one tries to fit a single function to all these (muddled) data points using a simple linear regression technique, the results of the obtained function will show erroneousity in terms of the output accuracy. Thus, our main objective is first to detect this ambiguity that the data samples actually belong to multiple functions (strategies) and then to find the number of local functions from which the data samples were actually collected as well as to determine a suitable approximate function for each of those strategies using a weighted LR approach.

We have considered the grid for discretization in the input space with r_1=10, as shown in Fig 1. The upper and lower limits of the interval I_1 as determined from Fig. 1 are rounded to the values of 0 and 30, respectively. The values of the positive parameters of Equation (4) c_1, c_2 and c_3 are set to the values of 1.0, 2.0 and 100, respectively. The value of the positive parameter δ that is used for finding the values of the distance weight factors during regression process is chosen as 1.0. For simplicity, the distance function for both the weight factor as well as in determining the mountain potential (Equation (4)) considered here, is defined by $\sqrt{(N - x_i)^2}$, N is the grid node and x_i an input data point. We have considered a function of the form $y = a_1 x + a_2 x^2$ for the linear regression process. The values of the coefficients (a_1 and a_2) are determined by the method of linear regression based on distance weight based on the data samples. We have set the desired

error limit (absolute value) to 0.075. The numerical values of the mountain function calculated using Equation (4) with $\sigma = 5.4$ in the first iteration are listed in Table 1. Table 2 and Table 3 show the mountain function values at the respective nodes in the successive iteration processes. In Table 1, it can been seen that node (4) has the highest mountain potential value 10.3309 corresponding to the local function with coefficient values of a_1= -2 and a_2 =1. In the second iteration, the maximum mountain potential (17.2995) is found at node (3) as shown in Table 2, which obtains the local function with coefficients (a_1 = 14 and a_2 = -0.5). Similarly, node (10) having the highest mountain value (7.0004) is detected in the third iteration step and the obtained local function has the coefficients values 50 and –1.5 for a_1 and a_2, respectively. In each node (4, 3 and 10) in the respective iteration steps all the corresponding data points (10 for function (11), 16 for function (10) and 5 for function (12) in the first, second and third iteration step, respectively) are selected, since these data samples are completely noiseless to ensure the prescribed permissible error limit (0.075). The common data point (x=18; y=23) between the two functions (Equation (7) and Equation (9)) is selected by a single local function i.e., Equation (9), which is first derived in the iteration procedure. Thus, the number of data samples selected by the local model at grid node (3) in Table 2 is shown as 16, whereas the local model derived at grid node (10) in Table 3 selects 5 instead of the original number of data samples (6). It has also been found that the mountain potential of some nodes (node 9 in Table 1 and node 7 in Table 2) show zero values. This is because in those cases where no data point were selected by the algorithm (as presented in Fig. 2) for the distance-weighted LR method.

The performance of the whole algorithm mainly depends on the appropriate values of the five positive parameters of the algorithm and their combination as well. The influence of these parameters on the performance of the MMM-based algorithm is discussed and analysed in Section 6.

6. Performance analysis of the MMM-based approach

In the above example (as discussed in Section 5), we have considered data from three different one-dimensional local functions and run the program with these data. In order to find a satisfactory result to determine the number of local functions that are involved in the data samples, we have chosen the values of the

positive constants c_1=1.0; c_2=2.0; c_3=100; $\delta = 1.0$ and $\sigma = 5.4$. The utilization of the positive parameters c_1, c_2 and c_3 is to have a balanced value of the modified mountain function potential. The positive constants σ and δ influence the values of the Euclidian distances $d(N_k, x_i)$ between the node, N_k and the data points, x_i in the expressions of the modified mountain function and the weighted linear regression, respectively. Thus, one of the most important problems was to find the appropriate value of the parameter δ as the precision of function approximation depends on this parameter. On the other hand, the role of the parameter σ is to maintain a reasonable amount of the modified mountain function value. Another issue is that the accuracy of the MMM-based algorithm depends also on the fitness of the grid nodes (i.e. the number of grid nodes assumed over the input space), but with a refined grid, the algorithm becomes computationally more expensive. Furthermore, with the increase in the dimensionality of the data the computational costs also increase.

We have analysed the performance of the proposed algorithm (as described in Section 5), which is combined with the MMM to detect ambiguities by finding the number of local functions actually involved in the dataset with different (finite) numbers of grid nodes chosen in the input space as well as the desired accuracy level that is to be set beforehand. In order to investigate the effect of those parameters, we have considered a two-dimensional problem, which actually involves four local functions as follows:

$$Y = 0.2\,x_1 + 0.3\,x_2 + 0.2\,x_1 x_2 \quad , \tag{10}$$

$$Y = 0.3\,x_1 + 0.3\,x_2 + 0.15\,x_1 x_2 \tag{11}$$

$$Y = 0.5\,x_1 + 0.2\,x_2 + 0.1\,x_1 x_2 \tag{12}$$

$$Y = 0.2\,x_1 + 0.2\,x_2 + 0.15\,x_1 x_2 \quad , \tag{13}$$

We incorporate a noisiness of 2 percent of the output value to the 40 data samples that are generated from each of the four different functions. The purpose of introducing noisiness to the data samples is that to analyse the capability of the proposed method how it deals with noisy data. Practically, the observed/measured data of a physical process are always contaminated by noise due to various factors. We have added the noisiness to the data samples as either in positive or negative sense, at random. The mixture of all data samples for the 40

different cases of each of the four functions, Equation (13), (14), (15) and (16) are depicted in Fig. 3. We have chosen the same functional form as the underlying functions, i.e., $Y = a_1 x_1 + a_2 x_2 + a_3 x_1 x_2$, where a1, a2 and a3 are the coefficients that are to be determined and, x1 and x2 are two input variables. Furthermore, we have assumed the values of the positive parameters (c_1, c_2 c_3 and σ) are the same as stated above. The influences of those parameters do not have much importance to the primary objective of the proposed MMM-based approach. Thus, in this study, the investigation is mainly made on the effects of the factor associated with the distance function in the weighted regression analysis δ, the number of grid nodes of the input space/hyperbox and the desired error limit.

To analyse the effect of the density of the grid nodes with the performance of the approach, the value of factor, δ is assumed 1.0. The value of the desired error level is set to 0.5. The values that are presented in Table 4 show the results obtained by the approach with various numbers of grid nodes in the input space. From Table 4, we find that detection of ambiguity or/otherwise the number functions does not depend much on the number of grid nodes particularly, when the proper value of δ is chosen for a given error limit but, the number of outliers reduces with increasing the grid nodes. Besides this, the number of outliers may be high in the case of more noisiness involved to the data samples while precise function accuracy is desired. However, with a higher number of grid nodes, a better result in finding the coefficient values of the derived local functions is obtained. That means, the underlying local models (functions), which are characterised by the function coefficients, belong to the different local strategies are evaluated more accurately for a given accuracy level to those functions. Nevertheless, with increasing the number of grid nodes, the program takes more computational time to evaluate the function coefficients. Particularly, a reasonable number of grid nodes is sufficient to identify the number of underlying local functions involved in the data set.

The results of the number of obtained functions along with the outliers for different values of the error limit for a number of grid nodes of 10x10=100 are depicted in Fig 4. Fig. 4 shows that a higher accuracy (less error) value generates more functions from a given set of data samples. However, the values of the function coefficients are also more accurately obtained with decreasing the given error limit. On the other hand, the

number of outliers is also reduced with a higher error value. Thus, the setting of the suitable value of error level is an important issue, which may provide the real measure (number of underlying local functions) of ambiguity involved in the data set. The optimum setting of error level should be chosen where the number of selected local models/functions would be more with a lowest number of outliers. However, the setting of proper error limit highly depends on the degree of noisiness associated to data samples.

The values of the distance weight factors that depended on the chosen value of positive parameter δ and the nature (or form) of the distance function considered influence the data samples associated with the local function during the regression analysis procedure. Fig. 5 describes the variation of the number of functions obtained and the outliers at the end of the program run with different values of δ for a given error limit of 0.5. Furthermore, which underlying function (whose function coefficients are determined using the distance-weighted LR method as discussed in sub-section 4.2) would be selected first primarily depends on the values of δ. Basically, the parameter δ helps in constructing a function around a node by providing an importance to the nearest data points to that node. However, the optimum value of σ is an important issue for which one can obtain the best result. The influences of the other parameters (c_1, c_2, c_3 and σ) are not very important for the main objective of this approach to detect ambiguities in regression problems. However, appropriate values of those parameters are significant to evaluate a rational mountain function value as well as to obtain better-approximated function.

However, one of the limitations of this proposed method is that the given accuracy level (error limit) prescribed to the algorithm (as illustrated in Fig.2) should not be higher than a certain value. This numerical value of the accuracy level is nothing but the minimum of the output-errors of the data sample calculated by the derived local function that is derived using the distance-weighted LR method based on the same data samples. Furthermore, in providing a lower desired error limit, it may happen that very similar data samples that originally belong to different local functions (strategies) are selected for a common function. This is because those data samples provide a lower error than the given threshold error level. These problems may be solved by modifying the algorithm in such a way that the program can start the iteration process with a

sufficiently low (crude) initial error level and gradually reduces the error value, and finally converges to an optimum value.

7. Conclusions

In this paper, we discuss different possible aspects with some practical examples, which may create ambiguities in regression problems when dealing with multi-strategic (multi-functional) data samples. The primary objective of this work is to identify the underlying local functions using a linear regression method in a mixture of data samples. In order to do this, we propose a novel approach to formulate as well as extract the local functions associated to the data set step-by-step in an iteration process. The proposed method approximates the near-best local model(s) by considering the three primary criteria such as determination of coefficients of the function of a given structure, identification of the region where the function is extended and the goodness of fit with significant number of data samples. In order to construct the local function(s) we have used here a weighted linear regression method. The number of step(s) needed in the iteration process determines the number of local underlying models/functions (identification of ambiguities) present to the whole data set. This proposed approach is accompanied with a method called the modified mountain method (whose working procedure is similar to the Yager's mountain method). The modified mountain method proposed here includes some important features of generalized linear regression techniques such as the error distance measure of each selected data point and the number of well-fitted data samples.

We demonstrate the proposed approach with artificial data taken from three dissimilar single input-single output functions to proof the feasibility of the method in order to detect and identify ambiguities in regression problems. Later, a further investigation is made to identify the possible basis to improve the performance of the method by analysing the effects of the involved parameters. In order to accomplish this we consider data samples taken from four different two inputs-one output systems. It has been observed that the performance of the proposed method mainly depends on three factors namely the prescribed error limit, the number of grid nodes assumed in the input space and the value of the distance weight factor used in the linear regression process. Among these three primary factors, the prescribed value of the error limit is most significant.

Acknowledgement

Authors are thankful to the support given by DAAD (German Academic Exchange Service) to carry out this research work.

References

[1] J.M. Barone, D.P. Filev, and R.R. Yager, Exponential possibility distributions and the mountain method, Proceedings of the 12th Annual NAFIPS Meeting, Allentown, PA, 1993, pp. 118-122.

[2] J. Barone, D.P. Filev and R.R. Yager, Mountain method-based fuzzy clustering: Methodological considerations, International Journal of General Systems 23 (1995), 281-305.

[3] J. Bezdek, R. Hathaway, M. Sabin and W. Tucker, Convergence theory for Fuzzy C-Means: counterexamples and repairs, IEEE Trans. on Systems, Man & Cybernetics-Part B, 17 (5) (1987), 873-877.

[4] S.L. Chiu, Extracting fuzzy rules for pattern classification by cluster estimation, Proceedings of the Sixth International Fuzzy Systems Association World Congress (IFSA'95), Sao Paulo, Brazil, July 1995, pp. 1-4.

[5] F. Glover, Tabu search-Part I, ORSA Journal Computer, 1 (1989), 190-206.

[6] F. Glover, Tabu search-Part II, ORSA Journal Computer, 2 (1990), 4-32.

[7] D.E. Goldberg, Genetic algorithms in search, optimization, and machine learning, Addison-Wesley, Reading, MA, USA, 1989.

[8] A. Kandel, Fuzzy techniques in pattern recognition, Wiley-Inter science: New York, 1982.

[9] S. Kirpatrick, C.D. Gelatt, and M.P. Vecchi, Optimization by simulated annealing, Science, 220 (1983), 671-680.

[10] J.A. Kelly and R.R. Yager, Determining clusters' fine center, extents and membership functions using a modified mountain method, International Symposium on Information Processing; Applications of Fuzzy Logic Technology II, Orlando, Proc. SPIE Vol. 2493 (1995), 347-358.

[11] A. Nandi and F. Klawonn, Detecting ambiguities in regression problems using TSK models, IEEE International conference on Fuzzy Systems (FUZZ-IEEE 2004), Budapest, Hungary, 26-29 July, 2004.

[12] N.R. Pal and D. Chakraborty, Mountain and subtractive clustering method: improvements and generalizations, Int. Journal of Intelligent Systems, 15 (2000), 329-341.

[13] R.R. Yager, and D.P. Filev, Approximating clustering via the mountain method, IEEE Trans on Systems, Man and Cybernetics, 24 (8) (1994), 1279-1284.

[14] R.R. Yager, and D.P. Filev, Generation of fuzzy rules by mountain clustering, Journal of Intelligent and Fuzzy Systems, 2 (1994), 209-219.

List of Figures:

List of Tables:

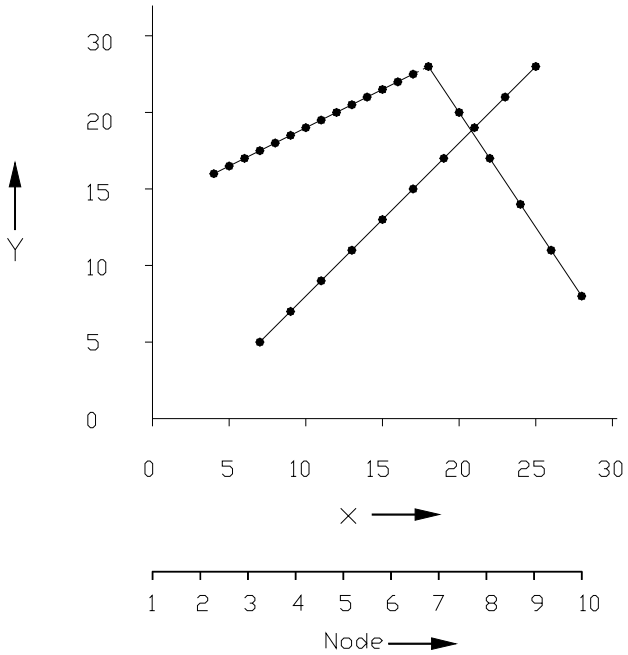

Fig. 1. Multi-functional given data samples

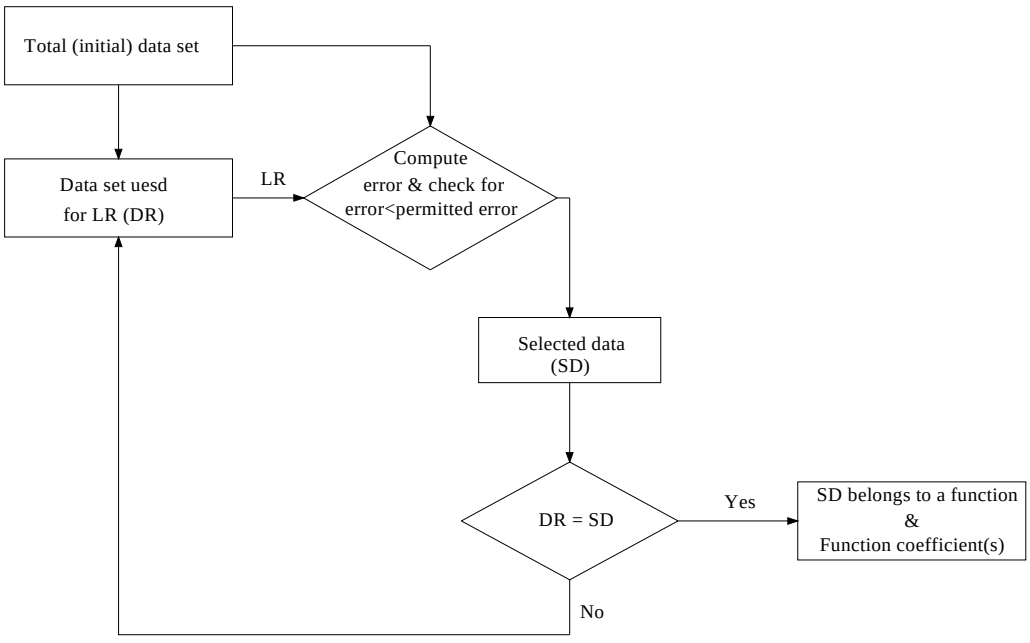

Fig. 2. Algorithm in finding a best approximating function from data using weighted linear regression method

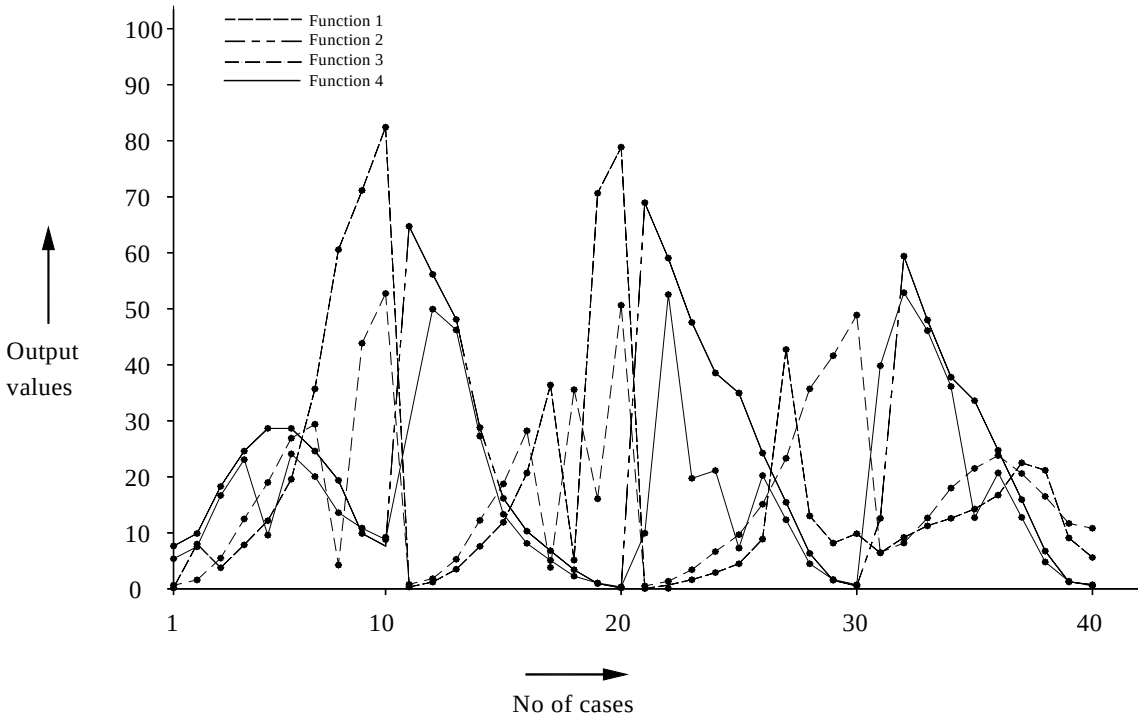

Fig. 3. Data samples for a two-dimensional problem taken from four different functions

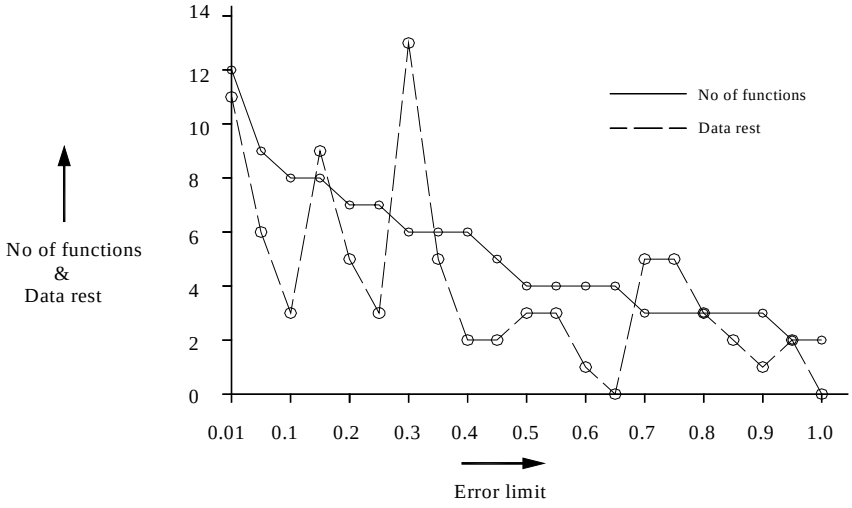

Fig. 4. Variation of the number of obtained functions along and outliers with different values of error limit

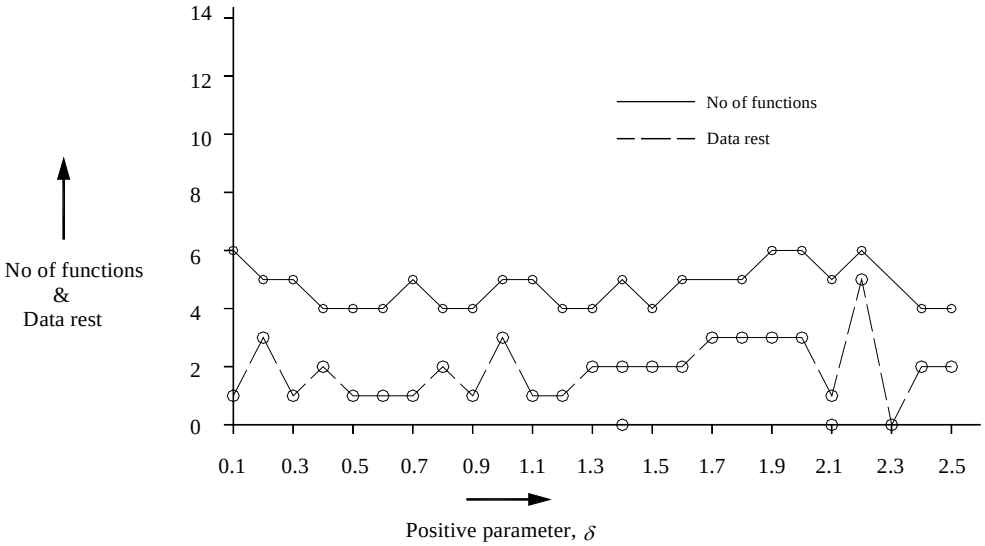

Fig. 5. Variation of the number of functions obtained and outliers with different values of δ

Table 1

Numerical values of the modified mountain function in the first iteration

	x									
Node	1	2	3	4	5	6	7	8	9	10
Mountain potential	2	10	10.03	10.3309	10.1004	1	2	2	0	4
Number of data points selected	2	10	10	10	10	1	2	2	0	2

Table 2

Numerical values of the modified mountain function in the second iteration

	x									
Node	1	2	3	4	5	6	7	8	9	10
Mountain potential	16	16.041	17.2995	1.0001	1	2	0	1	7.0214	9.0004
Number of data points selected	16	16	16	1	1	2	0	1	7	7

Table 3

Numerical values of the modified mountain function in the third iteration

	x									
Node	1	2	3	4	5	6	7	8	9	10
Mountain potential	5	5	5	5	5	5	5.0015	5.60254	5.0214	7.0004
Number of data points selected	5	5	5	5	5	5	5	5	5	5

Table 4

Results obtained by the approach with different grid nodes

No of grid nodes	Local functions with the values of function coefficients	Outlier(s)
5x5=25	$Y = 0.255737\ x_1 + 0.262316\ x_2 + 0.146598\ x_1 x_2$ $Y = 0.471553\ x_1 + 0.265526\ x_2 + 0.0972468\ x_1 x_2$ $Y = 0.188476\ x_1 + 0.308624\ x_2 + 0.201059\ x_1 x_2$ $Y = 0.30182\ x_1 + 0.359994\ x_2 + 0.136872\ x_1 x_2$	4
8x8=64	$Y = 0.244295\ x_1 + 0.241752\ x_2 + 0.148673\ x_1 x_2$ $Y = 0.48747\ x_1 + 0.251172\ x_2 + 0.0971293\ x_1 x_2$ $Y = 0.208074\ x_1 + 0.310874\ x_2 + 0.199556\ x_1 x_2$ $Y = 0.31048\ x_1 + 0.309994\ x_2 + 0.146872\ x_1 x_2$	3
10x10=100	$Y = 0.264178\ x_1 + 0.209812\ x_2 + 0.150181\ x_1 x_2$ $Y = 0.491878\ x_1 + 0.209302\ x_2 + 0.0906482\ x_1 x_2$ $Y = 0.237614\ x_1 + 0.204207\ x_2 + 0.19509\ x_1 x_2$ $Y = 0.30857\ x_1 + 0.310301\ x_2 + 0.14557\ x_1 x_2$	1
12x12=144	$Y = 0.234275\ x_1 + 0.214068\ x_2 + 0.150115\ x_1 x_2$ $Y = 0.500239\ x_1 + 0.210491\ x_2 + 0.0979373\ x_1 x_2$ $Y = 0.216922\ x_1 + 0.305893\ x_2 + 0.199011\ x_1 x_2$ $Y = 0.30182\ x_1 + 0.359994\ x_2 + 0.146872\ x_1 x_2$	1
15x15=225	$Y = 0.212776\ x_1 + 0.211174\ x_2 + 0.150117\ x_1 x_2$ $Y = 0.489096\ x_1 + 0.210735\ x_2 + 0.0984941\ x_1 x_2$ $Y = 0.215095\ x_1 + 0.305113\ x_2 + 0.198796\ x_1 x_2$ $Y = 0.30728\ x_1 + 0.31748\ x_2 + 0.15144\ x_1 x_2$	1
20x20=400	$Y = 0.213765\ x_1 + 0.208527\ x_2 + 0.149409\ x_1 x_2$ $Y = 0.48941\ x_1 + 0.217405\ x_2 + 0.104761\ x_1 x_2$ $Y = 0.210991\ x_1 + 0.290079\ x_2 + 0.191661\ x_1 x_2$ $Y = 0.301611\ x_1 + 0.31693\ x_2 + 0.151155\ x_1 x_2$	None