



Stabilization of regression trees

T. Urban, T. Kämpke

*Research Institute for Applied Knowledge Processing,
FAW Ulm, Germany.*

Abstract

In this paper, we present a hierarchical approach to simultaneous regression and classification. Regression obviously becomes more accurate by assessing a regression surface to each of a given class of a finite sample set compared to one regression surface for the sample as a whole. For class formation, a tree of regression surfaces is constructed that balances minimization of the regression error and “stabilization” towards unseen data.

Common tree-structured regression algorithms split nodes according to an independent variable. Terminal nodes correspond to one specific class. Such constructions gave rise to less complicated approaches that have mainly been used for adaptive classification in machine learning. The considerable advantage of regression trees over a single regression is often set off by trees’ poor behaviour on unseen data of supposedly the same nature. Also, a tree formation solely based on minimizing the regression error may not generate information by which to assign new data to terminal nodes. Generalization is addressed by a stabilization operation. Each sample vector is assigned to its nearest neighbour which in turn is assigned to its nearest neighbour etc. This results in so-called neighbour chains that partition the sample set. Class splitting during tree formation is here restricted to classes that contain neighbour chains completely. For classification, unseen sample data are assigned to its closest neighbour used in tree formation. The regression surface of the class containing the corresponding neighbour chain is then used for estimation.



1 Introduction

In the field of data mining, classification and regression are two processes essential to intelligent data analysis and prediction. The basic aim is to learn from a finite sample set in order to partition this set into homogenous classes. Each sample consists of several independent variables and of one dependent variable denoting the discrete class in classification and some of continuous value in regression. Standard algorithms like the tree-structured regression algorithms CART [2] and the decision tree and rule induction program C4.5 [4] are used as predictors, whereby those processes split nodes according to a threshold operating on one of the independent variables. So regression trees include only a fraction of the complete information set at every split node. Also, tree-structured regression solely based on minimizing the regression error neglects the generalization principle of the behaviour of the generated tree on unseen data.

An overview about other related algorithms for classification is presented in [5] and [1]. Other techniques of classification and clustering can be found as recursive partitioning algorithms like [3].

In this paper we present a hierarchical approach to simultaneous regression and classification. Thereby, a stabilization operation with nearest neighbour chains supports the generalization. The improvement in accuracy is reached by partitioning the learning data set and assessing specific regression surfaces to each of the subsets. The generated regression tree considers the whole information of the sample set at each split node rather than the information of one independent variable only. Each sample vector is assigned to its nearest neighbour which in turn is assigned to its nearest neighbour etc. This results in so-called neighbour chains that partition the sample set. The sizes of nearest neighbour chains are not predetermined and depend on the relative proximity within the sample data set. Class splitting during the generation of the tree is restricted to classes that contain neighbour chains completely. The depth of the regression tree depends on the number of samples in the terminal nodes. In any case the presented method improves the regression error over the regression to the sample as a whole.

In the phase of assignment of test data, an unseen sample data is assigned to its closest neighbour from the training set. The regression surface of the class containing the chain of that neighbour is then used for estimation. The stabilization operation is analogue to the transition from nearest neighbour classification to k-means classification. These stabilized regression trees have been successfully applied to forecasting the average telephone traffic in a real-life cellular phone network.

In the following section 2 we introduce the methodology of tree formation and the phase of estimation. We conclude with examples.

2 Basic method

Normally regression trees are used as predictors, whereby the tree structure partitions the complete data set by a sequence of binary splits each working on one of the independent variables [2]. Our approach is to determine the regression tree according to both error minimization and formation of neighbour preserving subsets with the latter not depending on a single independent variable only.

2.1 Least squares regression

A sample consists of finite many data $(\mathbf{x}, y)$ with $\mathbf{x}$ falling in a measurement space X and y being a real-valued number usually called the *response* or *dependent* variable. The variables in $\mathbf{x} = (x_1, x_2, \dots, x_M)$ are often referred to as the *M independent* variables. A prediction rule or regression function is a real-valued function $d(\mathbf{x})$ defined on X . Regression analysis is the generic term describing the construction of a predictor $d(\mathbf{x})$ starting from a sample S consisting of N cases $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$.

Definition 1 For linear regression the predictor $d(\mathbf{x})$ is defined as

$$d(\mathbf{x}) = a_0 + a_1 \cdot x_1 + \dots + a_M \cdot x_M \quad (1)$$

with $d(\mathbf{x}) \in \mathbb{R}^1$.

Hereby $\mathbf{a} = (a_0, \dots, a_M)$ denotes the coefficient vector to be estimated. The construction of the predictor $d(\mathbf{x})$ intends to predict the response variable corresponding to future measurement vectors as accurately as possible.

The restriction to linear regression is not essential, but suffices to demonstrate the method of improving accuracy of the prediction and stabilization of the regression tree. The accuracy of $d(\mathbf{x})$ will be measured by the expected average squared error $R_{MSE}(d)$ of a learning sample S . We assume that the samples $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$ are drawn independently from the same random vector $(\mathbf{X}, Y)$.

Definition 2 The mean squared error $R_{MSE}(d)$ of the predictor d is defined as

$$R_{MSE}(d) = E\{(Y - d(\mathbf{X}))^2\} \quad (2)$$

$$= \frac{1}{N} \sum_{n=1}^N (y_n - d(\mathbf{x}_n))^2. \quad (3)$$

In classification, $R_{MSE}(d)$ denotes the misclassification rate of a classifier d . For regression it is the common mean squared error estimate. Minimizing the mean squared error $R_{MSE}(d)$ leads to the problem of optimizing the coefficient vector $\mathbf{a}$ from definition 1.

Lemma 1 *There exists one optimal coefficient vector $\mathbf{a}_{opt}$ which minimizes the mean squared error $R_{MSE}(d)$ of the predictor d*

$$\min_a R_{MSE}(d) = \min_a E \{ (Y - d(\mathbf{X}))^2 \} \quad (4)$$

$$= E \{ (Y - \mathbf{a}_{opt}^T \cdot \mathbf{X})^2 \}. \quad (5)$$

Our regression method and stabilization operation is independent from the error term distribution and from the criterion of selecting a split at every intermediate node of the regression tree. In contrast to our approach, the regression method described in [2] focuses on data sets whose dimensionality requires some sort of variable selection. Therefore, in linear regression the common practice is to use either a stepwise selection or a best subsets algorithm. The resulting stabilized tree-structured approach can be compared with tree structured classification methods.

2.2 Partitioning the sample set

The sample set S typically undergoes iterative partitioning with S_+ denoting some proper subset of S . Furthermore, by d_{S_+} we denote a predictor optimized particularly to S_+ . Then the least squared error of predictor $d_{S_+}(\mathbf{x})$ is less or equal than the least squared error of predictor $d(\mathbf{x})$ for subset S_+ :

$$\frac{1}{|S_+|} \sum_{i \in S_+} (y_i - d_{S_+}(\mathbf{x}_i))^2 \leq \frac{1}{|S_+|} \sum_{i \in S_+} (y_i - d_S(\mathbf{x}_i))^2. \quad (6)$$

Figure 1 pictures the effect of the described inequality for a sample set S with $N = 15$ points and the underlying regression lines. The problem is to find the subsets and finally to assign unknown test samples to a subset. To this end we use the features of the k -nearest neighbour methodology especially for $k = 1$. The k -nearest neighbour graph represents each data point by a vertex. From each vertex v_0 exactly k arcs (= directed edges) emanate each pointing towards one of the k nearest neighbours of v_0 . Noteworthy, the number of arcs pointing towards a vertex may differ from k , as the relation expressed by arc orientation need not be reversible.

The Euclidean space is used as the metric space for generating nearest neighbour graphs. For $k = 1$ every data point $\mathbf{x}_i$ has one nearest neighbour $\mathbf{x}_j = NN(\mathbf{x}_i)$ with respect to the minimum Euclidean distance. A chain is defined as a weak connect component in the 1-nearest neighbour graph. In other words, two vertices belong to the same chain if arc sequences starting at these vertices eventually lead to a common vertex.

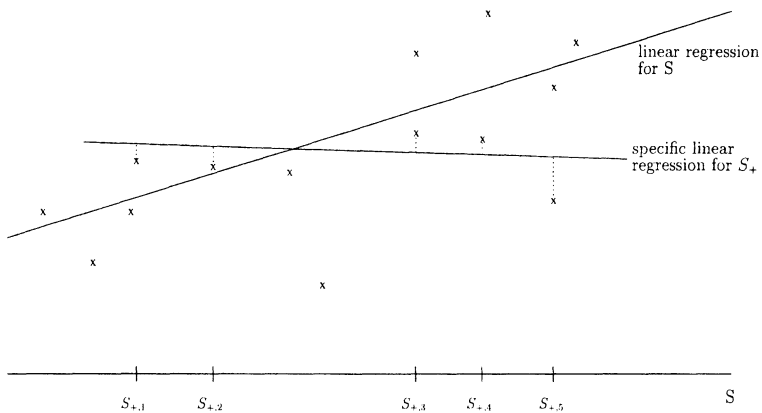


Figure 1: Data set S with $N = 15$ points in twodimensional space with regression line and subset S_+ of five elements with specific regression line.

We assume a configuration of a nearest neighbour chain with four elements as depicted in figure 2. Nearest neighbours are indicated by those vertices from which directed edges emanate.

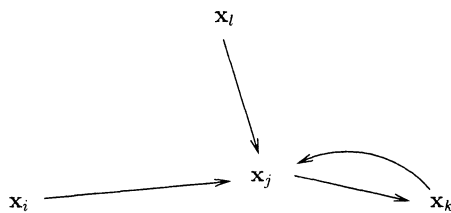


Figure 2: Example of a chain with four elements.

Example 1 The set

$$C = \{x_i, x_j, x_k, x_l\}$$

is a nearest neighbour chain if

$$x_j = NN(x_i) \wedge x_k = NN(x_j) \wedge x_j = NN(x_k) \wedge x_j = NN(x_l)$$

for pairwise different indices i, j, k, l , comp. figure 2.

The generation of nearest neighbour chains C_n reduces the complete data set with N cases to a sparse representation with chains $C_n, n = 1, \dots, \nu$, and $\nu \leq \frac{N}{2}$. In the worst case there is only one chain, and in the other extreme every chain consists of only two data points.

Data set S can be divided into subsets S_+ and S_- based on the chains and the minimum sum squared error $R_{SSE,C}(d)$.

Definition 3 The sum squared error $R_{SSE}(d)$ of the predictor d is defined as

$$R_{SSE}(d) = \sum_i (y_i - d(\mathbf{x}_i))^2. \quad (7)$$

The sets S_+ and S_- are constructed from S as follows. A chain becomes part of S_+ if all its elements $(\mathbf{x}_i, y_i)$ have actual values $y_i \geq d(\mathbf{x}_i)$. A chain becomes part of S_- if all its elements $(\mathbf{x}_i, y_i)$ have actual values $y_i < d(\mathbf{x}_i)$. In case a chain has elements $(\mathbf{x}_i, y_i)$ with $y_i \geq d(\mathbf{x}_i)$ and $y_i < d(\mathbf{x}_i)$, the chain is tentatively split into the subsets $\{(\mathbf{x}_i, y_i) | y_i \geq d(\mathbf{x}_i)\}$ and $\{(\mathbf{x}_i, y_i) | y_i < d(\mathbf{x}_i)\}$. For those subsets and for the two “pure” cases the two squared error sums $R_{SSE\{+\},C}(d)$ and $R_{SSE\{-\},C}(d)$ are considered:

$$R_{SSE\{+\},C}(d) := \sum_{(\mathbf{x}_i, y_i) \in C \text{ with } y_i \geq d(\mathbf{x}_i)} (y_i - d(\mathbf{x}_i))^2, \quad (8)$$

$$R_{SSE\{-\},C}(d) := \sum_{(\mathbf{x}_i, y_i) \in C \text{ with } y_i < d(\mathbf{x}_i)} (y_i - d(\mathbf{x}_i))^2. \quad (9)$$

The construction of the sets S_+ and S_- is based on the minimum sum squared error $R_{SSE\{+,-\}}(d)$ with

$$S_+ := \{C_i | R_{SSE\{+\},C_i}(d) > R_{SSE\{-\},C_i}(d)\}; \quad (10)$$

$$S_- := \{C_i | R_{SSE\{+\},C_i}(d) < R_{SSE\{-\},C_i}(d)\}. \quad (11)$$

For both sets S_+ and S_- containing the assigned 1-nearest neighbour chains it is necessary and advantageous to generate new predictors $d_{S_+}(\mathbf{x}_i)$ and $d_{S_-}(\mathbf{x}_i)$. Thus the least squared error for test data samples with the specific predictors d_{S_+} and d_{S_-} is probably less than the least squared error with the previous predictor $d(\mathbf{x})$.

This partitioning process can be repeated at the next level and so forth. Finally, after the process of separating and allocating the sample data in chains and in sets of chains with their specific predictors we get a tree structure like in figure 3. Every node denotes a set of nearest neighbour chains and their samples. For these sets there exists a specific predictor constructed from appropriate samples.

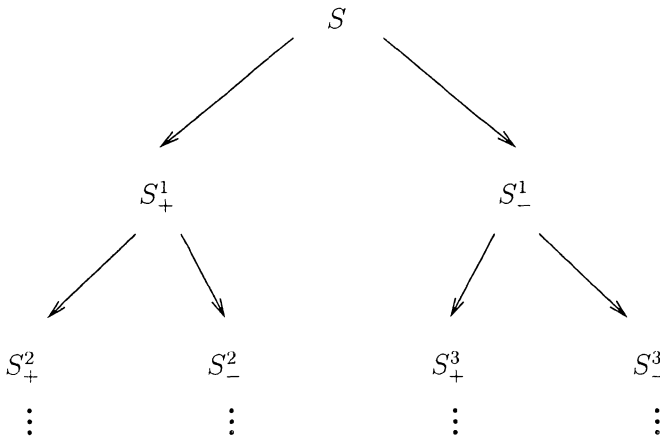


Figure 3: Stabilized regression tree as a result of partitioning the complete learning sample set S on different levels.

2.3 Assigning unseen test samples

After the generation of the regression tree all nearest neighbour chains are grouped in the terminal nodes.

In the classification procedure or predicting phase, unseen test data have to be assigned to the terminal nodes, so that it is possible to predict the dependent value of this unknown case. Generalization is addressed by the stabilization operation. Therefore, an unseen sample data is assigned to its closest neighbour used in tree formation. For estimation the predictor of the regression containing the corresponding neighbour chain is used.

In the case of a tree formation like in figure 3 we get four nodes in the third level of the regression tree and thus we have four regression surfaces.

3 Examples

The method of stabilized regression trees was applied in an investigation of average telephone traffic in real-life cellular phone networks. Therefore the given data set contained about 8000 cases of real-valued variables. 15 independent variables including population of a radio cell, land use of a radio cell etc. were analyzed with respect to prediction of average phone traffic within a radio cell. Linear regression of the whole sample set resulted in a mean squared error of 0.62. After partitioning the node in two subsets and computing the specific predictors the mean squared error reduces to a value of 0.59. This value is the average of the mean squared

errors of test data for both existing regression surfaces at this level. The value for level three was 0.57 and for level four containing 8 terminal nodes the value was 0.56. The improvement of almost 10 percent of the mean squared error denoted a minimization of the regression error towards unseen test data.

3.1 Synthetic data set

To illustrate the particular behaviour of the algorithm for simultaneous regression and classification we generated a synthetic data set S . The underlying data set contains 8 cases consisting of one response variable dependent on a single feature. For a least squares linear regression of the entire data set the straight line shown in figure 4 is the result of the estimation process.

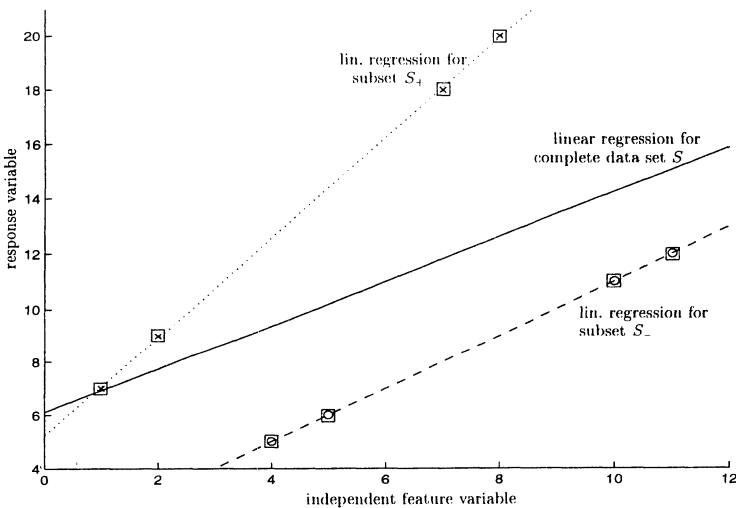


Figure 4: Results of the linear regression for the synthetic data set S and the binary split in subsets S_+ and S_- .

In the next step two subsets S_+ and S_- are generated by partitioning the whole sample set with regard to nearest neighbour chains. For both of these subsets a specific linear least squared regression results in two predictors, which are depicted as dotted and dashed lines in figure 4.

Therefore in this example we get an optimal result for prediction with only one binary split and two leaves in the stabilized regression tree. For the sake of completeness we illustrate the resulting regression tree. Every node describes a set of samples S and their appropriate predictors $d(S)$. Figure 5 illustrates the very simple regression tree.

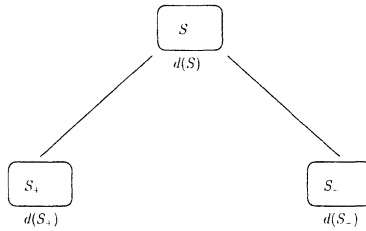


Figure 5: Stabilized regression tree as the result for the synthetic data set S .

The difference between the introduced process and other regression tree methods consists in the splitting procedure. Other common techniques such as CART [2] or decision trees like C4.5 [4] splits nodes according to one of the independent variables. In the case of the C4.5 decision tree generator the result of the splits is represented as dashed line for the first split and as dash-dotted line for the second split. Additionally in figure 6 the decided class for the dependent variable is listed at the head of the chart.

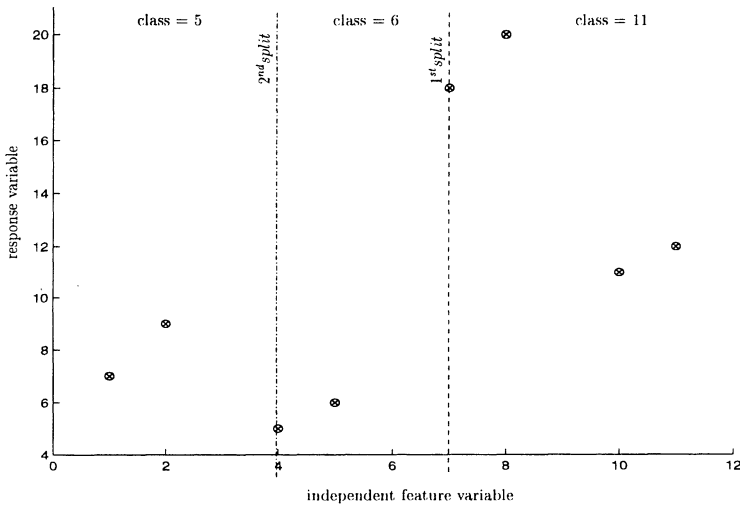


Figure 6: Results of the binary split of the independent variable with the c4.5 decision tree algorithm for the synthetic data set S .



4 Conclusion and outlook

In comparison to common classification and regression tree algorithms we partition the sample set based on all variables and the polynomial regression equation. The splitting procedure for the growing the regression tree consists of multivariate decisions on the regression surfaces of intermediate nodes. As a consequence, in some applications the accuracy is improved by this stabilization of the regression tree.

For further examinations the partitioning step is an important basic approach. The binary splits into subsets can be extended to splits with more than two subsets. This is likely to further improve the partition of the sample set and it also allows to store the information of the regression surfaces.

5 Acknowledgement

This work was partially supported by project D 3 of the SFB 527 (<http://www.uni-ulm.de/SFB527/Projects/d3.html>) sponsored by the Deutsche Forschungsgemeinschaft.

References

- [1] Agrawal, R., Imielinski, T., Swami, A., Database mining: A performance perspective, *IEEE Transactions on Knowledge and Data Engineering*, 5(6), pp. 914-925, 1993.
- [2] Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., *Classification and Regression Trees*, The Wadsworth Statistics/Probability Series, Wadsworth International Group, 1984.
- [3] Karypis, G., Kumar, V., Multilevel k -way hypergraph partitioning, *Proceedings of the Design and Automation Conference*, 1999.
- [4] Quinlan, J.R., *C4.5 Programs for machine learning*, Morgan Kaufmann, San Mateo, California, 1993.
- [5] Lim, T.-S., Loh, W.-Y., A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms, *Machine Learning*, forthcoming, 1999.