Eilers, Heim, Marx:

Varying Coefficient Tensor Models for Brain Imaging

Sonderforschungsbereich 386, Paper 436 (2005)

Online unter: http://epub.ub.uni-muenchen.de/

Varying Coefficient Tensor Models for Brain Imaging

Paul H.C. Eilers, Susanne Heim, and Brian D. Marx

[1] Department of Medical Statistics, Leiden Univerity Medical Center, 2300 RA, Leiden, The Netherlands
[2] Susanne Heim, Institute of Statistics, Ludwig-Maximilians-University, D-80539 Munich, Germany
[3] Brian D. Marx, Department of Experimental Statistics, Louisiana State University, Baton Rouge, LA 70803 USA

Abstract:
We revisit a multidimensional varying-coefficient model (VCM), by allowing regressor coefficients to vary smoothly in more than one dimension, thereby extending the VCM of Hastie and Tibshirani. The motivating example is 3-dimensional, involving a special type of nuclear magnetic resonance measurement technique that is being used to estimate the diffusion tensor at each point in the human brain. We aim to improve the current state of the art, which is to apply a multiple regression model for each voxel separately using information from six or more volume images. We present a model, based on P-spline tensor products, to introduce spatial smoothness of the estimated diffusion tensor. Since the regression design matrix is space-invariant, a 4-dimensional tensor product model results, allowing more efficient computation with penalized array regression.

Keywords: Brain imaging; P-splines; Varying coefficient model

1 Introduction

Varying-coefficient models (VCM) were introduced by Hastie and Tibshirani (1993) to handle situations in which regression coefficients vary smoothly (interact) with another variable. Unlike standard regression, where coefficients are assumed to be constant, VCMs produce coefficients that are smooth curves, e.g. reflecting slow change over time or space. Hastie and Tibshirani used iterative backfitting to estimate these curves, but Eilers and Marx (2002) showed how to use P-splines to eliminate backfitting, in effect estimating all varying coefficient terms simultaneously. In this paper, we show how to use multidimensional P-splines (tensor products of B-splines, combined with roughness penalties) to construct VCMs with coefficients that smoothly change in 3-dimensional space.
The motivating application, i.e. estimation of the the diffusion tensor in brain images from data obtained at a set of magnetic gradients, as well

as a proposal for applying varying coefficient tensor models (VCTM) are described in Heim et al. (2004). Presently linear regression is applied in each image voxel separately; the six unique elements of the diffusion tensor correspond to six coefficients in the regression. Noise can be reduced, and possibly also acquisition time, if the coefficients can be forced to change smoothly in space.
To allow enough detail in the model, relatively large numbers of B-spline knots are needed. Also the number of the data cubes ($M = 128 \times 128 \times 24$) is large. We propose to implement an efficient method for smoothing on grids with tensor products that was developed by Eilers, Currie, and Durbàn (2005). Such an approach avoids the computation of large Kronecker products of B-spline bases, and with a slight modification, this algorithm turns the 3-dimensional VCM into smoothing with 4-dimensional tensor products. In the next section we briefly describe the physical background. The VCM model and penalized estimation are described in Section 3. Array regression is presented in Section 4. An illustrative example, and a variety of details are discussed in Sections 5 and 6, respectively.

2 Diffusion tensor (DT) imaging

In general, the data basis consists of a set of diffusion weighted images $\{S_r : r = 1, \ldots, R\}$ and a non–weighted reference image S_0 at each of M voxels of a specified 3-dimensional volume. The voxels are indexed in space by $\mathbf{s} = (s_1, s_2, s_3)$. Each response image is acquired at a different uniform magnetic gradient. Diffusion is recorded along each gradient $\mathbf{g}_r = (g_{1r}, g_{2r}, g_{3r})', r = 1, \ldots, R$, as attenuation of the radio frequency signals which are emitted by the atomic nuclei of the randomly travelling water molecules in a magnetic resonance tomograph. The resulting covariance ellipsoid of the local diffusivity is characterized by the voxel-wise diffusion tensor

$$\mathbf{D}(\mathbf{s}) = \begin{pmatrix} D_1(\mathbf{s}) & D_4(\mathbf{s}) & D_5(\mathbf{s}) \\ D_4(\mathbf{s}) & D_2(\mathbf{s}) & D_6(\mathbf{s}) \\ D_5(\mathbf{s}) & D_6(\mathbf{s}) & D_3(\mathbf{s}) \end{pmatrix}.$$

Neuronal fibers in white matter are densely packed and highly ordered such that the water molecules therein preferentially pass along the biophysiological structures instead of perpendicular to them. Conversely, we determine the principal diffusion direction with the dominant eigenvector of the local diffusion tensor and identify it with the local fiber orientation in space.
The relation of the measured signal loss $S_r(\mathbf{s})$ in voxel $\mathbf{s}$ and applied gradient $\mathbf{g}_r$ is given by the Stejskal-Tanner equation

$$S_r(\mathbf{s}) \quad = \quad S_0(\mathbf{s}) \exp\{-b\, \mathbf{g}_r{}' \mathbf{D}(\mathbf{s}) \mathbf{g}_r\}\ , \quad r = 1, \ldots, R\,. \tag{1}$$

Here, b is a scalar comprising several acquisition parameters such as gradient strength and duration.

To fully determine the six free parameters of the DT in (1), one has to apply gradients in $r \geq 6$ independent directions. Usually, measurements are repeated for the same gradient set or the number of different gradients are chosen to be over determined in order to mitigate the effects of noise. Following Papadakis et al. (1999), (1) can be reformulated, for voxel $\mathbf{s}$, as

$$y_r(\mathbf{s}) = -\frac{1}{b}\log\left(\frac{S_r}{S_0}(\mathbf{s})\right) \quad = \quad \mathbf{x}_r{}'\beta(\mathbf{s})\,, \tag{2}$$

with the unknown elements of the diffusion tensor $\beta(\mathbf{s}) = (D_1, D_2, D_3, D_4, D_5, D_6)'(\mathbf{s})$ and a nonlinear mapping of the gradient to a six-dimensional vector $\mathbf{x}_r = (g_{1r}^2, g_{2r}^2, g_{3r}^2, 2g_{1r}g_{2r}, 2g_{1r}g_{3r}, 2g_{2r}g_{3r})'$. In (2) we recognize a linear regression problem, in which the 6-dimensional vector $\beta(s)$ has to estimated for each voxel, using 6 or more images obtained at different magnetic gradients. Note that $\mathbf{x}_r$ is the same for all voxels, and thus we face a repeated measures design.

3 The varying-coefficient model

The current state-of-the-art method of estimating the diffusion matrix from DT imaging experiments is through voxel-wise multiple regressions. Yet the voxel spatial structure encourages an assumption of smooth coefficients. A major benefit of smoothed coefficients is that the DT can be approximated at any arbitrary position within the recorded volume. Consequently, a fiber tracking algorithm is allowed to operate on a markedly finer grid, which should result in more precise neuronal tract reconstruction. Hence, the β_t ($t = 1, \ldots, T = 6$) could be preferably modelled non-parametrically via penalized tensor product B-splines or multidimensional P-splines.
We first present a direct VCM presentation. Consider

$$\beta_t(\mathbf{s}) = \sum_{v=1}^{KLH} T_v(\mathbf{s})\gamma_{tv} = \mathbf{T}(\mathbf{s})\gamma_t, \tag{3}$$

where $\mathbf{T}(\mathbf{s})$ denotes the 3-dimensional tensor product of 1-dimensional B-splines, evaluated at voxel $\mathbf{s}$, γ_t is the vector of unknown amplitudes, and $K \times L \times H$ is determined by the (generous and regularly gridded) knot partition and degree of the basis function.
Using tensor coefficient expression in (3), we aim to find a practical solution to the penalized objective

$$Q = \|y - \sum_{t=1}^{T} \Upsilon_{x_t(\mathbf{s})}\mathbf{T}(\mathbf{s})\gamma_t\|^2 + \text{Penalty}, \tag{4}$$

where $\Upsilon_{x_t(\mathbf{s})} = \text{diag}\{x_t(\mathbf{s})\}$. The penalty term places difference penalties on the rows, columns, and layers of tensor product coefficients, for each regressor t, such that

$$\text{Penalty} = \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{h=1}^{H}\{\lambda_1(\Delta_1^d\gamma_{tklh})^2 + \lambda_2(\Delta_2^d\gamma_{tklh})^2 + \lambda_3(\Delta_3^d\gamma_{tklh})^2\}. \tag{5}$$

Here Δ_1^d (Δ_2^d, Δ_3^d) denotes the d-th order differences across a row (down a column, along a layer) of the $K \times L \times H$ matrix of tensor product B-spline coefficients, $\Gamma_t = [\gamma_{klh}]$. In theory, an explicit solution to $\gamma = (\gamma_1', \ldots, \gamma_T')'$ can be found using

$$\hat{\gamma} = (\mathbf{U}'\mathbf{U} + \mathbf{P})^{-1}\mathbf{U}'y, \tag{6}$$

where $\mathbf{U} = [\Upsilon_{x_1(\mathbf{s})}\mathbf{T}, \ldots, \Upsilon_{x_T(\mathbf{s})}\mathbf{T}]$, and $\mathbf{P}$ consists of a carefully arranged matrix representation (using block diagonal matrices of Kronecker products) of the differences found in (5). Given γ, then the varying coefficient volumes can be built. For practical brain imaging applications, we may need, e.g., $K \times L \times H = 32 \times 32 \times 8$ knots, thus $\mathbf{T}$ has approximately 3×10^9 elements: if each floating point takes 8 bytes, then $\mathbf{T}$ will use several of Gb memory, which is beyond reach of current computers.
As the straight-forward approach to VCM runs into difficulties, we can alleviate computation by taking advantage of the repeated measures structure, i.e. x does not depend on $\mathbf{s}$ or vary across voxels. Since the x are on a grid, the mean can be expressed as a tensor product involving x. Hence we can apply the fast, compact smoothing algorithm of Eilers, Currie, and Durbàn (2005), which uses a relatively moderate amount of memory, making the brain VCM model tractable again.

4 Array regression applied to brain imaging data

To simplify the presentation, we will consider a VCM that models only data in one dimension, e.g. assuming smooth regression coefficients along a line of voxels. We then will move into two dimensional smooth coefficients using slice images. To start, we first present the current approach of voxel-wise multiple regression. For each voxel, we model the mean $E(y_r) = \sum_{t=1}^{T} x_{rt}\beta_t$ or $E(y) = X\beta$, where in our example $r = 1, \ldots, R$ and $T = 6$ given by the magnetic field gradients. The matrix X has dimension $R \times T$, where $R \geq T$, with normal equations $X'X\hat{\beta} = X'y$.
With a line of I voxels, the y vectors can be placed into a matrix Y of dimension $R \times I$, as well as the coefficients into a matrix A of dimension $T \times I$. Thus $Y = XA + E$, where E is a matrix of random (normal errors). To ensure smoothness on the rows of A, we project each row onto a (common) B-spline basis, and hence $A' = B\Gamma$, where B ($I \times K$) of degree q is built along the index $1, \ldots, I$, and Γ is of dimension $K \times T$. Thus we have $E(Y) = X\Gamma'B'$, which shows directly the tensor structure of X and B. There is only one set of penalties on Γ: one penalty for each row. Thus the penalty term in (5) is greatly simplified, as it only involves sums over t and k.

With 2-dimensional brain slices, instead of voxels, we conventionally think of pixels indexed by $i = 1, \ldots, I$ and $j = 1, \ldots, J$. Standard matrix notation now breaks down, as Y and Γ become 3-dimensional arrays. The R images at each pixel are denoted by $Y = [y_{rij}]$ $r = 1, \ldots, R$ and the gradient information by $X = [x_{rt}]$, $t = 1, \ldots, T$. If we arrange Y as a matrix with R rows and IJ columns (each column corresponding to one pixel), we can write $Y = XA + E$, with a $T \times IJ$ coefficient matrix $A = [\alpha_{tij}]$ and E is a $R \times IJ$ matrix of random (normal) errors. In this format, pixel-wise regression would simply lead to the normal equations $X'X\hat{A} = X'Y$. We again impose spatial smoothness of the elements of A by projecting them onto $(K \times L)$ tensor products of B-splines. Let $B = [b_{ik}]$ and $\tilde{B} = [\tilde{b}_{jl}]$ $(k = 1, \ldots, K;\ l = 1, \ldots, L)$ be B-spline bases, of degree q, along the two axes of the image. Thus $\alpha_{tij} = \sum_k \sum_l b_{ik}\tilde{b}_{jl}\gamma_{klt}$. For simplicity we use the same tensor product B-spline basis for each t. This leads to

$$\mu_{rij} = E(y_{rij}) = \sum_{t=1}^{T} x_{rt} \sum_{k=1}^{K}\sum_{l=1}^{L} b_{ik}\tilde{b}_{jl}\gamma_{tkl}, \tag{7}$$

in which, by virtue of space-invariant x, we recognize a 3-dimensional tensor product model. Again, we ensure further smoothness through additional difference penalties on the coefficients of the tensor products. The simplified version of (5) is written as

$$\text{Penalty} = \sum_{t=1}^{T}\{\lambda_1 \sum_{k=1}^{K}\sum_{l=1}^{L}(\Delta_1^d \gamma_{tkl})^2 + \lambda_2 \sum_{k=1}^{K}\sum_{l=1}^{L}(\Delta_2^d \gamma_{tkl})^2\}. \tag{8}$$

Here Δ_1^d (Δ_2^d) denotes the d-th order differences along a row (down a column) of the $T \times K \times L$ array of tensor coefficients, $\Gamma = [\gamma_{tkl}]$.
Eilers et al. (2005) showed how to fit a multidimensional tensor product model with penalties efficiently. Their method adapts to any number of dimensions, and thus can be applied to the original 3-dimensional DT varying coefficient model. Lack of space does not allow us to go into the details of 3-dimensional extensions, but can be found in Eilers et al. (2005). In this setting, both Y and Γ will become 4-dimensional arrays, and the B-spline basis will come in a triple: B, $\tilde{B}$, $\breve{B}$. The mean of Y can again be expressed in a tensor form, $E(Y) = X\Gamma^\star B^\star$, where $\Gamma^\star$ and $B^\star$ are both carefully arranged matrices involving Kronecker products.

5 Illustrative example

The example consists of the 128×128 middle axial slice of a brain volume of diffusion tensor images. First we consider 1-dimensional VCM models. Figure 1 (top panel) shows the response signals for the $75th$ row of the brain slice and the varying coefficients for the diagonal tensor elements (middle panel) and the off-diagonal elements (bottom panel). Figure 2 shows

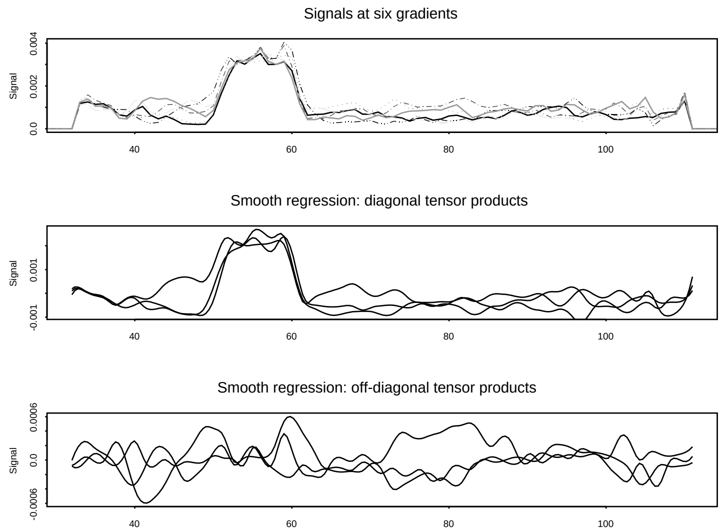

FIGURE 1. Signals at six gradients and smoothed DT coefficients.

the corresponding coefficients using pixel-wise multiple regression. Figure 3 (left, top and bottom) presents example 32×32 images from the first and third gradients. The center panels present the diagonal $D_1(\mathbf{s})$ (center) and the off-diagonal $D_4(\mathbf{s})$ element pixel-wise. The right panels presents the corresponding DT elements computed with VCM models and interpolated to a 128×128 resolution; thus demonstrating a benefit from an analytic P-spline approximation.

6 Details

In all cases, cubic B-splines, and a second order difference penalty, with the penalty parameter set to 0.01, were used. We used a knot at every second pixel for both the 1D and 2D cases (in the latter we analyzed only a 32×32 image). Unfortunately, the demands on computer memory and computation time load increase sharply with knot density. For 3D images of the original size it will essentially be impossible to explicitly form and solve the penalized normal equation. We would have to deal with about 50 by 50 by 10 by 6 knots, or 1.5 10^5 equations. We will research direct iterative methods to circumvent theses difficulties. Our approach naturally accommodates observation weights, so that missing data does not present a problem, and adjustments can be made for voxels with uncertainty, e.g.

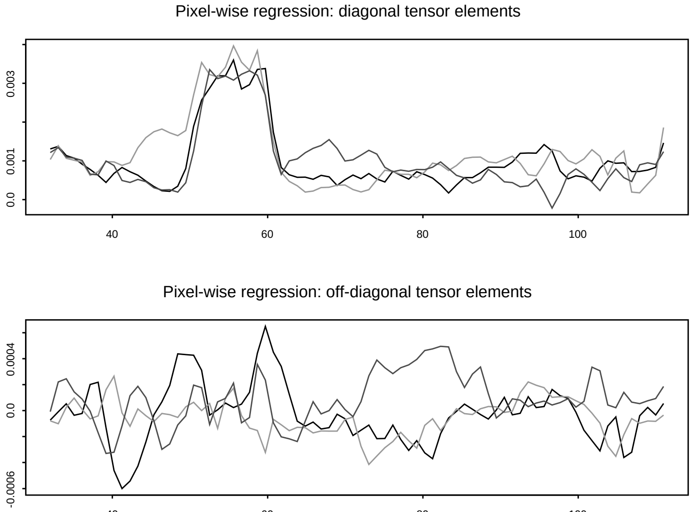

FIGURE 2. Estimated pixel-wise DT coefficients.

at the boundaries. Further as parameter estimation is grounded in classical methods, model summaries, such as: effective dimension and information criteria, and delete-one diagnostics, such as cross-validation (CV) are easily obtained. Accessible CV can also be a practical matter when searching for the "optimal" penalty tuning parameters, e.g. when isotropy is relaxed. Anisotropic penalization can be achieved by placing different λs on the indexing axes of Γ, leading to a higher dimensional grid-search. We are currently researching methods to optimally choose the penalty parameters (λ), including an E-M algorithm that alternates between: a) estimating the residual variance and the $\text{var}(\hat{\gamma}_t)$ to update penalty parameters, and b) estimating Γ with these updated penalty parameters. Since we are ultimately interested in plotting the smooth coefficients, we are also investigating choosing λ based on mean square error of the coefficient estimates, as an alternative prediction oriented criteria. Further, our penalized VCM approach can be transplanted into a generalized linear model framework, e.g. smoothing coefficients associated with a binary or threshold response.

References

Basser, P.J and Jones, D.K. (2002). Diffusion-tensor MRI: theory, experimental design and data analysis – a technical review. *NMR Biomed*, **15**, 456-467.

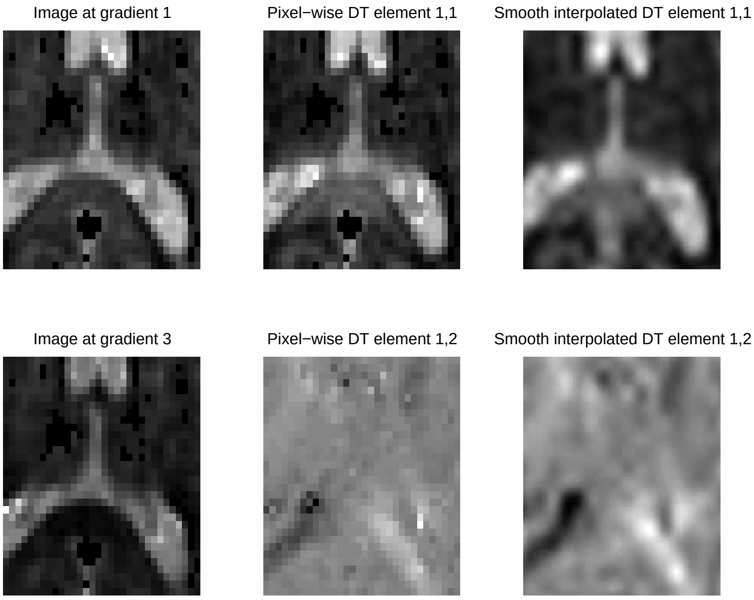

FIGURE 3. Two 32 by 32 pixel example images (left, top and bottom), two DT elements, as computed by pixel-wise regression (middle, top and bottom) and the same DT elements as computed with VCM model, and interpolated to a 128 by 128 grid (right, top and bottom).

Eilers, P.H.C, Currie, I., and Durbàn, M. (2005). Low memory, high speed smoothing on large multidimensional grids. *Computational Statistics and Data Analysis*. In Press.

Eilers, P.H.C. and Marx, B.D. (2002). Generalized linear additive smooth structures. *J. of Computat. and Graphical Statistics*, **11**(4), 758-783.

Hastie, T. and Tibshirani, R. (1993). Varying-Coefficient Models (with discussion). *Journal of the Royal Statistical Society* B **55**: 757-796.

Heim, S. Hahn, K., Auer, D.P., Fahrmeir, L. (2004). Proceedings of the 19*th* IWSM, Florence, Italy.

Papadakis, N.G., Xing, D. Huang, CL-H, Hall, L., and Carpenter, T.A. (1999). A comparative study of acquisition schemes for diffusion tensor imaging using MRI. *J Magn Reson*, **137**, 67-82.