Connections between optimisation-based regressor selection and analysis of variance

Jacob Roll, Ingela Lind

Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: http://www.control.isy.liu.se
E-mail: roll@isy.liu.se, ingela@isy.liu.se

20th February 2006

Report no.: LiTH-ISY-R-2728
Accepted for publication in Conference on Decision and Control, 2006

Technical reports from the Control & Communication group in Linköping are available at http://www.control.isy.liu.se/publications
Abstract
Earlier contributions have shown that Analysis of Variance (ANOVA) can be successfully used for finding good regressors for nonlinear models in a nonlinear black-box system identification context. In this paper, it is shown that the ANOVA problem can be recast as an optimisation problem. Two modified, convex versions of the ANOVA optimisation problem are then proposed, and it turns out that they are closely related to the nn-garrote and wavelet shrinkage methods, respectively. In the case of balanced data, it is also shown that the methods have a nice orthogonality property in the sense that different groups of parameters can be computed independently.

Keywords: ANOVA, 1-norm, Lasso, nn-garrote, nonlinear identification, regressor selection
Connections between optimisation-based regressor selection and analysis of variance

Jacob Roll and Ingela Lind

2006-02-20

Abstract

Earlier contributions have shown that Analysis of Variance (ANOVA) can be successfully used for finding good regressors for nonlinear models in a nonlinear black-box system identification context. In this paper, it is shown that the ANOVA problem can be recast as an optimisation problem. Two modified, convex versions of the ANOVA optimisation problem are then proposed, and it turns out that they are closely related to the nn-garrote and wavelet shrinkage methods, respectively. In the case of balanced data, it is also shown that the methods have a nice orthogonality property in the sense that different groups of parameters can be computed independently.
1 Introduction

Regressor selection is an important subproblem of system identification. We assume that the measured output data is given by the unknown nonlinear function \( g(\cdot) \), by the relation

\[
y(t) = g(\varphi(t)) + e(t),
\]

where \( e(t) \) is measurement noise and \( \varphi(t) \) is a regression vector formed from measured input and/or output data. The purpose of regressor selection is to decide what elements \( \varphi_k(t) \) of the regression vector \( \varphi(t) \) are important to explain the output \( y(t) \). Akaike (1969) is an early reference for linear systems while Billings and Voon (1986) was among the first to study regressor selection for nonlinear systems. There are many recent contributions, but the problem is still seen as unexplored (Efron et al., 2004, pp. 494–499). The reason why it is important to study the regressor selection apart from model estimation is a curse-of-dimensionality problem. The number of parameters in nonlinear models often grows very quickly with the number of included regressors, so a comparison between all possible models and regressor combinations is not feasible. Furthermore, there are infinitely many nonlinear models to select from even when the regressors are fixed.

In the last decade, there has been a growing interest in different kinds of regularization methods using 1-norm regularization, such as LASSO (Tibshirani, 1996), nn-garrote (Breiman, 1995), and wavelet shrinkage (Donoho and Johnstone, 1994). Since the 1-norm penalty term tends to force several parameter values to zero, these methods can be viewed as optimisation-based regressor selection methods.

A classical data analysis tool, which can be used for regressor selection is Analysis of Variance (ANOVA Miller (1997); Montgomery (1991)). In Lind (2006); Lind and Ljung (2005), it was shown how ANOVA can be applied to structure identification of dynamic systems.

In this paper, it is shown that the ANOVA problem can be recast as an optimisation problem (Section 4). Two modified versions of the ANOVA optimisation problem are then proposed in Section 5, and as pointed out in Section 8, it turns out that they are closely related to the nn-garrote and wavelet shrinkage methods, respectively. In the case of balanced data, it is also shown that the methods have a nice orthogonality property in the sense that different groups of parameters can be computed independently (Section 6).

2 ANOVA

The statistical analysis method ANOVA (Miller (1997); Montgomery (1991), among many others) is a widely spread tool for finding out which factors contribute to given measurements. It has been used and discussed since the 1930’s and is a common tool in, e.g., medicine and quality control applications. The method is based on hypothesis tests with F-distributed test variables computed from the residual quadratic sum. There are several slightly different variants (Miller, 1997). Here the fixed effects model with two factors will be described. The generalisation to a larger number of factors is immediate.

In the standard ANOVA setup, the regression vectors are assumed to lie on a discrete grid. For continuously varying regression vectors, we instead divide the
range of each regressor $\varphi_k$ into $m_k$ different intervals, and so the regressor space is partitioned into a set of cells. For the study of ANOVA, it is then convenient to reorder the data by grouping them according to which cells they belong to. For the two-factor model, introduce the notation $b \in \{(j_1, j_2), j_k = 1, \ldots, m_k\}$ for the different cells, and write $\varphi(b, p), y(b, p)$ for the $p$th data point in cell $b$.

Assume that there are $N_b$ data points in cell $b$.

Assume that the collected measurement data can be described by a linear statistical model,

$$y((j_1, j_2), p) = \theta_0 + \theta_{1,j_1} + \theta_{2,j_2} + \theta_{1,2,j_1,j_2} + \epsilon((j_1, j_2), p)$$

(2)

where the $\epsilon((j_1, j_2), p)$ are independent Gaussian distributed variables with zero means and constant variance $\sigma^2$. The parameter $\theta_0$ is the overall mean. For each (quantised) level $j_1 = 1, \ldots, m_1$ of the first regressor ($\varphi_1(t)$) there is a corresponding effect $\theta_{1,j_1}$, and for each level $j_2 = 1, \ldots, m_2$ of the second regressor ($\varphi_2(t)$) the corresponding effect is $\theta_{2,j_2}$. The interaction between the regressors is described by the parameters $\theta_{1,2,j_1,j_2}$. The sum of a batch of indexed parameters over any of its indices is zero.

For a linear ($y(t) = \eta_1 \varphi_1(t) + \eta_2 \varphi_2(t) + e(t)$) or a non-linear additive system ($y(t) = g_1(\varphi_1(t)) + g_2(\varphi_2(t)) + e(t)$), the interaction parameters $\theta_{1,2,j_1,j_2}$ are zero. These are needed when the non-linearities are of a non-additive nature, i.e., $y(t) = g(\varphi_1(t), \varphi_2(t)) + e(t)$.

Since the regressors are quantised, it is a very simple procedure to estimate the model parameters by computing means (for simplicity we assume that all regressor level combinations are sampled equally, which means that all $N_b$ are equal):

$$\bar{y}_{\cdot \cdot} = \frac{1}{m_1 m_2 N_b} \sum_{j_1=1}^{m_1} \sum_{j_2=1}^{m_2} \sum_{p=1}^{N_b} y((j_1, j_2), p), \quad \bar{y}_{j_1 \cdot} = \frac{1}{m_2 N_b} \sum_{j_2=1}^{m_2} \sum_{p=1}^{N_b} y((j_1, j_2), p),$$
$$\bar{y}_{\cdot j_2} = \frac{1}{m_1 N_b} \sum_{j_1=1}^{m_1} \sum_{p=1}^{N_b} y((j_1, j_2), p), \quad \bar{y}_{j_1 j_2} = \frac{1}{N_b} \sum_{p=1}^{N_b} y((j_1, j_2), p).$$

(3)

which are the overall mean, the means over the regressor levels and the cell means. For example, the constant $\theta_0$ would correspond to $\bar{y}_{\cdot \cdot}$, while the effects from the first regressor are computed as $\theta_{1,j_1} = \bar{y}_{j_1 \cdot} - \bar{y}_{\cdot \cdot}$. The number of free parameters in the model is equal to the number of cells.

ANOVA is used for testing which of the parameters significantly differ from zero and for estimating the values of the parameters with standard errors, which makes it a tool for exploratory data analysis. The residual quadratic sum, $SS_T$, is used to design test variables for the different batches (e.g., the $\theta_{1,j_1}$, s) of parameters. Under the assumptions on $\epsilon((j_1, j_2), p)$ stated above and in the case when all $N_b$ are equal, the residual quadratic sum can be divided into four
independent parts:

\[
SS_T = \sum_{j_1=1}^{m_1} \sum_{j_2=1}^{m_2} \sum_{p=1}^{N_b} (y(j_1, j_2, p) - \bar{y}_{..})^2
\]

\[
= \sum_{j_1=1}^{m_1} m_2 N_b (\bar{y}_{j_1..} - \bar{y}_{..})^2 + \sum_{j_2=1}^{m_2} m_1 N_b (\bar{y}_{j_2..} - \bar{y}_{..})^2
\]

\[
+ \sum_{j_1=1}^{m_1} \sum_{j_2=1}^{m_2} N_b (\bar{y}_{j_1,j_2} - \bar{y}_{j_1..} - \bar{y}_{..,j_2} + \bar{y}_{..})^2
\]

\[
+ \sum_{j_1=1}^{m_1} \sum_{j_2=1}^{m_2} \sum_{p=1}^{N_b} (y(j_1, j_2, p) - \bar{y}_{j_1,j_2})^2
\]

\[
= SS_A + SS_B + SS_{AB} + SS_E. \quad (4)
\]

Each part is related to one batch of parameters. If all the parameters in the batch are zero, the corresponding quadratic sum is \(\chi^2\)-distributed if divided by the true variance \(\sigma^2\) (see, e.g., Montgomery (1991, page 59)). Since the true variance is not available, the estimate \(\hat{\sigma}^2 = \frac{SS_E}{m_1 m_2 (N_b - 1)}\) is used to form \(F\)-distributed test variables, e.g., for \(\theta_{1,j_1}^1\):

\[
v_A = \frac{SS_A / (m_1 - 1)}{SS_E / (m_1 m_2 (N_b - 1))}. \quad (5)
\]

If all the \(\theta_{1,j_1}\)'s are zero, \(v_A\) belongs to an \(F\)-distribution with \(m_1 - 1\) and \(m_1 m_2 (N_b - 1)\) degrees of freedom. If any \(\theta_{1,j_1}\) is nonzero it will give a large value of \(v_A\), compared to an \(F\)-table. This is, of course, a test of the null hypothesis that all the \(\theta_{1,j_1}\)'s are zero, which correspond to the case that the regressor \(\varphi_1\) does not have any main effect on the measurements \(y\).

The most important modeling simplifications made are the assumptions that the variance is equal for all \(\epsilon((j_1, j_2), p)\) and that the random error component is Gaussian distributed. The \(F\)-tests are quite robust against violations against both assumptions (Krishnaiah, 1980, Chapter 7). When a regressor, with values belonging to a continuous interval, is quantised to a discrete set of levels both these assumptions are violated. Another complication with using ANOVA in the system identification context is that the regressors often are correlated, e.g., in NARX-models (see Lind (2006)). The correlated continuous-level regressors violate the assumption that the \(\epsilon((j_1, j_2), p)\) are independent.

When all factor level combinations are not represented by an equal amount of measurement data, the design is called unbalanced. This is further discussed in Section 8.

In Section 4, we will write ANOVA as a simple optimisation problem, which will then be extended in Section 6.1 to show the relationships to other optimisation-based estimation and model selection methods. The latter will be introduced next, in Section 3.

### 3 Optimisation-based model selection

A common way of deciding on model complexity is to introduce an extra term to the objective function of the model fitting problem, penalising high complexity
models. Examples of these methods include the different Akaike criteria (Akaike, 1981), the MDL criterion (Rissanen, 1978, 1986), BIC (Schwarz, 1978) and Mallows CP (Mallows, 1973). Ordinary regularisation can also be thought of as controlling the complexity of the model (see, e.g., Ljung, 1999), and could also be used in combination with a threshold, so that parameters obtaining values below the threshold are set to zero.

An alternative to using a threshold is to let the regularisation/penalty term be the 1-norm of the parameters. In this way, parameters with small values will tend to become exactly zero. 1-norm penalty terms are used in several different estimation methods, some of which are described below.

In the following, $X(t) = F(\varphi(t))$ denotes some fixed function of $\varphi(t)$, e.g., a basis function expansion. It should be noted that $X(t)$ can very well be of higher dimension than $\varphi(t)$, so that several elements $X_k(t)$ of $X(t)$ can be functions of the same regressor $\varphi_k(t)$. Also, let $X = [X(1) X(2) \ldots X(N)]$ and let $X_i$ denote the $i$th row of $X$ (corresponding to the $i$th basis function). The model classes used in most methods below use this kind of function expansions and hence are linear in the parameters. A special kind of basis function expansion used in some methods is the ANOVA expansion (Friedman, 1991), where the system function $f$ is decomposed into functions of different combinations of regressors (here $d$ is the dimension of $\varphi$):

$$f(\varphi) = f_0 + \sum_{k=1}^{d} f_k(\varphi_k) + \sum_{k=1}^{d-1} \sum_{k+k+1}^{d} f_{k,l}(\varphi_k, \varphi_l) + \sum_{k=1}^{d-2} \sum_{k+k+1}^{d} \sum_{j=l+1}^{d} f_{k,l,j}(\varphi_k, \varphi_l, \varphi_j) + \ldots + f_{1,2,\ldots,d}(\varphi_1, \varphi_2, \ldots, \varphi_d).$$ (6)

3.1 Non-negative garrote

The non-negative garrote (nn-garrote) was introduced by Breiman (1995) as a shrinkage method for ordinary least squares regression. The method is a two-step procedure:

Step 1 Solve the ordinary least squares problem

$$\min_{\theta} \sum_{t=1}^{N} (y(t) - \theta^T X(t))^2.$$ (7)

The solution is denoted by $\hat{\theta}$.

Step 2 Solve the regularised problem

$$\min_{\mathbf{c}} \sum_{t=1}^{N} \left( y(t) - \sum_{k} c_k \hat{\theta}_k X_k(t) \right)^2,$$

subject to $\|\mathbf{c}\|_1 \leq \rho$,

$$c_k \geq 0,$$

where $\|\mathbf{c}\|_1 = \sum_k |c_k|$ is the 1-norm of the vector $\mathbf{c}$. The bound $\rho > 0$ can be regarded as a design parameter, the value of which can be determined by, e.g., cross validation (see, e.g., Hastie et al., 2001; Ljung, 1999).
Note that by using Lagrange multipliers, (8) can be transformed into

$$\min_{c} \sum_{t=1}^{N} \left(y(t) - \sum_{k} c_k \hat{\theta_k} X_k(t)\right)^2 + \lambda \|c\|_1, \quad (9)$$

subject to $c_k \geq 0$,

for some value of $\lambda$, which depends on $\rho$.

In Yuan and Lin (2006) the nn-garrote method was extended to the case with grouped regressors, which means that $c$ is of a smaller dimension than $\theta$. This is called group nn-garrote. The idea is to collect regressors with the same origin, e.g. regressors formed from the same $\varphi_k$ as in a polynomial expansion, in the same group. Assume that the vector $X(t)$ is grouped according to

$$X(t) = \begin{pmatrix} X_{K_1}(t) \\ X_{K_2}(t) \\ \vdots \\ X_{K_L}(t) \end{pmatrix},$$

where $X_{K_1}(t) = \begin{pmatrix} X_1(t) \\ \vdots \\ X_{k_1}(t) \end{pmatrix}$, $X_{K_2}(t) = \begin{pmatrix} X_{k_1+1}(t) \\ \vdots \\ X_{k_2}(t) \end{pmatrix}$, etc.

and similarly for $\theta$. Then the group nn-garrote can be written

$$\min_{c} \sum_{t=1}^{N} \left(y(t) - \sum_{l=1}^{L} c_l \hat{\theta}_{K_l}^T X_{K_l}(t)\right)^2 + \lambda \|c\|^T, \quad (10)$$

subject to $c_l \geq 0$,

In Yuan and Lin (2005) the consistency of nn-garrote is studied. They consider the “solution path” (the solutions to (10) as a function of $\lambda$) and show that it “contains an estimate that correctly identifies the set of important variables” with probability tending to one. It also turns out that the solution path is piecewise linear, which can be exploited to give a total computational complexity as an ordinary least squares problem. This property is shown using similar ideas as for LARS (Section 3.3).

### 3.2 Lasso

The Lasso (Least Absolute Shrinkage and Selection Operator) method was proposed by Tibshirani (1996). The Lasso estimate is computed as the solution to

$$\min_{\theta} \sum_{t=1}^{N} (y(t) - \theta^T X(t))^2, \quad (11)$$

subject to $\|\theta\|_1 \leq \rho$.

Also here, $\rho$ can be determined by, e.g., cross validation.

Comparing with nn-garrote, we can see that $\theta_k$ in Lasso correspond to the nn-garrote coefficients $c_k \hat{\theta}_k$. The main difference is that $\theta_k$ in Lasso are penalised equally according to their absolute value, while only $c_k$ is penalised in nn-garrote, i.e., the penalty is determined by the relative size of the coefficient $c_k \hat{\theta}_k$ compared to the least squares estimate $\hat{\theta}_k$. Furthermore, the nn-garrote coefficients $c_k \hat{\theta}_k$ are restricted to have the same sign as the least squares estimate.
3.3 LARS

The LARS (Least Angle Regression) method is closely related to the Lasso, although it is differently formulated. It was proposed by Efron et al. (2004).

Assume that $X$ has been normalised so that $\|X_i\|_2 = 1$ for all $i$. The LARS algorithm now builds a series of linear models (analogous to the solution path of Yuan and Lin (2006) in Section 3.1)

$$\hat{y}(t|\theta) = \hat{\theta}^T X(t)$$

starting from $\theta = 0$, and successively adding one regressor at a time. The first regressor to be added is selected as the one for which $X_i$ is most correlated with $y$. $\theta_i$ is increased until another regressor $X_j$ has the same correlation with the residual $y - \hat{\theta}_i X_i$ as $X_i$ has. Then both $\theta_i$ and $\theta_j$ are increased such that $X_i$ and $X_j$ are always equally correlated to the residual, until a third regressor has reached the same correlation, etc. In this way the solution path is constructed, and continues until the ordinary least squares solution has been reached, or until the residual is zero.

Which model along the solution path to use can be determined by some criterion such as AIC-like criteria. It can be shown (Efron et al., 2004) that a slightly altered version of LARS efficiently computes the Lasso solutions for all choices of $\rho$ in (11).

3.4 ISRR

The ISRR (Iteratively Scaled Ridge Regression), proposed by Bortolin (2005) is an iterative method that differs from Lasso and nn-garrote by using 2-norm in stead of 1-norm for the penalties. This gives a faster method in each iteration, since it has an analytical solution. The estimates for the parameters $\theta$ are in iteration $k$ given by the solution to

$$\min_{\theta} \sum_{t=1}^{N} \left( y(t) - \theta^T X(t) \right)^2,$$

subject to $\theta^T D_k^{-2} \theta \leq \rho_k,$

with $D_k = \text{diag}(\theta_{k-1})$ – the solution to the previous iteration. The iteration starts with the unconstrained problem, to give the penalties, and is then iterated for several (decreasing but positive) $\rho_k$. The best value of the computed $\theta_k$ is determined by an evaluation criterion, e.g., cross validation.

3.5 Wavelet shrinkage

SURE shrinkage (Stein Unbiased Risk Estimation formulated by Donoho and Johnstone (1994)) is a variable selection method for wavelet models (Hastie et al., 2001);

$$\min_{\theta} \| (y(t) - W\theta) \|^2_2 + 2\lambda \|\theta\|_1,$$

where $W$ is an $N \times N$ orthonormal wavelet basis matrix evaluated at $N$ uniformly spaced observations. Note that this is the same optimisation criterion as in the Lasso (11).
3.6 Wavelet ANOVA

Wei and Billings (2004) combines the wavelet decomposition with the ANOVA expansion (6), giving a model which is linear in the parameters. They then combine this model with their method for variable selection, the forward orthogonal least squares (OLS) algorithm based on the error reduction ratio (Billings et al., 1988; Wei et al., 2004).

Wahba (1990) and Wahba et al. (1994) also use the ANOVA expansion, but in combination with a non-parametric smoothing spline model.

3.7 MARS

Multivariate Adaptive Regression Splines (MARS) was first introduced by Friedman (1991). The aim is to combine recursive partitioning with spline fitting. The splines are used for their smoothness, while the recursive partitioning is used for its adaptability and for its local properties, which limits the effects of the curse of dimensionality. To further improve the interpretability, the ability to capture simple (such as linear) models and properties in high dimensional spaces, the ANOVA expansion (6) is used.

3.8 Supanova

The Supanova algorithm (Gunn and Kandola, 2002) is a support vector machine with an ANOVA function expansion. The ordinary support vector machine (Hastie et al., 2001) is a model

\[ \hat{y}(t) = \sum_{i=1}^{N} \alpha_i K(X(i), X(t)), \]  

where \( K(X(i), X(t)) \) is a reproducing kernel function, and \( \alpha_i \) is the weight on the estimation data vector \( X(i) \). \( N \) is the number of estimation data. The model is evaluated in \( X(t) \), which can be any point in the regressor space.

By using the ANOVA function expansion (6), the Supanova algorithm decomposes the kernel function into \( 2^r \) subkernels, where \( r \) is the number of possible regressors (the length of \( X(i) \) and \( X(t) \)):

\[ \hat{y} = \sum_{i=1}^{N} \sum_{j=1}^{2^r} c_j K_j(X(i), X(t)), \quad c_j \geq 0. \]  

Here, the first subkernel will estimate the value of \( f(0) \) with \( K_1(X(i), X(t)) = 1 \). The next \( r \) subkernels are one-dimensional, \( K_{k+1}(X(i), X(t)) = k(X_k(i), X_k(t)) \). The subkernels are of increasing dimension until the last subkernel, which is of dimension \( r \). Two-dimensional kernels are given by the tensor products of the one-dimensional kernels, e.g.,

\[ K_{k+r+1}(X(i), X(t)) = k(X_k(i), X_k(t))k(X_{k+1}(i), X_{k+1}(t)), \]  

three-dimensional subkernels are tensor products of two- and one-dimensional subkernels, and so on. To make sure identification is possible (avoid ambiguities), all the subkernels should be zero on their axes. Possible kernel functions
that fulfil this, are, e.g., infinite splines, odd order B-splines and polynomial kernels. The decomposition is effective only if some of the subkernels gets zero weight \( c_j \). A regularized cost function of the form

\[
\min_{\alpha,c} \left\| y - \sum_{i=1}^{N} \alpha_i \sum_{j=1}^{m} c_j K_j(X(i),X) \right\|_{1,c} + \lambda_\alpha \sum_{j=1}^{2^r} c_j \alpha^T K_j \alpha + \lambda_c 1^T c,
\]

subject to \( c_j \geq 0 \ \forall j \),

is used to give that kind of sparsity in \( c_j \). Here \( y \) and \( X \) are the stacked \( y(t) \) and \( X(t) \) for \( t = 1, \ldots, N \). The \( \epsilon \)-insensitive 1-norm is used to get a limited number of support vectors, data points with nonzero \( \alpha_i \). Here, the first term corresponds to the mismatch between output data and model output, while the second term is related to the smoothness of the model and the generalisation ability. Finally, the third term controls the sparsity in \( c \). The cost function can be minimized by iteratively solving for \( \alpha \) with \( c \) fixed and solving for \( c \) with \( \alpha \) fixed. The first problem is a quadratic program, and the second a linear program. These can be efficiently solved by dedicated solvers. A comparison between the Supanova algorithm and some other nonlinear model structures is given in Lind and Roll (2006).

### 4 ANOVA as an optimisation problem

In this section, we will start the preparations for studying the connections between ANOVA and the optimisation based model selection methods described in Section 3. We will do this by formulating ANOVA itself as a simple optimisation problem. In the following sections, we will assume that the data set is balanced, unless something else is stated.

First, note that the F-tests performed by ANOVA can be interpreted as comparing the size of the numerator and denominator in expressions such as (5). By suitable normalisation, we can say that \( \theta_{1,j1} \) should be included in the model if and only if the numerator of \( v_A \) is greater than the denominator. To be able to make explicit which terms in the model (2) should be included, we introduce a binary vector \( c \in \{0,1\}^4 \) and use the following model:

\[
y((j_1,j_2),p) = c_0 \theta_0 + c_1 \theta_{1,j_1} + c_2 \theta_{2,j_2} + c_{1,2} \theta_{1,2,j_1,j_2} + \epsilon((j_1,j_2),p) \quad (18)
\]

The value of \( c_0 \) should normally be chosen to 1. Now we can formulate an optimisation problem in the binary variables \( c \), which gives the same optimal solution as ANOVA:

\[
\min_c \quad SS_T + c_1 (c_1 - 2) SS_A + c_2 (c_2 - 2) SS_B + c_{1,2} (c_{1,2} - 2) SS_{AB} \\
+ SS_E(f_1 c_1 + f_2 c_2 + f_{1,2} c_{1,2})
\]

subject to \( c \in \{0,1\}^4 \) \quad (19)

where

\[
f_1 = \frac{m_1 - 1}{N - m_1 m_2} F_0(m_1 - 1, N - m_1 m_2),
\]

\[
f_2 = \frac{m_2 - 1}{N - m_1 m_2} F_0(m_2 - 1, N - m_1 m_2),
\]
\[ f_{1,2} = \frac{(m_1 - 1)(m_2 - 1)}{N - m_1 m_2} F_\alpha \left( (m_1 - 1)(m_2 - 1), N - m_1 m_2 \right) \]

and \( F_\alpha(df_1, df_2) \) is the value taken from an F-table with \( df_1 \) and \( df_2 \) degrees of freedom and \( \alpha \) denoting the significance level. Clearly, (19) is the same as three separate hypothesis tests. If \( SS_A \) is large, larger than \( SS_E f_1 \), \( c_1 \) will be set to 1 to minimise the criterion. If \( SS_A < SS_E f_1 \), \( c_1 \) will be set to 0. Hence, we obtain the same choice of factors to include as in standard ANOVA.

Since \( c \) is a binary vector, there are of course many different ways in which \( c \) could enter (19), which would give the same result. For instance, \( c_i(c_i - 2) \) could be replaced by \( -c_i \). The reason for the choice in (19) is that it allows to make a connection to a more general optimisation problem, which will be studied in the following sections.

## 5 ANOVA-inspired optimisation problems

Having written ANOVA as a simple optimisation problem over \( c \), one could take a step further and try also to formulate optimisation problems over \( \theta \). In this section, two optimisation-based identification algorithms, inspired by the ANOVA method, will be presented. In fact, as we will see in Section 6.1, the ANOVA problem (19) can be obtained as a small modification of one of the algorithms, with \( c \) being binary variables.

### 5.1 Relaxed problem

Let us first define the objective function

\[ V(c, \theta) = \sum_{t=1}^{N} \left( y(t) - M(c, \theta, \varphi(t)) \right)^2 \]  \( (20) \)

where \( M(c, \theta, \varphi(t)) \) is a model of the system. Here, we will make use of an ANOVA expansion (6) with piecewise constant basis functions;

\[
M(c, \theta, \varphi(t)) = c_0 \theta_0 + \sum_{k=1}^{d} c_k \left( \sum_{i_1=1}^{m_k} \theta_{k,i_1} \mathbf{I}_{b_{k,i_1}}(\varphi_k) \right) + \sum_{k=1}^{d-1} \sum_{l=k+1}^{d} c_{k,l} \left( \sum_{i_1=1}^{m_k} \sum_{i_2=1}^{m_l} \theta_{k,i_1,i_2} \mathbf{I}_{b_{k,i_1}}(\varphi_k) \mathbf{I}_{b_{l,i_2}}(\varphi_l) \right) + \ldots + c_{1,2,d} \left( \sum_{i_1=1}^{m_1} \sum_{i_2=1}^{m_2} \ldots \sum_{i_d=1}^{m_d} \theta_{1,2,d,i_1,i_2,...,i_d} \prod_{k=1}^{d} \mathbf{I}_{b_{k,i_k}}(\varphi_k) \right) \]  \( (21) \)

where \( b_{k,i} \) is the \( i \)-th interval belonging to the regressor \( \varphi_k \) and \( \mathbf{I}_b(x) = 1 \) if \( x \in b \) and zero otherwise. In this expression \( c_0 \theta_0 \) is the total mean, the first sum consists of the main effects, the second sum consists of the two-factor interactions and so on. Note that each effect has its own parameter in \( c \), but can have many parameters in \( \theta \), one for each basis function. Also note that (21) can be interpreted as a straightforward generalisation of (2).
Being true to the ANOVA method, \( c \) should be chosen as a binary vector, \( c \in \{0, 1\}^d \). However, this would lead to mixed integer optimisation problems, which are generally hard to solve efficiently. Instead, let us relax the constraint and let \( c \) vary continuously in the interval \([0, 1]^d\).

Now, the algorithm can be written in two steps, both solving a convex optimisation problem (Boyd and Vandenberghe, 2004):

**Step 1** Solve

\[
\min_{\theta} \ V(1, \theta) \quad (22)
\]

subject to \( \mathbf{A} \theta = 0 \)

Denote the optimal solution by \( \hat{\theta} \). The problem can be solved analytically, as in Section 2.

**Step 2** Solve

\[
\min_{c} \ V(c, \hat{\theta}) + J\|\mathbf{F}c\|_1 \quad (23)
\]

subject to \( c \in [0, 1]^d \)

What is needed now is to specify \( \mathbf{A} \), \( J \) and \( \mathbf{F} \). \( \mathbf{A} \) is defined from the linear constraints coming from ANOVA, saying that the parameters \( \theta_{k_1, \ldots, k_l; i_1, \ldots, i_l} \) should sum to zero over any of its indices, i.e.,

\[
\sum_{i_{k_j}=1}^{m_{k_j}} \theta_{k_1, \ldots, k_l; i_1, \ldots, i_l} = 0 \quad \forall \ j = 1, \ldots, l. \quad (24)
\]

For \( J \), we choose \( J = V(1, \hat{\theta}) \). Note that this choice corresponds to \( SS_E \) in ANOVA.

The weight matrix \( \mathbf{F} \) is chosen to be diagonal and is computed from the degrees of freedom for each effect and the F statistical table. The weights for \( c_{k_1, \ldots, k_l} \) are

\[
f_{k_1, \ldots, k_l} = \frac{df_{k_1, \ldots, k_l}}{df_E} F_\alpha(df_{k_1, \ldots, k_l}, df_E), \quad (25)
\]

where \( df_{k_1, \ldots, k_l} = \prod_{i=1}^{l}(m_i - 1) \), \( df_E = N - \prod_{i=1}^{l} m_i \) and \( F_\alpha(df_1, df_2) \) is the value taken from an F-table with \( df_1 \) and \( df_2 \) degrees of freedom and \( \alpha \) denoting the significance level.

The obtained numerical values of the parameter vector \( \theta \) correspond to the values computed with ANOVA, which is no surprise since they are the same analytical problem.

In preliminary experiments, the numerical values of \( c \) often (but not always) include values exactly equal to zero, when the corresponding ANOVA tests say that the effect is insignificant (see Section 7). Elements exactly equal to one hardly ever occur, except for \( c_0 \) which is not penalised and hence always equal to one.

In Section 6.1 some more analysis of the relation between the presented algorithm and ANOVA will be given.
5.2 Linear in parameters

The optimisation problem can also be reparameterised with linear parameters. In this case we use the following model structure:

\[ \mathcal{M}(\vartheta, \varphi) = \vartheta_0 + \sum_{k=1}^{d} \left( \sum_{i_1=1}^{m_k} \vartheta_{k,i_1} I_{b(k,i_1)}(\varphi_k) \right) \]

\[ + \sum_{k=1}^{d-1} \sum_{i=k+1}^{d} \left( \sum_{i_1=1}^{m_k} \sum_{i_2=1}^{m_i} \vartheta_{k,i_1,i_2} I_{b(k,i_1)}(\varphi_k) I_{b(i_2)}(\varphi_l) \right) + \ldots \]

\[ + \sum_{i_1=1}^{m_1} \sum_{i_2=1}^{m_2} \ldots \sum_{i_d=1}^{m_d} \vartheta_{1,2,\ldots,d; i_1,i_2,\ldots,i_d} \prod_{k=1}^{d} I_{b(k,i_k)}(\varphi_k) \] (26)

The penalties \( F \) must then be extended to match the number of parameters. The idea here is to retain a penalty that has a similar effect as when penalising \( c \) in the previous section. For this, consider a parameter \( \vartheta_{k_1,\ldots,k_l; i_1,\ldots,i_{k_1}} \). This parameter corresponds to the product \( c_{k_1,\ldots,k_l} \theta_{k_1,\ldots,k_l; i_1,\ldots,i_{k_1}} \). It therefore seems reasonable to give the parameter the penalty

\[ \beta_{k_1,\ldots,k_l} f_{k_1,\ldots,k_l} \]

\[ \frac{1}{|\theta_{k_1,\ldots,k_l; i_1,\ldots,i_{k_1}}|} \] (27)

where \( f_{k_1,\ldots,k_l} \) is the penalty for the \( c_{k_1,\ldots,k_l} \) parameter from (25), and \( \beta_{k_1,\ldots,k_l} \) is some normalisation term. Here \( \theta_{k_1,\ldots,k_l; i_1,\ldots,i_{k_1}} \) is the solution to Step 1 below.

To determine \( \beta_{k_1,\ldots,k_l} \), note that the total penalty of the terms related to \( c_{k_1,\ldots,k_l} \) is

\[ \beta_{k_1,\ldots,k_l} f_{k_1,\ldots,k_l} \sum_{i_{k_1}=1}^{m_{k_1}} \ldots \sum_{i_{k_l}=1}^{m_{k_l}} \frac{\vartheta_{k_1,\ldots,k_l; i_{k_1},\ldots,i_{k_l}}}{\theta_{k_1,\ldots,k_l; i_{k_1},\ldots,i_{k_l}}} \]

which corresponds to

\[ \beta_{k_1,\ldots,k_l} f_{k_1,\ldots,k_l} \frac{1}{|c_{k_1,\ldots,k_l}|} \prod_{i=1}^{l} m_{k_i} \]

Therefore, it is natural to choose \( \beta_{k_1,\ldots,k_l} = \prod_{i=1}^{l} \frac{1}{m_{k_i}} \).

The minimisation should be performed subject to the linear constraints \( A\vartheta = 0 \), where \( A \) is the same matrix as in Section 5.1.

Now, we can perform the estimation in the following steps:

Step 1 Solve

\[ \min_{\vartheta} V(\vartheta) \] (28)

subject to \( A\vartheta = 0 \)

This is exactly the same problem as Step 1 in the previous section. Denote the optimal solution by \( \hat{\vartheta} \).

Step 2 Solve

\[ \min_{\vartheta} V(\vartheta) + J\|F\vartheta\|_1 \] (29)

subject to \( A\vartheta = 0 \)
with \( J = V(\hat{\theta}) \).

**Step 3** Fix the parameters that became zero in step 2, and solve the minimisation in Step 1 again, but with respect to the remaining, non-zero parameters. (In practice, because of numerical issues the parameters with a magnitude smaller than a given threshold should be set to zero.)

Note that the optimisation problems in all steps are convex. As we will see in Section 6, for balanced data the problem in Step 1 can be separated into several smaller optimisation problems, one for each set of parameters \( \{ \theta_{i_1,\ldots,i_k} \}_{i_{k_1}=1,\ldots,i_{k_2}=1} \).

This means that if in any such set, no parameters are set to zero, the solution for these parameters is equal to their solution in step 1, and this particular subproblem does not need to be solved again.

In practice, it seems like the suggested penalty is slightly too small. By multiplication by a factor 2, the results of our preliminary experiments have improved and seem very promising (see Section 7).

### 6 Some analysis

We will now see that an optimisation problem of the type described in Section 5 can be separated into simpler, independent problems, provided that the dataset is balanced. To begin with, consider the criterion function \( V(\mathbf{c}, \theta) \) from (20), and assume that the linear constraints from (24) are satisfied.

As in Section 2, reorder the data by grouping them according to which cells they belong to. Recall the notation \( \mathbf{b} \in \{ (j_1, j_2, \ldots, j_d) \mid j_k = 1, \ldots, m_k \} \) for the different cells, and write \( \varphi(\mathbf{b}, p), y(\mathbf{b}, p) \) for the \( p \)th data point in cell \( \mathbf{b} \).

Assume that the data set is balanced, and that there are \( N_b \) data points in each cell. We may then write \( V(\mathbf{c}, \theta) \) as

\[
V(\mathbf{c}, \theta) = \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \ldots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - M(\mathbf{c}, \theta, \varphi((j_1, \ldots, j_d), p)) \right)^2
\]

\[
= \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \ldots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_0 \theta_0 - \sum_{k=1}^{d} c_k \left( \sum_{t_1=1}^{m_k} \theta_{k,t_1} I_{b(k,t_1)}(\varphi_k((j_1, \ldots, j_d), p)) \right) \right.
\]

\[
- \sum_{k=1}^{d} \sum_{l=k+1}^{d} c_{k,l} \left( \sum_{t_1=1}^{m_k} \sum_{t_2=1}^{m_l} \theta_{k,t_1} \theta_{l,t_2} I_{b(k,t_1)} I_{b(l,t_2)}(\varphi_k((j_1, \ldots, j_d), p)) \right) - \ldots
\]

\[
- c_{1,2,\ldots,d} \left( \sum_{t_1=1}^{m_1} \sum_{t_2=1}^{m_2} \ldots \sum_{t_d=1}^{m_d} \theta_{1,t_1} \theta_{2,t_2} \ldots \theta_{d,t_d} I_{b(1,t_1)} I_{b(2,t_2)} \ldots I_{b(d,t_d)}(\varphi_1((j_1, \ldots, j_d), p)) \right)\]

\[
\left. \right)^2
\]

\[
(30)
\]
\[ \begin{align*}
&= \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_0 \theta_0 - \sum_{k=1}^d c_k \theta_{k;j_k} \right. \\
&\quad - \sum_{k=1}^{d-1} \sum_{l=k+1}^d c_{k,l} \theta_{k,l;j_k;j_l} - \cdots - c_{1,2,\ldots,d} \theta_{1,2,\ldots,d;j_1,\ldots,j_d} \right)^2 \\
&= \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_0 \theta_0 \right)^2 \\
&\quad + \sum_{k=1}^d \left( \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_k \theta_{k;j_k} \right)^2 \right) \\
&\quad + \sum_{k=1}^{d-1} \sum_{l=k+1}^d \left( \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_{k,l} \theta_{k,l;j_k;j_l} \right)^2 \right) \\
&\quad + \cdots \\
&\quad + \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_{1,2,\ldots,d} \theta_{1,2,\ldots,d;j_1,\ldots,j_d} \right)^2 \\
&\quad + \text{lots of terms that sum to zero because of the linear constraints} \\
&= (2^d - 1) \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} y^2((j_1, \ldots, j_d), p) \quad (31)
\end{align*} \]

Note that the the parameters in the last expression are separated into \(2^d\) different sums: One containing only \(c_0\) and \(\theta_0\), one containing only \(c_1\) and \(\theta_{1;j_1}\), \(j_1 = 1, \ldots, m_1\), etc. We can draw several conclusions from these calculations:

- For balanced data, minimisation of \(V(c, \theta)\) with respect to either \(c\) or \(\theta\), under the linear constraints used in ANOVA, can be done by separately minimising the \(2^d\) sums of squared residuals in the last expression, under corresponding constraints.

- The same kind of computations also hold for the reparameterisation into a linear parameterisation, as presented in Section 5.2.

- The same holds for every regularised version of \(V(c, \theta)\) (or \(V(\vartheta)\)) where the penalties can be separated in the same way, i.e., be written as \(g_0(c_0, \theta_0) + g_1(c_1; \{(\theta_{1;i})_{i=1}^{n_1}\} + \cdots\). This is the case for the methods considered in Section 5.

If the data set is unbalanced, we cannot separate the problem into many smaller optimisation problems in the same way, since the cross-terms do not cancel. The unbalanced case is discussed further in Section 8.
6.1 Connection to ANOVA

If the terms from (30) are collected in another manner than above it is possible to get

\[
V(\mathbf{c}, \theta) = \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \left( y((j_1, \ldots, j_d), p) - c_0 \theta_0 \right)^2 \\
+ \sum_{k=1}^{d} c_k(c_k - 2) \sum_{j_{k_1}=1}^{m_{j_{k_1}}} \theta_{k;j_{k_1}}^2 N_b \left( \prod_{i=1}^{d} m_i \right) \\
+ \sum_{k=1}^{d-1} \sum_{l=k+1}^{d} c_{k,l}(c_{k,l} - 2) \sum_{j_{k_1}=1}^{m_{j_{k_1}}} \sum_{j_{l_1}=1}^{m_{j_{l_1}}} \theta_{k,l;j_{k_1},j_{l_1}}^2 N_b \left( \prod_{i=1}^{d} m_i \right) \\
+ \ldots \\
+ c_{1,2,\ldots,d}(c_{1,2,\ldots,d} - 2) \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \theta_{1,2,\ldots,d;j_1,j_2,\ldots,j_d}^2 \\
+ \text{lots of terms that sum to zero because of the linear constraints}
\]

\[
+ \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \cdots \sum_{j_d=1}^{m_d} \epsilon(\theta, (j_1, \ldots, j_d), p) \left( \sum_{k=1}^{d} c_k \theta_{k;j_k} \\
+ \sum_{k=1}^{d-1} \sum_{l=k+1}^{d} c_{k,l} \theta_{k,l;j_k,j_l} + \ldots + c_{1,2,\ldots,d} \theta_{1,2,\ldots,d;j_1,j_2,\ldots,j_d} \right),
\]

(32)

where

\[
\epsilon(\theta, (j_1, \ldots, j_d), p) = y((j_1, \ldots, j_d), p) - \mathcal{M}(1, \theta, \varphi((j_1, \ldots, j_d), p))
\]

are the residuals from the model with \( \mathbf{c} = 1 \). For the two-dimensional case, the expression (32) corresponds to

\[
V(\mathbf{c}, \theta) = SS_T + c_1(c_1 - 2)SS_A + c_2(c_2 - 2)SS_B + c_{1,2}(c_{1,2} - 2)SS_{AB} \\
+ \sum_{p=1}^{N_b} \sum_{j_1=1}^{m_1} \sum_{j_2=1}^{m_2} \epsilon(\theta, (j_1, j_2), p) (c_1 \theta_{1;j_1} + c_2 \theta_{2;j_2} + c_{1,2} \theta_{1,2;j_1,j_2}).
\]

(34)

The last term is zero if the model is the mean value in each cell \( \mathbf{b} \), which is the case for the solution to the unpenalised problem. The constraints

\[
\sum_{p=1}^{N_b} \left( y((j_1, \ldots, j_d), p) - \mathcal{M}(1, \theta, \varphi((j_1, \ldots, j_d), p)) \right) = 0, \; \forall \; j_1, j_2, \ldots, j_d,
\]

(35)

describe this condition. Note that this is also the reason why ordinary ANOVA has an orthogonality property for balanced data.

In Section 5.1 we noted that values of \( c_x \) exactly equal to zero are common, while values exactly equal to one are scarce. The reason can be seen in Figure 1, where \( c_x(c_x - 2) \), the function of \( c_x \in [0, 1] \) in front of each term in (32), is
plotted. Since the derivative of the function is flat close to \( c_x = 1 \), while it is steep close to \( c_x = 0 \), this means that the cost difference between \( c_x \) being close to one or exactly one is negligible, while around zero it really matters whether \( c_x \) is exactly zero or just close.

We now turn back to the method in Section 5.1, and rewrite it for the two-dimensional case (34). In step 1, the optimal \( \hat{\theta} \) will make the last term of (34) equal to zero. In step 2, if we use the constraint \( c \in \{0, 1\}^4 \), we are back to the optimisation problem (19), which is equal to ANOVA in its original formulation. This means that for balanced data, the method in Section 5.1 can be seen as a direct relaxation of ANOVA.

7 Example

Example 1 To illustrate the relaxed ANOVA problem in Section 5.1 and the linear in parameters problem in Section 5.2, let us consider a non-trivial example of a NARX model structure. The model was used in Yao and Tong (1994) and Lind (2006, Section 6.2.7). The output is given by

\[
y(t) = 0.3y(t-1)e^{u(t-1)} + \sin(u(t-1)) + e(t),
\]

where \( u(t) \) is the output from an AR(2) model:

\[
u(t) = 0.1u(t-1) - 0.56u(t-2) + n(t).
\]

The noise term \( e(t) \) has the same distribution as the noise term \( 0.6n(t) \). \( n(t) \) is the sum of 48 independent uniformly distributed random variables, in the range \([-0.25, 0.25]\). According to the central limit theorem the noise terms can then be treated as coming from an approximate Gaussian distribution, but with the support bounded to \([-12, 12]\). The candidate regressors \( \varphi_1(t) = y(t-1) \), \( \varphi_2(t) = y(t-2) \), \( \varphi_3(t) = u(t) \) and \( \varphi_4(t) = u(t-1) \) are assumed. The model contains two-factor interaction and the candidate regressors are correlated.

In the experiments, 3000 data points were collected. Three intervals for categorisation were chosen separately for each regressor, such that one third of the
data set falls into each interval. Then the data set was balanced to give at most 8 data points in each cell. The resulting (almost) balanced data set consisted of 644 data points. ANOVA, the relaxed ANOVA and linear in parameters methods were run on this data set, using twice the penalty compared to the algorithms in Sections 5.1 and 5.2. The results are presented in Table 1. We can see that both ANOVA and the relaxed ANOVA succeed in finding the true model structure, and that \( c \) is large where the ANOVA \( p \) is small (and vice versa). For the linear in parameters method, the resulting structure is sparse as desired. Furthermore, the model structure almost corresponds to the result of the other two methods. The price to be paid for the sparseness can be measured in the sum of the squared residuals, which is 505 for the full least squares solution and 551 for the sparse, linear in parameters solution. The corresponding value for the relaxed ANOVA is 571. On a validation data set from the same system, the squared residuals for the full least squares solution is 583 while the squared residuals for the linear in parameters solution and for the relaxed ANOVA are 563 each. This means that the sparse structures have better generalisation ability than the full locally constant model.

Table 1: Results from ANOVA, relaxed ANOVA and the linear in parameters method. The ANOVA \( p \) is the probability of the null hypotheses in the ANOVA F-tests. Next column shows \( c \) from (23). \( \max |\varphi| \) stands for the values \( \max |c_{k_1,...,i_{k_1},k_2,...,i_{k_2}}| \) for each element in \( c \). The second last column shows the resulting number of nonzero parameters for each effect in the linear in parameters method, compared to the total number of parameters. The last column shows \( |\varphi_{k_1,...,i_{k_1},k_2,...,i_{k_2}}| \) for the linear in parameters method.

| Effect                   | ANOVA \( p \) | \( c \) | \( \max |\varphi| \) | \# \( \varphi \neq 0 \) | \( \max |\varphi| \) |
|--------------------------|----------------|-------|-------------------|-----------------|----------------|
| Constant                 | 0.00           | 1.00  | 0.13              | 1/1             | 0.13           |
| \( \varphi_1 \)          | 0.00           | 0.97  | 0.55              | 2/3             | 0.57           |
| \( \varphi_2 \)          | 0.35           | 0.00  | 0.00              | 0/3             | 0              |
| \( \varphi_3 \)          | 0.33           | 0.00  | 0.00              | 0/3             | 0              |
| \( \varphi_4 \)          | 0.00           | 0.97  | 0.70              | 2/3             | 0.70           |
| \( (\varphi_1, \varphi_2)\) | 0.04           | 0.05  | 0.01              | 4/9             | 0.15           |
| \( (\varphi_1, \varphi_3)\) | 0.77           | 0.00  | 0.00              | 0/9             | 0              |
| \( (\varphi_1, \varphi_4)\) | 0.00           | 0.95  | 0.73              | 6/9             | 0.76           |
| \( (\varphi_2, \varphi_3)\) | 0.36           | 0.00  | 0.00              | 0/9             | 0              |
| \( (\varphi_2, \varphi_4)\) | 0.13           | 0.00  | 0.00              | 4/9             | 0.14           |
| \( (\varphi_3, \varphi_4)\) | 0.24           | 0.00  | 0.00              | 0/9             | 0              |
| \( (\varphi_1, \varphi_2, \varphi_3)\) | 0.54           | 0.00  | 0.00              | 0/27            | 0              |
| \( (\varphi_1, \varphi_2, \varphi_4)\) | 0.80           | 0.00  | 0.00              | 0/27            | 0              |
| \( (\varphi_1, \varphi_3, \varphi_4)\) | 0.03           | 0.06  | 0.02              | 8/27            | 0.23           |
| \( (\varphi_2, \varphi_3, \varphi_4)\) | 0.88           | 0.00  | 0.00              | 0/27            | 0              |
| \( (\varphi_1, \varphi_2, \varphi_3, \varphi_4)\) | 0.78           | 0.00  | 0.00              | 0/81            | 0              |

8 Discussion

It turns out that the relaxed versions of ANOVA in Section 5 have tight connections to the nn-garrote and wavelet shrinkage (Sections 3.1 and 3.5 respectively).
These connections are explored below. Section 8.3 also discusses the implications of unbalanced data.

8.1 ANOVA and nn-garrote

Note the similarities between (10) and the relaxed problem in Section 5.1. The difference is that the penalty on \( c_k \) is \( \lambda P_k \) in the group nn-garrote, while in Section 5.1 the penalty for \( c_k \) is

\[
\sum_{t=1}^{N} \left\| y(t) - M(1, \hat{\theta}, \varphi(t)) \right\|^2_2 \cdot f_{k_1, \ldots, k_l},
\]

with \( f_{k_1, \ldots, k_l} \) defined in (25). We also have a specific choice of \( X(t) \) in Section 5.1, e.g.,

\[
X_1(t) = \left[ I_{b_1, 1}(\varphi_1(t)) \ I_{b_1, 2}(\varphi_1(t)) \ I_{b_2, 1}(\varphi_1(t)) \ I_{b_2, 2}(\varphi_1(t)) \right]^T,
\]

together with the linear constraints on the parameters (24).

Hence, with small modifications of the group nn-garrote method, it could be used to solve relaxed ANOVA problems. These modifications concern the penalty terms, and also restrict the choice of \( X(t) \). To the authors’ knowledge, this connection has not been made before.

8.2 Linear parameterisation as wavelets

Instead of using linear constraints on the parameters it is possible to use basis functions that automatically fulfills the constraints. For example, the wavelet Haar basis functions (Hastie et al., 2001) can be used to incorporate the constraints (24).

**Example 2** Consider the linear parameterisation in Section 5.2, with only one candidate regressor \( \varphi(t) \). We use four cells, which gives

\[
M(\theta, \varphi) = \theta_0 + \sum_{i=1}^{4} \theta_{1,i} I_{b_1,i}(\varphi),
\]

(38)

with the constraints \( \sum_{i=1}^{4} \theta_{1,i} = 0 \). This model can also be represented by the Haar basis in Figure 2. The expansion in Haar basis can be written

\[
M(\eta, \varphi) = \eta_0 + \eta_{0,0} \psi_{0,0}(\varphi) + \eta_{1,0} \psi_{1,0}(\varphi) + \eta_{1,1} \psi_{1,1}(\varphi)
\]

(39)

The relation between the parameters is then \( \vartheta_0 = \eta_0 \) and

\[
\begin{align*}
\vartheta_{1,1} &= \eta_{0,0} + \eta_{1,0} \\
\vartheta_{1,2} &= \eta_{0,0} - \eta_{1,0} \\
\vartheta_{1,3} &= -\eta_{0,0} + \eta_{1,1} \\
\vartheta_{1,4} &= -\eta_{0,0} - \eta_{1,1},
\end{align*}
\]

(40)

which obviously sums to zero. In the typical wavelet estimation situation the minimisation criterion is chosen as (13):

\[
\min_{\eta} \sum_{t=1}^{N} \left\| y(t) - M(\eta, \varphi(t)) \right\|^2 + 2\lambda \| \eta \|_1.
\]

(41)
If we now compare the different models, we see that this corresponds to penalising $|\vartheta_{1,1} + \vartheta_{1,2}|/2$, $|\vartheta_{1,1} - \vartheta_{1,2}|/2$ and $|\vartheta_{1,3} - \vartheta_{1,4}|/2$, instead of $|\vartheta_{1,1}|$, $|\vartheta_{1,2}|$, $|\vartheta_{1,3}|$ and $|\vartheta_{1,4}|$ as in Section 5.2. Intuitively, when using the criterion in (13), we strive to minimise the number of nonzero coefficients in front of the Haar basis functions, while when using penalties as in Section 5.2, we aim at minimising the number of intervals with a nonzero main effect (in this, one-dimensional case).

To conclude, it is possible to express the criterion function $V(\vartheta)$ in (28) using the Haar basis wavelets in a ANOVA expansion (6). However, the penalties used in wavelet shrinkage and in Section 5.2 differ from each other. By a simple transformation of the penalties, they can be made equivalent.

8.3 Unbalanced data

In an unbalanced design the independence between the different sums of squares, Equation (4), is lost. For ANOVA in its original formulation, it has been proposed that the F-tests should be done in a different manner (Miller, 1997). An exact treatment of a badly unbalanced design (when the number of measurements in each cell differ by a factor 10 or more) is especially hard to interpret. A computationally efficient solution with easy interpretation (suggested in Montgomery (1991)) is to randomly discard excess data in the superfluous cells until a balanced design is obtained. At least two measurements in each cell should be kept. Drawbacks of this solution are that it does not make use of all data and that it might be sensitive to outliers. It may very well be that the best results are obtained by keeping the dataset unbalanced to a certain extent (see Lind
For the methods presented in Section 5, we could solve the optimisation problems also with unbalanced data and get a unique solution (as long as there is at least one data point in each cell). However, the separation between the different groups of parameters analysed in Section 6 is lost, since the cross-terms in (31) will not cancel. This means that the comparisons with the F-tests employed in standard ANOVA no longer hold. Note also that for the same reasons, the F-tests in ANOVA are correlated for unbalanced data, and give different results depending on in which order they are performed, in contrast to the optimisation-based methods.

The loss of separation however does not necessarily imply that the models obtained by the optimisation-based methods are bad for unbalanced data. In preliminary experiments, no such evidence has been found. The issue on whether or not to balance the dataset is further discussed in Lind (2006).

9 Conclusions

In this paper, it has been shown that ANOVA can be formulated as an optimisation problem, which has a natural relaxation (Section 5.1) that can be efficiently computed. As seen in Section 7, the relaxed method shows promising performance. More thorough theoretical and experimental investigations are topics for future research, but as noted in Section 8.1, the method is very closely related to the group nn-garrote method that is shown to have nice properties in terms of selecting the true regressors (Yuan and Lin, 2005).

In Section 5.2, another related method with a linear-in-parameters structure was presented, and it turned out that it had close connections to Haar basis wavelets. In the example of Section 7, a sparse structure was obtained just as intended, at the cost of a small increase in the sum of squared residuals.

For balanced data, an advantage of both the proposed methods is that they share an orthogonality property, meaning that the optimisation problems can be decomposed into smaller problems that can be solved independently of each other.

A possible interpretation of the close connection between the grouped nn-garrote and the relaxed method of this paper would be that we can get a hint from ANOVA on how to choose the size of the penalties in nn-garrote, for the particular choice of basis functions. An interesting question to consider is how this choice could be generalised to other kinds of basis functions.

10 Acknowledgments

This work has been supported by the Swedish Research Council (VR) which is gratefully acknowledged.
References


