

This is the preprint for the original manuscript published in *Statistics in Medicine*
<https://doi.org/10.1002/sim.8680>.

Partially linear monotone methods with automatic variable selection and monotonicity direction discovery

Solveig Engebretsen*, Ingrid K. Glad

February 24, 2021

¹*SAMBA, Norwegian Computing Center, Oslo, Norway*

²*Department of Mathematics, University of Oslo, Oslo, Norway*

*Corresponding author. email: solveig.engebretsen@nr.no

Abstract

In many statistical regression and prediction problems, it is reasonable to assume monotone relationships between certain predictor variables and the outcome. Genomic effects on phenotypes are for instance often assumed to be monotone. However, in some settings, it may be reasonable to assume a partially linear model, where some of the covariates can be assumed to have a linear effect. One example is a prediction model using both high-dimensional gene expression data, and low-dimensional clinical data, or when combining continuous and categorical covariates. We study methods for fitting the partially linear monotone model, where some covariates are assumed to have a linear effect on the response, and some are assumed to have a monotone (potentially nonlinear) effect. Most existing methods in the literature for fitting such models are subject to the limitation that they have to be provided the monotonicity directions a priori for the different monotone effects. We here present methods for fitting partially linear monotone models which perform both automatic variable selection, and monotonicity direction discovery. The proposed methods perform comparably to, or better than, existing methods for estimating the partially linear monotone model, in terms of estimation, prediction, and variable selection performance, in simulations experiments in both classical and high-dimensional data settings.

Keywords: Monotone regression, Partially linear, Partially linear monotone model, Monotone splines, Shape constrained regression.

1 Introduction

It is often reasonable to assume a monotone relationship between a predictor variable and the response, especially in medicine and biology. For instance, genomic effects on phenotypes are often assumed to be monotone [1, 2]. Dose-effect relationships are also often assumed to be monotone [3]. Multivariate monotone regression methods have recently been of great research interest, and some examples of methods developed are [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18], see also [19] for an overview of such methods. Sometimes, it is reasonable to assume monotone effects for some covariates, and linear effects for some covariates, that is, a partially linear monotone model. Though the methods are able to estimate linear effects also when they are specified as monotone, it is advantageous to provide the method with all the prior information available, as a less flexible method in general exhibits less variance. Frequentist methods that can be used to fit partially linear monotone models are developed in [4, 5, 6, 7], and we will denote them by the cgam method, mboost method, scam method, and scar method, respectively. Mboost [5], scam [6], and scar [7] rely on a priori information about the monotonicity directions of the nonlinear effects, while cgam [4] can estimate the monotonicity directions. In this paper, we extend the monotone splines lasso method developed in [9] to the partially linear monotone setting. The proposed new methods have the advantage that they perform automatic monotonicity direction discovery, and can be applied in the high-dimensional data setting, as well as in standard, classical regression situations.

Let d_1 be the number of covariates with linear effect on the response, d_2 be the number of covariates with nonlinear effect on the response, and n be the number of observations. The partially linear model is given by

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \sum_{k=1}^{d_2} g_k(\mathbf{Z}^{(k)}) + \boldsymbol{\varepsilon}, \quad (1.1)$$

where \mathbf{Y} is the response variable, \mathbf{X} denotes the $n \times d_1$ design matrix for the covariates assumed to potentially have a linear effect on the response, \mathbf{Z} denotes the $n \times d_2$ design matrix for the covariates assumed to potentially have a nonlinear effect on the response, $\mathbf{Z}^{(k)}$ is the k th column of \mathbf{Z} , and g_1, g_2, \dots, g_{d_2} are unknown smooth functions, assumed to have mean zero for unique identification of the functions. In the partially linear monotone model, the functions g are assumed to be monotone. The nonlinear functions in a partially linear model are commonly

estimated by kernel regression, spline approximation, piecewise polynomials, or local polynomials [20]. All the methods considered in this study rely on spline approximations. More specifically, *scar* is based on step functions, *scam* relies on P-splines, *mboost* combines P-splines with boosting, and *cgam* and the methods proposed in this paper rely on I-splines.

In the low-dimensional setting, there are several methods which can fit partially linear models without monotonicity constraints, and an overview can be found in for instance [20]. Methods for fitting partially linear models in the high-dimensional setting have also been developed, see for instance [21, 22, 23, 24, 25]. However, we will focus on the partially linear *monotone* model in this paper.

In recent years, exploiting gene expression data for prediction of clinical outcomes has become immensely popular, see for instance [26, 27, 28, 29, 30]. This requires methods which can handle high-dimensional data, where the number of parameters exceeds the number of observations. Predictive modelling with use of genomic data has however been subject to some criticism [31, 32], and it has been suggested that including clinical covariates can improve the predictive performance [33, 34, 35, 36, 37, 38]. The partially linear monotone model can be used for this purpose. We know from the literature that genomic effects are typically modelled as general monotone functions, while the clinical variables can be modelled as linear functions. The latter is particularly relevant when the clinical covariates are categorical. In the high-dimensional data setting, *mboost* is the only method existing for fitting partially linear monotone models. The application of *mboost* to the high-dimensional setting was mentioned, but not carried out, in the original proposing paper [5].

Another situation calling for the partially linear monotone model is when there are some covariates for which one is particularly interested in inspecting the relationship with the outcome, and some variables which are less of interest, but that one wishes to adjust for. Then the variables of greatest interest could enter the monotone part of the model, while the remaining variables to adjust for could enter the linear part. This could typically be the case with genomic data and clinical covariates above.

As mentioned, most of the existing methods in the literature for fitting partially linear monotone models assume correct information about the monotonicity direction for each covariate. However, such information is not always available. In particular, in a high-dimensional data setting, intuition about monotonicity directions is difficult—imagine fixing the monotonicity direction for 20 000 different genes. It can also be difficult to fix monotonicity directions in the classical setting, in particular if there is strong collinearity between the variables. Both unmeasured and measured confounders can affect the directions of the effects. In particular, it might be that the direction of the effect is of key interest to the study, for instance when testing the efficacy of a certain treatment. Our proposed methods and

cgam are the only ones, to our knowledge, which can estimate the partially linear monotone model without specifying the monotonicity directions. Moreover, in a high-dimensional data setting, the methods presented in this paper are the only methods which can estimate the monotonicity directions.

We propose two methods for fitting a partially linear monotone model, and extend them with an adaptive step. The methods build upon monotone splines lasso [9]. Monotone splines lasso is a penalised regression method, with a specific type of penalty term favouring monotone solutions. The first method proposed, which we have named PLAMM-1, has a common penalty parameter for the linear and the nonlinear monotone effects. The second method, named PLAMM-2, has two separate penalty parameters. The idea behind using two separate penalty parameters is to avoid the possibility that the clinical data, which typically contain more information than genomic data, are not included in the final model, due to the high-dimensionality of the genomic data, as discussed in [33, 39, 40]. These two methods can be used in the high-dimensional setting as well as in the classical setting. In the method with two penalty parameters, estimation has to be performed iteratively.

The structure of the paper is as follows. We start by introducing the proposed new methods, PLAMM-1 and PLAMM-2, in sections 2 and 3, respectively. We then compare the performance of the methods to the existing methods in the literature, by assessing the estimation, selection, and prediction performances in simulation experiments in both low- and high-dimensional settings, in section 4. In section 5, the methods are applied to two medical data sets. We illustrate the methods on a low-dimensional prostate cancer data set, with categorical variables included in the linear part of the model, and continuous variables modelled as monotone. We also apply the methods to a high-dimensional genomic dataset with bone mineral density measurements as the response variable, assuming that the clinical covariates have linear effects, and that the gene expressions have monotone effects. We end with discussion and concluding remarks in section 6.

2 PLAMM-1

PLAMM-1 is a straightforward extension of monotone splines lasso [9] to the partially linear setting. PLAMM-1 stands for partially linear additive monotone method 1. The PLAMM-1 framework was also briefly sketched in [19], however, the method was not explored in practice. The idea is to approximate the monotone effects by I-splines, which are monotone spline basis functions, introduced in [18].

The I-spline approximations are given by

$$\tilde{g}_j(x) = \sum_{k=1}^m \gamma_{jk} I_k^{(l)}(x), \quad (2.1)$$

where \tilde{g}_j is a spline approximation of g_j , $I_k^{(l)}$ is an I-spline basis function of order l , with corresponding coefficients γ_{jk} , $k = 1, \dots, m$, and m is the number of I-splines used in the approximation. The number of basis splines is related to the number of knots, K , as $m = K + l$. In this paper, we will for simplicity use I-spline basis functions of order two and knots based on uniform quantiles. As in [9], the I-spline basis functions are centred, to ensure unique identification of the functions.

The I-splines are monotonically increasing, and hence the spline approximation for g_j is monotone as long as all the basis coefficients are either nonnegative or nonpositive. A cooperative lasso penalty [41] is used to favour sign-coherence for the coefficients.

Let \mathbf{Z}' be the design matrix with the nonlinear explanatory variables, evaluated in the I-spline basis. The dimension of \mathbf{Z}' is thus $n \cdot md_2$, since there are d_2 such variables, and m spline basis functions per covariate. Further, let $\mathbf{X}' = (\mathbf{X}, \mathbf{Z}')$ be the full design matrix, with the linear variables in \mathbf{X} , and the nonlinear monotone variables represented in the I-spline basis. The columns of \mathbf{X}' are thus grouped, such that the first d_1 columns consist of singleton groups, while the latter md_2 columns are grouped into groups of size m , one per covariate. We let these groups be denoted by \mathcal{G}_j , $j = 1, \dots, d_1, d_1 + 1, \dots, d_1 + d_2$. Let $\boldsymbol{\phi} = (\boldsymbol{\beta}, \boldsymbol{\gamma})$ be the corresponding vector of coefficients. We further use the notation $\boldsymbol{\phi}_{\mathcal{G}_j} = \boldsymbol{\beta}_j$ for $j = 1, \dots, d_1$ and $\boldsymbol{\phi}_{\mathcal{G}_j} = (\gamma_{j1}, \dots, \gamma_{jm})$ for $j = d_1 + 1, \dots, d_1 + d_2$. The estimated parameters for PLAMM-1 are then given by

$$\hat{\boldsymbol{\phi}} = \operatorname{argmin}_{\boldsymbol{\phi}} \|\mathbf{y} - \mathbf{X}'\boldsymbol{\phi}\|_2^2 + \lambda \|\boldsymbol{\phi}\|_{\text{coop}}. \quad (2.2)$$

The parameter λ is a tuning parameter controlling the regularisation, and can be chosen by cross-validation. The cooperative penalty term is given by

$$\|\boldsymbol{\phi}\|_{\text{coop}} = \sum_{j=1}^{d_1+d_2} w_j \left(\|\boldsymbol{\phi}_{\mathcal{G}_j}^+\|_2 + \|\boldsymbol{\phi}_{\mathcal{G}_j}^-\|_2 \right), \quad (2.3)$$

where $\boldsymbol{\phi}_{\mathcal{G}_j}^+ = \max(\boldsymbol{\phi}_{\mathcal{G}_j}, 0)$ and $\boldsymbol{\phi}_{\mathcal{G}_j}^- = \max(-\boldsymbol{\phi}_{\mathcal{G}_j}, 0)$. Note that the penalty term corresponds to an L_1 penalty for the linear covariates. As usual, the covariates are assumed to be standardised, to avoid scaling dependency in the penalty terms.

Since the group sizes are not equal, we use weights w_j on the penalty terms. The standard group lasso weights are square root of group size. However, the spline basis functions for each covariate are not independent, and hence the standard group lasso weights are not the most appropriate for this setting [42]. We therefore let the weights be chosen by a (nested) cross-validation procedure. Hence, $w_j = 1$ for the linear covariates, and $w_j = m^\alpha$ for the nonlinear, monotone covariates, where α is chosen by cross-validation.

Since the cooperative lasso penalty has the variable selection property, PLAMM-1 performs automatic variable selection in both the linear and nonlinear covariates. The resulting estimated functions are not guaranteed to be monotone, but non-monotone solutions are penalised more. While this can be seen as a limitation of the method, as monotone solutions are sought, it makes the method more robust to violation of the monotonicity assumption. If nonmonotone solutions are not accepted, the penalty parameter can be increased until a monotone solution is obtained.

Like monotone splines lasso and lasso, PLAMM-1 can be extended by an adaptive procedure. The adaptive version will in the following be denoted by APLAMM-1. Let $\hat{\boldsymbol{\phi}}^{\text{init}}$ be the initial fit for the linear coefficients and the basis coefficients. Then let

$$m_j = \begin{cases} \infty, & \text{if } \|\hat{\boldsymbol{\phi}}_{\mathcal{G}_j}^{\text{init}}\|_2 = 0, \\ 1/\|\hat{\boldsymbol{\phi}}_{\mathcal{G}_j}^{\text{init}}\|_2, & \text{otherwise.} \end{cases} \quad (2.4)$$

The APLAMM-1 solution is then given by

$$\hat{\boldsymbol{\phi}} = \operatorname{argmin}_{\boldsymbol{\phi}} \|\mathbf{y} - \mathbf{X}'\boldsymbol{\phi}\|_2^2 + \lambda \sum_{j=1}^{d_1+d_2} m_j w_j \left(\|\boldsymbol{\phi}_{\mathcal{G}_j}^+\|_2 + \|\boldsymbol{\phi}_{\mathcal{G}_j}^-\|_2 \right), \quad (2.5)$$

where, for simplicity, we let w_j be the same group size weights as for the initial PLAMM-1 fit.

By assuming that the true, underlying g -functions can be represented exactly as a linear combination of m I-spline basis functions, it follows directly from the properties of cooperative lasso (and the monotone splines lasso) that PLAMM-1 is asymptotically unbiased, and has the property of exact support recovery, under certain conditions provided in [41].

3 PLAMM-2

Though the parameter α of PLAMM-1 allows for different penalties on the linear and nonlinear terms, an alternative and possibly more flexible method is obtained with a separate penalty for the linear and the nonlinear terms. We call this method PLAMM-2 (partially linear additive monotone method 2). For example in the setting of including clinical covariates in the linear part, and genomic variables in the nonlinear part, the genomic variables are typically of much higher dimension than the clinical variables. By using two separate penalty parameters, the clinical and high-dimensional covariates are handled in different ways, taking care of the risk for the clinical predictors to disappear due to the vast amount of genomic features, as discussed in [33, 39, 40].

In order to perform variable selection, we use an L_1 penalty on the linear coefficients, and a cooperative lasso penalty on the basis coefficients. As will become clear, the method is quite general, and could be used with other (or no) penalties on the linear coefficients. The PLAMM-2 solution is given by

$$(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) = \operatorname{argmin}_{(\boldsymbol{\beta}, \boldsymbol{\gamma})} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}'\boldsymbol{\gamma}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \lambda_2 \|\boldsymbol{\gamma}\|_{\text{coop}}. \quad (3.1)$$

The solution is found by an iterative scheme, similar to the approach of [22]. We start with an initial guess for $\boldsymbol{\beta}$ (e.g. zero), and denote it by $\hat{\boldsymbol{\beta}}^{(0)}$. We use $\hat{\boldsymbol{\beta}}^{(k)}$ to find $\hat{\boldsymbol{\gamma}}^{(k+1)}$ by

$$\hat{\boldsymbol{\gamma}}^{(k+1)} = \operatorname{argmin}_{\boldsymbol{\gamma}} \|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}^{(k)} - \mathbf{Z}'\boldsymbol{\gamma}\|_2^2 + \lambda_2 \|\boldsymbol{\gamma}\|_{\text{coop}}. \quad (3.2)$$

Similarly, $\hat{\boldsymbol{\gamma}}^{(k+1)}$ is used to find $\hat{\boldsymbol{\beta}}^{(k+1)}$, by

$$\hat{\boldsymbol{\beta}}^{(k+1)} = \operatorname{argmin}_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}'\hat{\boldsymbol{\gamma}}^{(k+1)}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1. \quad (3.3)$$

The updating is repeated until convergence is reached in both $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\gamma}}$, in terms of change in the squared distance between the current and the previous estimate. The optimal penalty parameters λ_1 and λ_2 are estimated for each iteration, by use of cross-validation. We show convergence of the iterative algorithm for fixed penalty parameters in the online supplementary information.

PLAMM-2 can easily be extended with an adaptive step, in the same manner as e.g. PLAMM-1. We denote the adaptive PLAMM-2 method by APLAMM-2. Let $\hat{\boldsymbol{\beta}}^{\text{init}}$ and $\hat{\boldsymbol{\gamma}}^{\text{init}}$ be the initial PLAMM-2 parameter estimates. Then let

$$m_j = \begin{cases} \infty, & \text{if } |\hat{\beta}_j^{\text{init}}| = 0, \\ 1/|\hat{\beta}_j^{\text{init}}|, & \text{otherwise,} \end{cases} \quad (3.4)$$

for $j = 1, \dots, d_1$, and

$$m_j = \begin{cases} \infty, & \text{if } \|\hat{\boldsymbol{\gamma}}_{\mathcal{G}_j}^{\text{init}}\|_2 = 0, \\ 1/\|\hat{\boldsymbol{\gamma}}_{\mathcal{G}_j}^{\text{init}}\|_2, & \text{otherwise,} \end{cases} \quad (3.5)$$

for $j = d_1 + 1, \dots, d_1 + d_2$, where $\boldsymbol{\gamma}_{\mathcal{G}_j} = (\gamma_{j1}, \dots, \gamma_{jm})$.

The APLAMM-2 solution is given by

$$(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) = \operatorname{argmin}_{(\boldsymbol{\beta}, \boldsymbol{\gamma})} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}'\boldsymbol{\gamma}\|_2^2 + \lambda_1 \sum_{j=1}^{d_1} m_j |\beta_j| + \lambda_2 \sum_{j=d_1+1}^{d_1+d_2} m_j (\|\boldsymbol{\gamma}_{\mathcal{G}_j}^+\|_2 + \|\boldsymbol{\gamma}_{\mathcal{G}_j}^-\|_2) \quad (3.6)$$

This is solved by an iterative scheme, similar to the nonadaptive method.

4 Simulation experiments

To assess the estimation, prediction, and variable selection performance of the proposed methods, we perform simulation experiments in different settings. The simulation set-up is similar to the set-up in [43, 9, 19]. We will study the performance in both low-dimensional and high-dimensional settings, with both dependent and independent covariates, varying the amount of noise. We are interested in the variable selection performance of the methods, and we thus also include some noise covariates in all but one setting. Though variable selection may be most important in high-dimensional data settings, parsimonious models can also be attractive in the classical setting, as they are easier to interpret [44, 45].

The true model is the same in all simulation settings, given by

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \beta_4 x_{i4} + g_1(z_{i1}) + g_2(z_{i2}) + g_3(z_{i3}) + g_4(z_{i4}) + \varepsilon_i \quad (4.1)$$

where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, and σ is chosen to control the signal to noise-ratio (SNR), that is, the ratio between the standard deviation of the signal and σ .

We let the true linear coefficients be $(\beta_1, \beta_2, \beta_3, \beta_4) = (2, 2, -2, -2)$. The monotone functions are the same as in [9], and are given as

$$\begin{aligned} g_1(x) &= -\exp(x^2), & g_2(x) &= -\log(x + 0.1), & (4.2) \\ g_3(x) &= 2 \tanh(20x^2) + 0.5 \exp(x^3), & g_4(x) &= \frac{2 \exp(10x - 5)}{1 + \exp(10x - 5)}. \end{aligned}$$

The g -functions are centred.

We let

$$\begin{aligned} x_{ij} &= \frac{w_{ij} + tu_i}{1+t}, & z_{ij} &= \frac{a_{ij} + tu_i}{1+t} \text{ for } j = 1, \dots, 4, & (4.3) \\ x_{ij} &= \frac{w_{ij} + tv_i}{1+t}, & z_{ij} &= \frac{a_{ij} + tv_i}{1+t} \text{ for } j = 5, \dots, d_2, \end{aligned}$$

where u_i, v_i, w_{ij} , and a_{ij} are drawn from a truncated normal distribution with mean 0.5 and standard deviation 1. The variable t controls the correlation between the covariates, and they are independent for $t = 0$. We will also work with $t = 1$, corresponding to a within-set correlation of 0.5, in the set of signal covariates and that of noise covariates. We vary the amount of noise covariates such that $d_1 = 4, 10$ or 100 , and $d_2 = 4, 10, 100$ or 500 . We consider $n = 50$ and $n = 80$, and an SNR of 2 or 4.

We compute the mean squared error for the estimated functions for the different methods, by the distance between the true function and the estimated function in the observation points. We also compute the prediction error. This is

done by generating 500 new observations from the true model, predict their values based on the estimated models with the different methods, and measure the mean squared distance between the true values and the predicted values. In addition, the methods are compared through the estimation of the linear coefficients, the number of true positives, and the number of false positives, where applicable. We simulate 100 times and draw different random variables in each simulation. We report averages over the 100 simulations.

4.1 Compared methods and parameter choices

The new methods PLAMM-1, PLAMM-2, APLAMM-1, and APLAMM-2 are compared to the appropriate existing methods for the different settings. In the low-dimensional setting without any noise covariates, the methods are compared to *cgam*, *scam*, *scar*, and *mboost*, using the R-packages *cgam*, *scam*, *scar*, and *mboost*. *Cgam* has an option of variable selection discovery for each covariate, and we include this option, in order to compare the variable selection performance. In the low-dimensional settings with noise covariates, the methods are compared to *cgam*, *scam*, and *mboost*. *Scar* cannot be used in this setting, as it requires the user to specify monotonicity directions for all the nonlinear covariates, and we do not know which directions to put on the noise covariates. For *scam* and *mboost*, this is solved by not imposing monotonicity restrictions on the noise covariates. We let *cgam* select the monotonicity direction for each nonlinear covariate. Finally, in the high-dimensional setting, the methods are compared to *mboost*. Both our proposed methods and *mboost* perform automatic variable selection.

The tuning parameters of PLAMM-1, APLAMM-1, PLAMM-2, and APLAMM-2 are chosen by 10-fold cross-validation. The group penalty parameter α is chosen by a nested cross-validation procedure using a coarse grid-search for values in the range from 0.5 to 2.0, with increment 0.25. Applying the different methods requires different choices, and we refer to [4, 46] for details on *cgam*, [6] for details on *scam*, [5] for details on *mboost*, and [7] for details on *scar*. We use the default number of knots for *cgam*, of the order $n^{1/7}$, and equal spacing (default option). The penalty parameter of *scam* is chosen by the default GCV-option. For *scam*, it is in general recommended to use a large number of knots, and let the penalty smooth the fitted functions to avoid overfitting. However, we easily run into the problem of having too many parameters compared to observations, hence we use 15 interior knots in the setting with no noise covariates, 6 knots in the settings with $n = 80$, and 4 interior knots in the setting with $n = 50$. For *mboost*, 10-fold cross-validation is used as a stopping criterion for the boosting iterations, and a maximum number of 20 interior knots is used. *Scar* has the advantage that there are no tuning parameters or other method choices that have to be specified. *Scam*, *mboost*, and *scar* are provided with the (correct) monotonicity directions for the

true covariates.

4.2 Simulations in the low-dimensional data setting

4.2.1 No noise covariates

In the setting with no noise covariates, we also use independent covariates ($t = 0$), $\text{SNR} = 4$, $n = 80$, and $d_1 = d_2 = 4$. The results are provided in Table 1. In this simple setting, all the methods seem to perform well. For the estimation errors for the monotone functions and the linear coefficients, scam and cgam have the smallest estimation errors, however the errors for our four proposed methods and mboost are only slightly higher. Regarding prediction performance, scam and cgam again perform the best, while our proposed methods and mboost have slightly higher prediction errors. None of the methods which perform variable selection had any problems selecting all the variables in this setting. Scar was clearly inferior to the other methods here.

4.2.2 Low-dimensional settings with noise covariates

We now consider different low-dimensional settings where $d_1 = d_2 = 10$, hence there are six noise covariates in each group. The results with a strong signal ($\text{SNR} = 4$), independent covariates and $n = 80$ are provided in Table 2. As expected, all the methods perform worse when we include noise covariates. Our methods have slightly lower estimation errors for the monotone functions than the other methods, and the two adaptive versions perform the best. Scam and cgam perform the best in estimating the linear coefficients, but APLAMM-1 and APLAMM-2 also perform well. Regarding prediction, APLAMM-1 performs the best among all methods, and all the four proposed methods perform better than scam, mboost, and cgam in this setting, but they only have slightly lower prediction errors. All the methods select all the true variables. APLAMM-1 selects the fewest false nonlinear covariates (i.e. noise covariates that are modelled as nonlinear), and APLAMM-2 selects the fewest false linear covariates (i.e. noise covariates that are modelled as linear). Mboost selects almost all the false nonlinear covariates.

The estimated functions for g_3 and g_4 are provided in s 1 and 2 for the different methods. The monotone functions seem to be well-captured by the different methods. It is difficult to visually judge the relative performances.

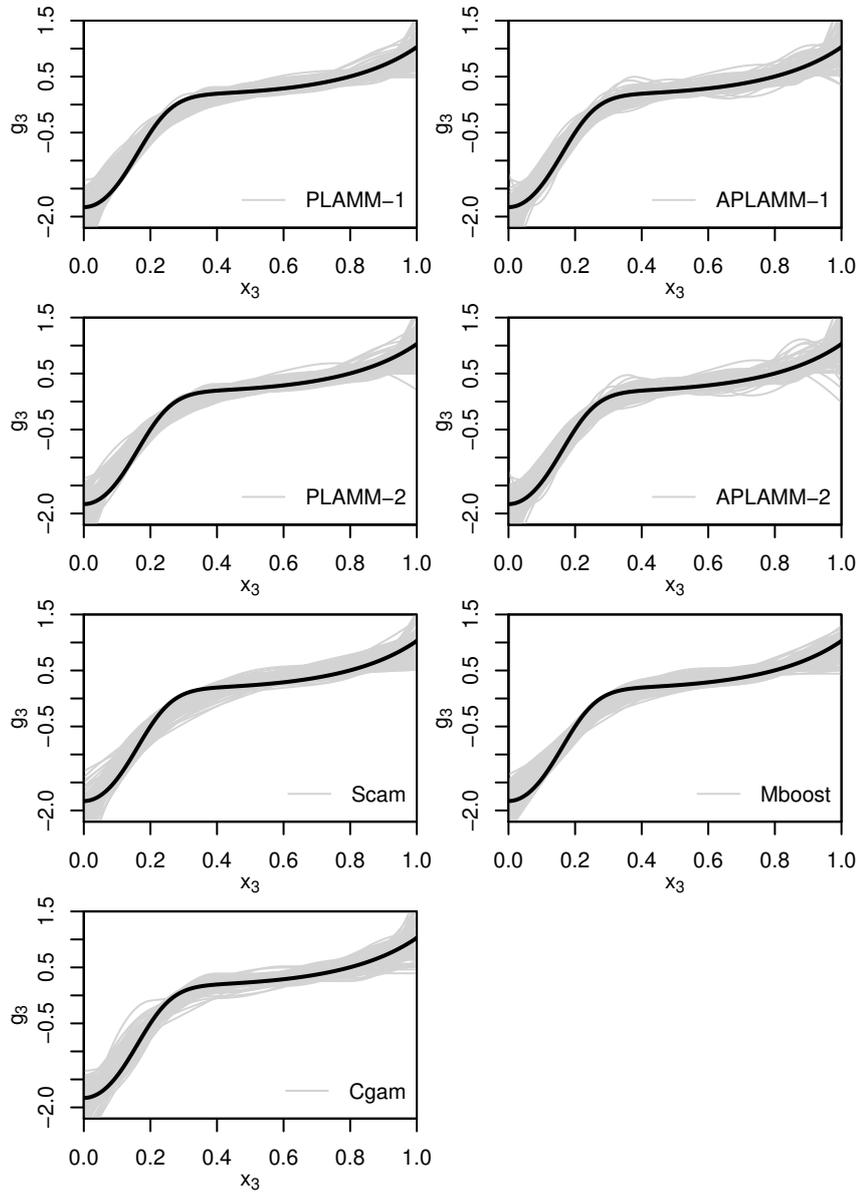


Figure 1: Estimated functions for g_3 with $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 0$, for all the different methods. The true function is given in black.

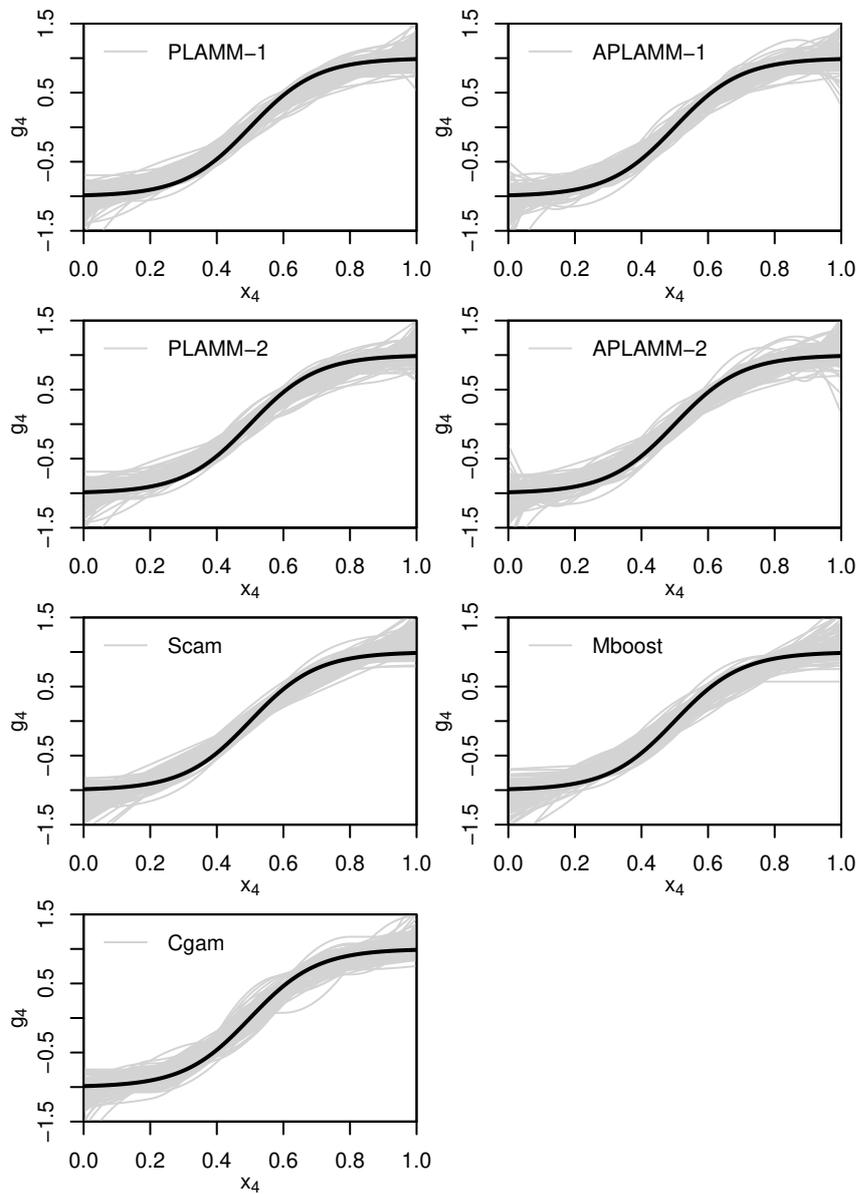


Figure 2: Estimated functions for g_4 with $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 0$, for all the different methods. The true function is given in black.

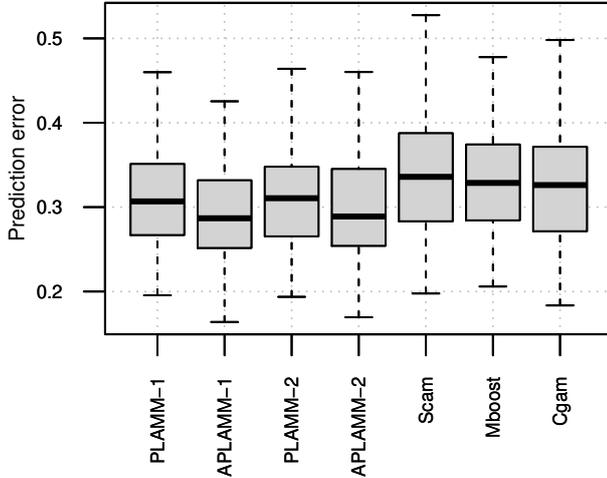


Figure 3: Prediction errors for the different methods, in the setting with $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 0$.

A box plot of the prediction error in this setting is given in Figure 3. Again, we can see that the methods perform similarly. The four proposed methods have slightly lower prediction errors than the other methods. Scam seems to be slightly less stable (i.e. higher variance) than the other methods. A box plot with mean squared errors for g_1 and g_2 , and the estimated β_1 and β_3 for the different methods, is provided in Figure 4. The methods perform similarly also in terms of estimation error. Mboost seems to have larger estimation errors for the linear coefficients than the other methods, while scam and cgam have the smallest estimation errors for the linear coefficients.

Fewer observations We consider the more difficult problem of estimating the model with only $n = 50$ observations, with $\text{SNR} = 4$ and $t = 0$. The results are provided in Table 3. The estimation errors for the monotone functions are smallest for PLAMM-1 and APLAMM-1. As before, scam and cgam perform the best in estimating the linear coefficients, followed by APLAMM-1. APLAMM-1 and PLAMM-1 perform the best in terms of prediction, and all the suggested methods outperform scam. Mboost and cgam perform similarly to PLAMM-2 and APLAMM-2. In terms of selection, APLAMM-1 performs the best, while mboost again selects almost all the false nonlinear covariates.

Dependent covariates We consider dependent covariates, that is, $t = 1$. The results for the setting with $n = 80$, $\text{SNR} = 4$ and $t = 1$ are provided in Table 4.

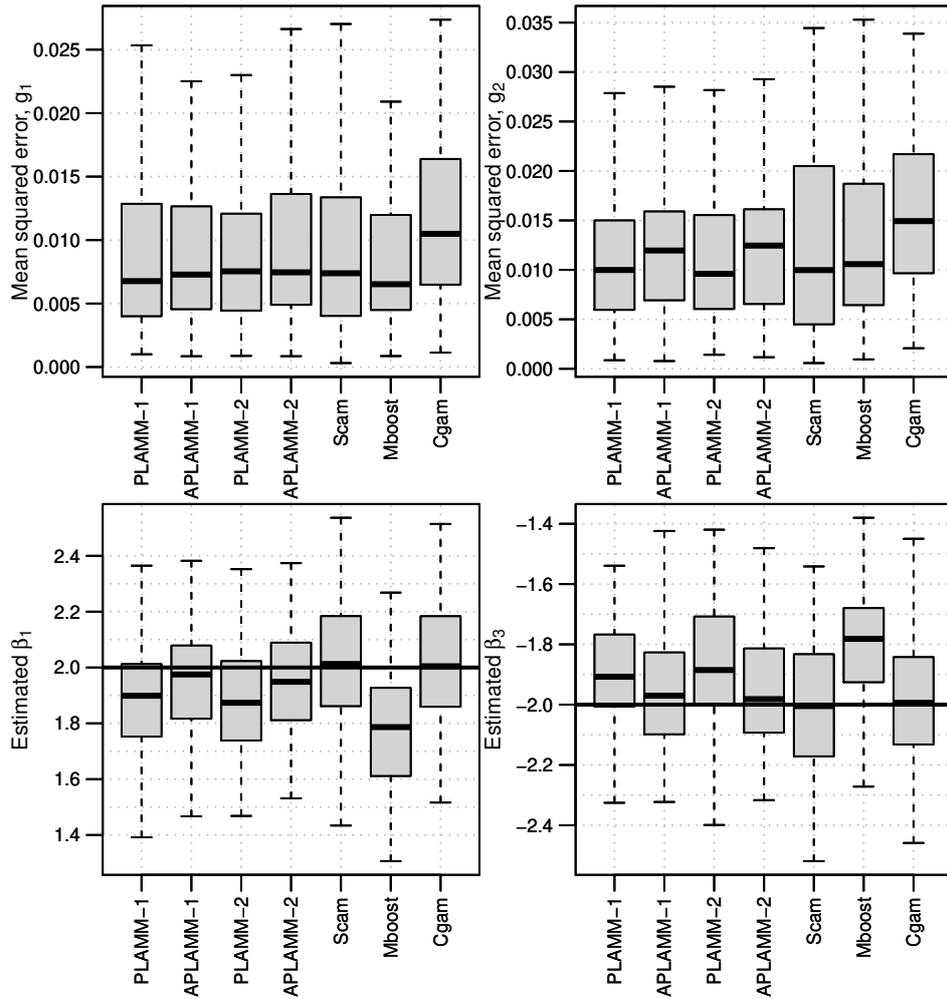


Figure 4: Mean squared errors for the estimated g_1 and g_2 , and the estimated β_1 and β_3 for the different methods, in the setting with $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 0$.

APLMM-1 obtains the smallest estimation errors for the monotone functions in this setting. Cgam, mboost, and scam outperform PLMM-2 and APLMM-2 in estimation in this setting. For the linear coefficients, the estimates with scam and cgam are closest to the true values, followed closely by the APLMM-1 estimates. APLMM-1 has the smallest prediction errors, and APLMM-2 has the largest prediction errors. APLMM-1 selects all the true covariates, while APLMM-2 selects fewer true linear covariates in this setting. Mboost selects almost all the false nonlinear covariates. APLMM-1 selects the fewest false nonlinear covariates, and APLMM-2 selects the fewest false linear covariates.

Weak signal We explore the situation with more noise, that is, $\text{SNR} = 2$. The results for the setting with $n = 80$, $\text{SNR} = 2$ and $t = 0$ are provided in Table 5. The estimation errors for the nonlinear functions are similar for all the methods in the setting with a weaker signal, and they are larger than for the other low-dimensional settings considered. The lowest estimation errors for the nonlinear functions were obtained with mboost. Cgam and scam again perform the best in terms of estimating the linear coefficients, followed by APLMM-2. None of the methods perform particularly well in terms of prediction, but our proposed methods perform similarly, and better than cgam, scam, and mboost. All the methods are good at selecting the true covariates. APLMM-2 selects the fewest false covariates, while mboost again selects almost all the false nonlinear covariates.

4.3 High-dimensional settings

We consider the performance of our proposed methods in three high-dimensional data settings. The first one is $n = 50$, $d_1 = 100$, $d_2 = 100$, $\text{SNR} = 4$ and $t = 0$. In the second setting, we assume more noise covariates in the nonlinear part, as this is typically the setting when combining clinical data with molecular data, like for instance gene expressions. Hence, for the second setting, all the variables are the same as in the first high-dimensional setting, except d_2 , which is increased to $d_2 = 500$. The third example is the same as the second, but with more information, increasing the number of observations to $n = 100$.

The results in the setting with $d_1 = d_2 = 100$ are given in Table 6. In terms of estimation for both the monotone functions and the linear coefficients, APLMM-1 is clearly better than the other methods, and mboost is clearly inferior. APLMM-1 is also the best method in terms of prediction and selection. Mboost has the largest prediction errors. Though APLMM-1 selects fewer true covariates than

PLAMM-1, the number of false positives is clearly reduced in the adaptive step. The nonlinear covariates are clearly favoured for both PLAMM-2, APLAMM-2, and mboost.

The results in the setting with $d_1 = 100$ and $d_2 = 500$ are given in Table 7. The methods perform worse when including more noise covariates. In terms of selection, all the methods select too many false covariates in this setting. APLAMM-1 performs the best in terms of estimation of the linear coefficients, and APLAMM-2 obtains the smallest estimation errors for the monotone functions. Mboost again has the largest prediction errors, while PLAMM-1 obtains the lowest prediction error in this setting.

By increasing the number of observations to 100, the methods perform a lot better, see Table 8. All the methods perform well in selecting the true covariates in this setting. APLAMM-1 performs the best in terms of selection and estimation. APLAMM-1 also outperforms all the other methods in terms of prediction, while APLAMM-2 has the largest prediction errors in this setting, and mboost performs the second worst.

5 Medical data examples

We illustrate the methods on a low-dimensional data set on prostate cancer patients and on a high-dimensional bone mineral density data set. For technical details on modelling choices and specifications, see section 5.3.

5.1 Low-dimensional data example: prostate cancer

We illustrate the methods on a low-dimensional data set on prostate cancer patients, previously analysed for instance in [47] and with lasso in [48], and originally published in [49]. The setting is quite simple, with a relatively large number of observations compared to the number of covariates. The data contain measurements of a prostate specific antigen, which is our response variable, for 97 men who are having a radical prostatectomy. The available explanatory variables are cancer volume, prostate weight, age, benign prostatic hyperplasia amount, capsular penetration, indicator of seminal vesicle invasion (svi), the Gleason score, and

the percentage Gleason scores 4 or 5 (p_{gg45}). The Gleason score is a categorical variable. We categorise it into two groups, those with Gleason scores of 6, and those with scores 7 or higher, as only a few patients had Gleason scores above 7. We let the categorical variables svi and Gleason score enter the linear part of the model, and the continuous variables enter the monotone nonlinear part of the model.

5.1.1 Results

The estimated effects of svi and Gleason score are provided in Table 9. The estimates vary considerably for the different methods. For scam, the direction of the effect of svi is opposite to that of the other methods, and hence likely incorrect, in particular since it is reasonable to assume a positive effect of svi. For mboost, the effect of Gleason is a lot smaller than for the other methods. Comparing with the lasso analysis in [48], the Gleason score was not selected, while svi had a positive estimated effect.

The estimated effects for the continuous predictors are provided in Figure 5. Cancer volume and prostate weight are selected by all the methods, while age and benign prostatic hyperplasia are only selected by cgam, scam, and mboost, and p_{gg45} is only selected by scam and mboost. None of the methods estimated an effect of capsular penetration. The estimated effect of cancer volume seems to be largest for high and low cancer volume. Scam and cgam seem to be more sensitive to influential observations than the other methods. Out of these continuous variables, the lasso variable selection [48] also ended up selecting only cancer volume and prostate weight, in agreement with the new proposed methods. The directions of the effects with lasso are also in agreement with the monotone methods.

5.1.2 Prediction performance

To assess the prediction performance of the methods on this data set, we randomly split the data into a training and a test set, consisting of two-thirds and one-third of the observations, respectively. We fit the model to the training set, and estimate its prediction performance on the test set, in terms of mean squared prediction errors. In order to avoid sensitivity to the particular training and test set split, we repeat the procedure 50 times, and report the average prediction error. The monotonicity direction estimation for mboost and scam is performed for every training set split, and all the tuning parameters for the different methods are re-estimated.

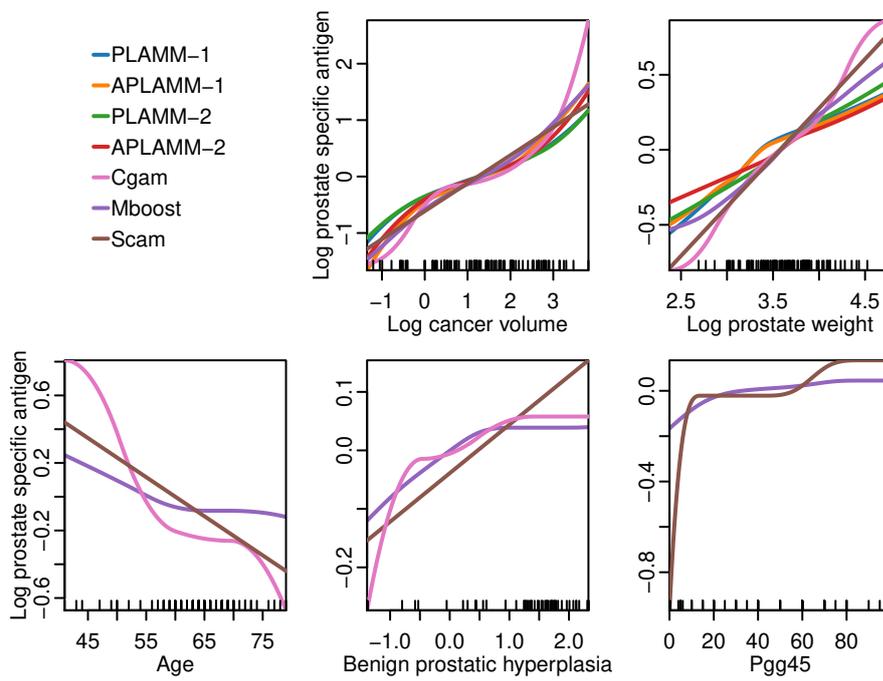


Figure 5: Estimated effects of the different explanatory variables selected by the different methods, on log prostate specific antigen. The estimated α for PLAMM-1 was 1.5.

The prediction performances are also provided in Table 9. In general, the methods seem to perform similarly to each other, but cgam obtained a larger prediction error than the other methods. PLAMM-2 obtained the lowest prediction error, and mboost performed second best.

5.2 High-dimensional data example: bone mineral density

We illustrate the methods using a bone mineral density data set from [50], consisting of 84 menopausal women who have had a transiliacal bone biopsy. The data consist of bone mineral density measurements, which is our response variable, gene expressions, age, body mass index (BMI), parathyroid hormone values (pth, measured in pmol/L), vitamin D values (measured in nmol/L), and carboxy-terminal telopeptide of type 1 collagen (ctp, measured in $\mu\text{g/L}$), measured in blood samples. For simplicity and computational efficiency, we only use the 1000 genes with the largest variance in the analysis, based on the implicit assumption that the response will vary the most with the genes that vary the most. The data also contain other clinical measurements, but we have chosen these variables, as there is only one missing observation, while the rest of the data have multiple missing values. We remove the one observation with missing data for these variables, and work with 83 out of the 84 observations. We used DAVID bioinformatics resources 6.8 [51] to get the gene names from the affymetrix IDs. For illustration, we also compare with a linear lasso model. We let the clinical covariates enter the linear part of the model, and the gene expression covariates enter the monotone part of the model.

5.2.1 Results

The estimated parameters for the clinical variables are given in Table 10. BMI is the only covariate which is selected by all the monotone methods, and it has a positive estimated effect, as expected. Age and pth are selected by all the monotone methods except mboost, which only selects one linear covariate. The estimated linear effects for PLAMM-1 and APLAMM-1 are much larger than the estimated coefficients for the other methods. APLAMM-1 selected the fewest linear covariates among the new proposed methods. Lasso selected age, BMI, and pth, while adaptive lasso, as mboost, only selected BMI. The estimated coefficients with lasso are smaller than for the other methods.

All the methods estimated a sparse model, with relatively few genes selected, out of the 1000 possible genes. PLAMM-1 selected the most genes, 15. PLAMM-

2 selected 11, mboost selected 8, APLAMM-2 selected 7, and APLAMM-1 selected 4. The selected genes and their effects are provided in Figures 6 and 7. The total number of different genes selected by the methods is 20. Considering that there were 1000 possible different genes, and that the gene expressions are correlated, the set of selected genes were similar for the different methods. Eight genes were only selected by one method. All the methods selected the genes SOST and AFFX-M27830-M-AT (uncharacterised), and these genes were also found to be important in [50]. COL3A1 is selected by all methods except mboost, and PLGLB2 is selected by all methods except APLAMM-1. The effect of PLGLB2 is clearly nonlinear, and seems to have a threshold-type effect. Lasso selected 35 genes, and adaptive lasso selected 23, including most of the genes selected by the different monotone methods.

5.2.2 Prediction performance

We also assess the prediction performance of the methods on the bone mineral density data set in the same way as before, by randomly splitting the data in a test and a training set, repeated 50 times, containing one-third and two-thirds of the observations, respectively. The monotonicity direction estimation for mboost is performed for every training set, and all the tuning parameters for the different methods are re-estimated. The results are provided in Table 10. PLAMM-1 obtained the smallest prediction errors, and PLAMM-2 performed second best. The two adaptive methods performed the worst among the methods.

5.3 Modelling choices and specifications

For PLAMM-1, PLAMM-2, APLAMM-1, and APLAMM-2, we used 10-fold cross-validation to estimate the penalty parameters, as in the simulations. The penalty parameter α of PLAMM-2 was estimated by the same coarse grid search as in the simulations, ranging between 0.5 and 2.0. We let cgam discover which covariates should be included in the final model, and the monotonicity directions for the nonlinear variables. For mboost and scam, we had to provide the monotonicity directions for all the nonlinear variables. We did this by fitting an initial ridge regression model with linear effects, and letting the monotonicity directions correspond to the signs of the estimated linear coefficients. The default number of equally spaced knots was used for cgam. 10-fold cross-validation was used as a stopping criterion for the boosting iterations for mboost, and a maximum number of 20 interior knots was used. For scam, 10 interior knots were used, and the penalty parameters were chosen by the default GCV option.

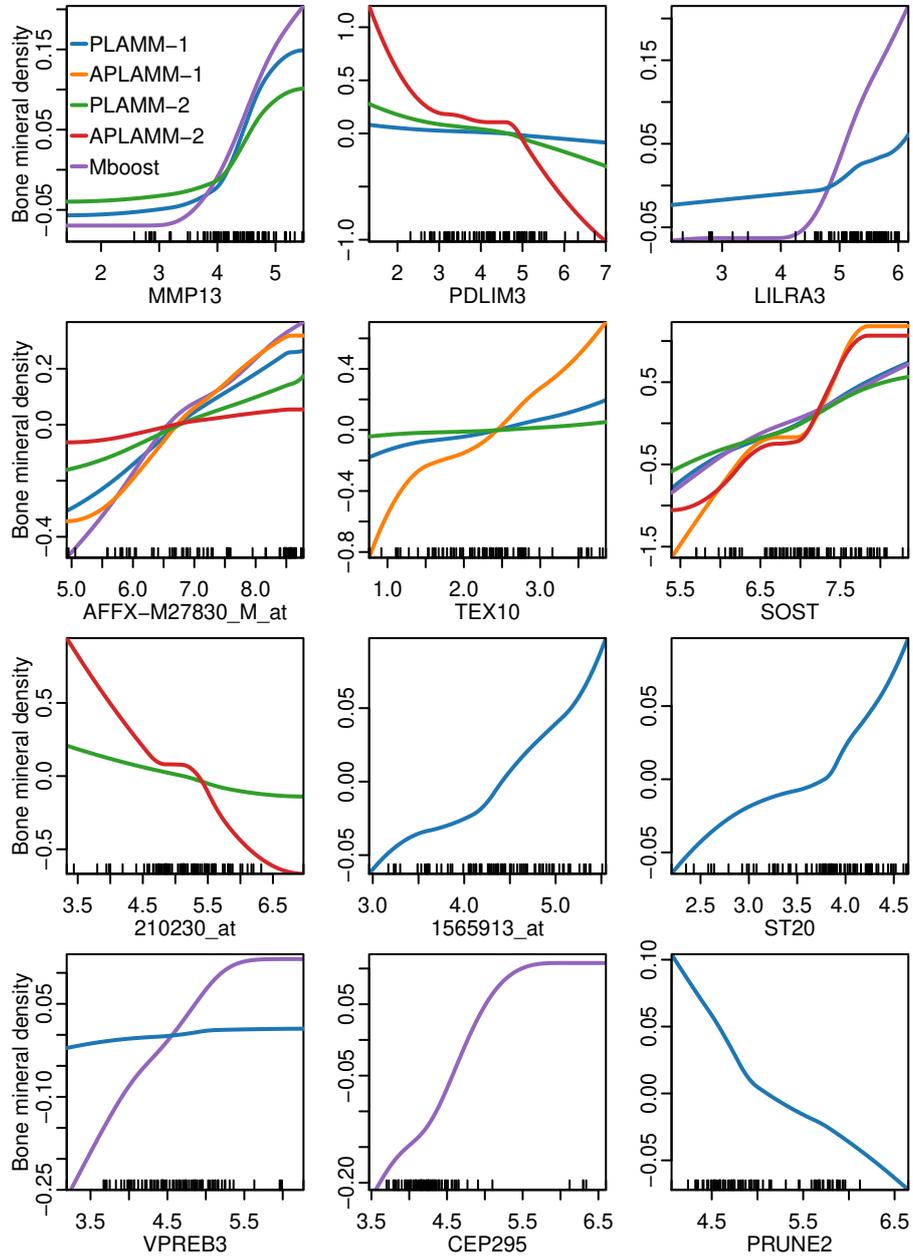


Figure 6: Estimated effects on bone mineral density of the different genes selected by the different methods. The estimated α for PLAMM-1 was 0.75.

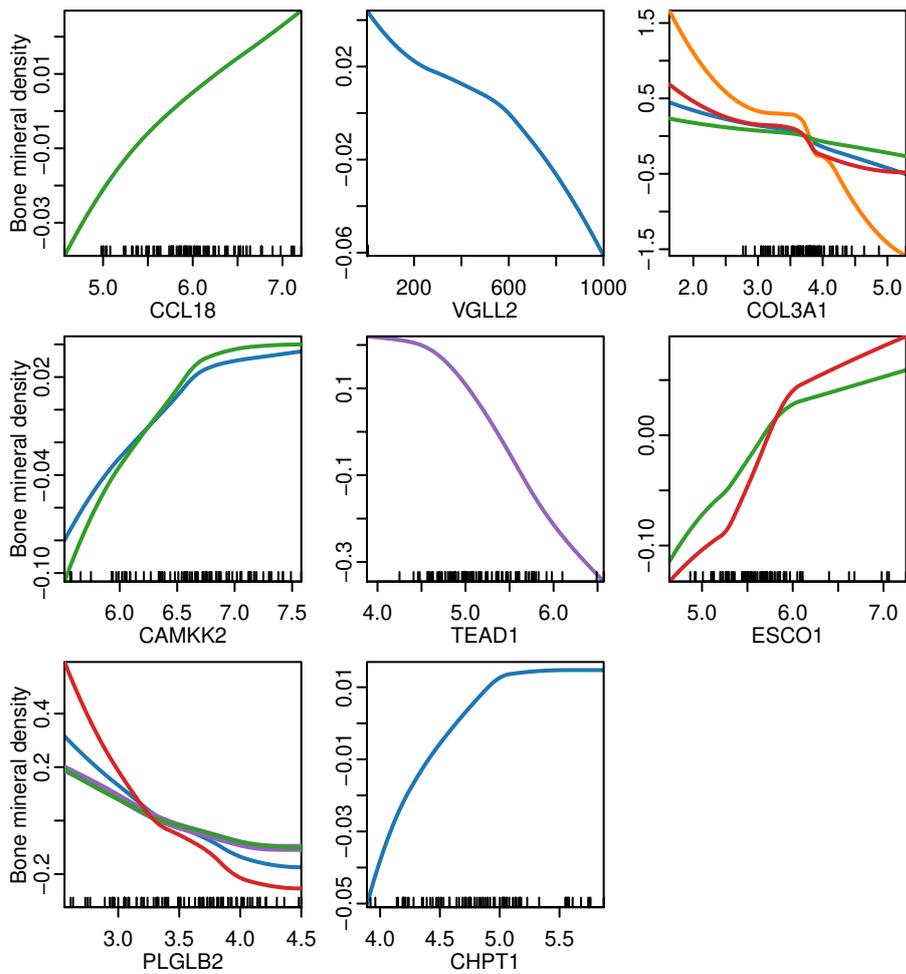


Figure 7: Estimated effects on bone mineral density of the different genes selected by the different methods. The estimated α for PLAMM-1 was 0.75.

6 Discussion

We have proposed four estimation methods for partially linear monotone models, all based on monotone splines lasso, combining I-splines with a cooperative lasso penalty. We named the methods PLAMM-1, PLAMM-2, APLAMM-1, and APLAMM-2.

PLAMM-1 has one common penalty parameter for the linear and the nonlinear variables, but weighs the penalty with group size raised to the power α . The parameter α is thus a tuning parameter which controls the relative penalty on the monotone and linear effects. APLAMM-1 is the adaptive version of PLAMM-1. PLAMM-2 instead uses two different penalty parameters on the linear and the monotone covariates, and the estimation is performed iteratively. APLAMM-2 is the adaptive version of PLAMM-2. PLAMM-2 is a more general method than PLAMM-1, as it can be used with any type of penalty on the linear covariates, while PLAMM-1 poses a cooperative penalty on the linear covariates, corresponding to an L_1 penalty for single covariates. Hence, for grouped linear covariates, PLAMM-1 favours sign-coherence, through use of the cooperative lasso penalty. PLAMM-2 can use any penalty desired, for instance group lasso. If the grouped linear covariates are ordered categorical variables, then sign-coherence may be an attractive property. This is also one of the original suggested application areas for the cooperative lasso penalty, discussed in [41].

Our proposed methods perform automatic monotonicity direction discovery. We often do not have prior information about the monotonicity directions of the effects, especially in settings with strong collinearity or confounding variables. There are no obvious ways in which to estimate monotonicity directions for methods which do not incorporate this. That is also why we, in our simulation experiments, fitted the noise covariates without monotonicity restrictions for scam and mboost. In our medical data example applications, we chose to let the monotonicity directions be the directions from an initial ridge regression fit, assuming a linear model, for the methods which cannot estimate the monotonicity directions. This might not be the optimal solution, and the success of this approach is likely to depend on how much the true, underlying relationships deviate from linear functions. Moreover, estimating the monotonicity directions and the model in two separate steps complicates the understanding. For example, in the case of mboost, we do not know whether a covariate was left out of the model because it had no underlying relationship with the response, or whether it was because the estimated monotonicity direction was incorrect. Hence, automatic monotonicity direction discovery is a desirable and sometimes necessary property.

In a high-dimensional data setting, intuition about monotonicity directions is even less likely to be available. Hence, the automatic monotonicity direction property of our methods is particularly attractive in the high-dimensional settings. We

find that our methods outperform mboost in selection, estimation, and prediction, in simulation experiments in high-dimensional data settings, despite the fact that mboost is provided with correct information about the monotonicity directions of the effects. APLAMM-1 performs the overall best in the high-dimensional data setting.

The methods are also compared to existing methods for estimating partially linear monotone models in the classical setting. In a setting with no noise covariates, the methods are compared to cgam, scam, scar, and mboost. In this simple nonsparse setting, scam and cgam perform the best among the methods, however our methods perform almost as well. We performed multiple simulations in sparse settings with noise covariates. APLAMM-1 performed the overall best in these settings in terms of prediction, selection, and estimation of the monotone effects. Scam and cgam performed the best in terms of estimating the linear parameters. Cgam was in general good at selecting the true covariates, but selected too many false. Mboost tended to select many false covariates in the nonlinear part of the model. Mboost, scam, and cgam are more general than our proposed methods, as they allow other constraints than linear and monotonicity constraints. Hence, if the underlying model assumption includes covariates which are assumed to be nonmonotone, cgam, mboost or scam should be used instead.

Though we conclude that APLAMM-1 performs the overall best among the methods, it was not obvious in all settings whether or not there was a benefit in the adaptive step for both PLAMM-1 and PLAMM-2. For instance, the prediction errors did not always decrease with the adaptive step. In particular, in the medical data examples, the adaptive methods obtained larger prediction errors than their nonadaptive counterparts. In terms of estimation, the adaptive step in general improved the estimation for the linear coefficients, while the estimation errors for the monotone functions did not improve with the adaptive step. In most settings, the adaptive step improved the selection performance. However, there is a risk of sacrificing some true positives in exchange for a (much greater) reduction in false positives. It is not obvious how to determine the cost/loss of the trade-off between true and false positives.

We have chosen to estimate the parameter α for PLAMM-1, and then use the same α for APLAMM-1, in order to use as much of the information in the data as possible to estimate the model. This is also the reason why we have chosen to use a coarse grid search, instead of a finer grid. It would have been possible to also estimate α separately for APLAMM-1 and/or use a finer grid, however it is not obvious whether that would improve or worsen the performance.

Another potential tuning parameter is the number of knots for the different methods. We have chosen to fix the number of knots. In general, the more knots, the greater the flexibility of the estimated function, but the less information to estimate each spline coefficient [18]. For I-splines, [18] argues that a great flexibility

can be obtained with few knots, and that it is more important to estimate each basis spline well. For PLAMM-1, PLAMM-2, APLAMM-1, and APLAMM-2, we have fixed the number of interior knots to six, and the cooperative lasso penalty is used to avoid overfitting. For cgam, we have used the default number of knots, which is $n^{1/7}$. For scam, it is recommended to use a relatively large number of knots, to avoid underfitting. We have therefore used a relatively large number of knots for scam, compared to the number of observations. Overfitting is handled by a smoothing penalty term. Mboost has an inbuilt automatic selection of the number of knots. We could have estimated the number of knots for all the methods, by minimising an estimate of the prediction error, for instance the cross-validation error on the training set, as in [52]. However, again, it is not obvious whether using more information in the data in the tuning procedure would lead to a better or worse performance of the different methods. Moreover, [53] claims that when there are shape constraints on the functions (e.g. monotonicity), then the estimated functions are robust to the number of knots.

We illustrated the use of the methods on a low-dimensional prostate cancer data set. We included categorical variables in the linear part of the model, and the continuous variables as monotone. The setting was quite simple, with a relatively large number of observations, compared to the number of predictors. Mboost and cgam selected more covariates than PLAMM-1 and PLAMM-2, and hence a denser model. In terms of prediction errors, the methods performed similarly in this rather simple real data setting, but cgam obtained the largest prediction errors. The methods were also tried out on a high-dimensional genomic data set, where bone mineral density was modelled using a partially linear monotone model. Clinical covariates were included in the linear part of the model, and gene expression variables were assumed to have monotone effects. Mboost selected only one linear covariate (BMI), which might be due to the high-dimensionality of the genomic variables. This was also the case for the adaptive lasso. The problem of informative clinical covariates getting lost due to the high dimensionality of genomic data was also discussed in [33, 39, 40]. An alternative way of combining clinical and genomic data, is to first fit the model using only the clinical data, then use the genomic data to fit the residuals, as considered in [33, 39, 54, 40]. Comparing with such an approach would have been an interesting option, in particular for mboost, which only selected one linear covariate. PLAMM-1 and PLAMM-2 do not seem to have this problem, which is likely due to different penalties on the linear and the nonlinear covariates. This is more similar to the so-called "favoring strategy" considered in e.g. [33, 40], which was also used in [55], where our methods can be seen as data-driven favouring strategies. Different penalties on the clinical and genomic variables were also used in [36].

We have proven theoretical convergence of the iterative algorithm for the PLAMM-2 method. The convergence criterion used in practice was that the

squared distance between the estimated coefficients in two consecutive iterations was smaller than a prespecified tolerance. We used a tolerance of 10^{-4} for the simulation studies, and 10^{-3} for the medical data examples. However, in the simulation experiments (but not in the medical data examples), we stopped the algorithm after a maximum number (40) of iterations. In most of the settings considered, PLAMM-2 converged in more than 90% of the simulations, before reaching the maximum number of iterations. Our experience is that it normally took less than ten iterations before the method converged. The convergence was most difficult in the high-dimensional settings considered, which may partly be due to the tolerance criterion, which is more difficult to reach the higher the dimension, and due to the high-dimensional data setting being more challenging. We chose to include the results based on all the simulations, regardless of whether the method converged or not. This was done because the method is still an approximation to the solution even if it had not converged, and in order to avoid cherry picking settings that PLAMM-2 handles well.

We compared our methods to existing frequentist methods which can fit partially linear monotone models. However, it should be noted that there exist also Bayesian methods which are able to handle partially linear monotone models, see for instance [11]. However, we have chosen to only compare our proposed methods to frequentist methods, as this seems most fair.

All the methods we have considered assume a priori knowledge of which covariates should be modelled as linear, and which should be included in the non-linear part. However, such knowledge is not always available, in particular in the high-dimensional data setting, as noted in [56]. There exist also methods which can separate the covariates into linear and nonlinear effects, see [57] for a method which can be used in the low-dimensional setting, and [56] and [58] for methods which can be used in the high-dimensional setting. A topic for future work is thus to extend the ideas of PLAMM-1 and PLAMM-2 to a method for separation into linear and monotone covariates, which utilises the monotonicity property of the functions.

In this paper, we have proposed two different methods for estimating a partially linear monotone model, and extended them with an adaptive step. The main advantage of our methods over existing methods in the literature, is that they perform automatic monotonicity direction discovery, while most of the previously proposed methods have to be provided with (correct) information about the monotonicity directions. In particular, our methods are the first which can perform automatic monotonicity direction discovery in a high-dimensional data setting. Out of the four proposed methods, APLAMM-1 performed the best, in terms of prediction, estimation, and selection. APLAMM-1 outperformed the existing methods in all but the simplest dense setting with no noise covariates. Hence, we recommend using APLAMM-1 in all but the simplest settings where

most of the variables are expected to be of relevance.

Implementations

The implementations of PLAMM-1 and PLAMM-2 are based on the implementation of monotone splines lasso, found at <http://www.mn.uio.no/math/english/people/aca/glad/r-scripts/mlasso/>. It is based on the R-package *scoop* for cooperative lasso [41], found at <http://julien.cremeriefamily.info/scoop>. Our implementations of PLAMM-1 and PLAMM-2 can also be found at <https://www.mn.uio.no/math/english/people/aca/glad/r-scripts/mlasso/>.

Acknowledgement

The authors are grateful to Sjur Reppe for providing access to the bone mineral density data.

Funding

The authors acknowledge partial funding from the Norwegian Research Council centre Big Insight project 237718.

Data availability

The prostate cancer data are published in [49], and can also be found for instance in the R-package *ElemStatLearn* [59]. The bone mineral density data are available from the European Bioinformatics Institute (EMBL-EBI) ArrayExpress repository, ID: E-MEXP-1618 [60].

Supporting information

Additional supporting information may be found online in the Supporting Information section as part of the online article.

References

- [1] Q. Han, T. Wang, S. Chatterjee, and R. J. Samworth, “Isotonic regression in general dimensions,” *The Annals of Statistics*, vol. 47, no. 5, pp. 2440–2471, 2019.
- [2] R. Luss, S. Rosset, and M. Shahar, “Efficient regularized isotonic regression with application to gene–gene interaction search,” *The Annals of Applied Statistics*, vol. 6, no. 1, pp. 253–283, 2012.
- [3] C.-I. C. Lee, “On estimation for monotone dose-response curves,” *Journal of the American Statistical Association*, vol. 91, no. 435, pp. 1110–1119, 1996.
- [4] M. C. Meyer, “A framework for estimation and inference in generalized additive models with shape and order restrictions,” *Statistical Science*, vol. 33, no. 4, pp. 595–614, 2018.
- [5] B. Hofner, T. Kneib, and T. Hothorn, “A unified framework of constrained regression,” *Statistics and Computing*, vol. 26, no. 1-2, pp. 1–14, 2016.
- [6] N. Pya and S. N. Wood, “Shape constrained additive models,” *Statistics and Computing*, vol. 25, no. 3, pp. 543–559, 2015.
- [7] Y. Chen and R. J. Samworth, “Generalized additive and index models with shape constraints,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 78, no. 4, pp. 729 – 754, 2016.
- [8] L. Wang and L. Xue, “Constrained polynomial spline estimation of monotone additive models,” *Journal of Statistical Planning and Inference*, vol. 167, pp. 27–40, 2015.
- [9] L. C. Bergersen, K. Tharmaratnam, and I. K. Glad, “Monotone splines lasso,” *Computational Statistics & Data Analysis*, vol. 77, pp. 336–351, 2014.
- [10] Z. Fang and N. Meinshausen, “Lasso isotone for high-dimensional additive isotonic regression,” *Journal of Computational and Graphical Statistics*, vol. 21, no. 1, pp. 72–91, 2012.
- [11] M. C. Meyer, A. J. Hackstadt, and J. A. Hoeting, “Bayesian estimation and inference for generalised partial linear models using shape-restricted splines,” *Journal of Nonparametric Statistics*, vol. 23, no. 4, pp. 867–884, 2011.

- [12] O. Saarela and E. Arjas, “A method for Bayesian monotonic multiple regression,” *Scandinavian Journal of Statistics*, vol. 38, no. 3, pp. 499–513, 2011.
- [13] B. Bornkamp, K. Ickstadt, and D. Dunson, “Stochastically ordered multiple regression,” *Biostatistics*, vol. 11, no. 3, pp. 419–431, 2010.
- [14] G. Tutz and F. Leitenstorfer, “Generalized smooth monotonic regression in additive modeling,” *Journal of Computational and Graphical Statistics*, vol. 16, no. 1, pp. 165–188, 2007.
- [15] H. Dette and R. Scheder, “Strictly monotone and smooth nonparametric regression for two or more variables,” *Canadian Journal of Statistics*, vol. 34, no. 4, pp. 535–561, 2006.
- [16] F. Leitenstorfer and G. Tutz, “Generalized monotonic regression based on B-splines with an application to air pollution data,” *Biostatistics*, vol. 8, no. 3, pp. 654–673, 2006.
- [17] P. Bacchetti, “Additive isotonic models,” *Journal of the American Statistical Association*, vol. 84, no. 405, pp. 289–294, 1989.
- [18] J. O. Ramsay, “Monotone regression splines in action,” *Statistical Science*, vol. 3, no. 4, pp. 425–441, 1988.
- [19] S. Engebretsen and I. K. Glad, “Additive monotone regression in high and lower dimensions,” *Statistics Surveys*, vol. 13, pp. 1–51, 2019.
- [20] W. Härdle and H. Liang, “Partially linear models,” in *Statistical methods for biostatistics and related fields*, pp. 87–103, Springer, 2007.
- [21] X. Liu, L. Wang, and H. Liang, “Estimation and variable selection for semi-parametric additive partial linear models,” *Statistica Sinica*, vol. 21, no. 3, pp. 1225–1248, 2011.
- [22] P. Du, G. Cheng, and H. Liang, “Semiparametric regression models with additive nonparametric components and high dimensional parametric components,” *Computational Statistics & Data Analysis*, vol. 56, no. 6, pp. 2006–2017, 2012.
- [23] F. Wei, “Group selection in high-dimensional partially linear additive models,” *Brazilian Journal of Probability and Statistics*, vol. 26, pp. 219–243, Aug 2012.

- [24] J. Guo, M. Tang, M. Tian, and K. Zhu, “Variable selection in high-dimensional partially linear additive models for composite quantile regression,” *Computational Statistics & Data Analysis*, vol. 65, pp. 56–67, 2013.
- [25] J. Lv, H. Yang, and C. Guo, “Variable selection in partially linear additive models for modal regression,” *Communications in Statistics-Simulation and Computation*, vol. 46, no. 7, pp. 5646–5665, 2017.
- [26] S. L. Pomeroy, P. Tamayo, M. Gaasenbeek, L. M. Sturla, M. Angelo, M. E. McLaughlin, J. Y. Kim, L. C. Goumnerova, P. M. Black, C. Lau, J. C. Allen, D. Zzag, J. M. Olson, T. Curran, C. Wetmore, J. A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D. N. Louis, J. P. Mesirov, E. S. Lander, and T. R. Golub, “Prediction of central nervous system embryonal tumour outcome based on gene expression,” *Nature*, vol. 415, no. 6870, pp. 436–442, 2002.
- [27] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, “Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks,” *Nature medicine*, vol. 7, no. 6, pp. 673–679, 2001.
- [28] H. Salem, G. Attiya, and N. El-Fishawy, “Classification of human cancer diseases by gene expression profiles,” *Applied Soft Computing*, vol. 50, pp. 124–134, 2017.
- [29] X. Lin, Y. Zhao, W.-m. Song, and B. Zhang, “Molecular classification and prediction in gastric cancer,” *Computational and structural biotechnology journal*, vol. 13, pp. 448–458, 2015.
- [30] P.-E. Colombo, F. Milanezi, B. Weigelt, and J. S. Reis-Filho, “Microarrays in the 2010s: the contribution of microarray-based gene expression profiling to breast cancer classification, prognostication and prediction,” *Breast Cancer Research*, vol. 13, no. 3, p. 212, 2011.
- [31] S. Michiels, S. Koscielny, and C. Hill, “Prediction of cancer outcome with microarrays: a multiple random validation strategy,” *The Lancet*, vol. 365, no. 9458, pp. 488–492, 2005.
- [32] K. R. Hess, C. Wei, Y. Qi, T. Iwamoto, W. F. Symmans, and L. Pusztai, “Lack of sufficiently strong informative features limits the potential of gene expression analysis as predictive tool for many clinical classification problems,” *BMC bioinformatics*, vol. 12, no. 1, p. 463, 2011.

- [33] R. De Bin, W. Sauerbrei, and A.-L. Boulesteix, “Investigating the prediction ability of survival models based on both clinical and omics data: two case studies,” *Statistics in medicine*, vol. 33, no. 30, pp. 5310–5329, 2014.
- [34] J. R. Nevins, E. S. Huang, H. Dressman, J. Pittman, A. T. Huang, and M. West, “Towards integrated clinico-genomic models for personalized medicine: combining gene expression signatures and clinical factors in breast cancer outcomes prediction,” *Human molecular genetics*, vol. 12, no. suppl_2, pp. R153–R157, 2003.
- [35] A. J. Stephenson, A. Smith, M. W. Kattan, J. Satagopan, V. E. Reuter, P. T. Scardino, and W. L. Gerald, “Integration of gene expression profiling and clinical variables to predict prostate carcinoma recurrence after radical prostatectomy,” *Cancer: Interdisciplinary International Journal of the American Cancer Society*, vol. 104, no. 2, pp. 290–298, 2005.
- [36] A. I. Vazquez, Y. Veturi, M. Behring, S. Shrestha, M. Kirst, M. F. Resende, and G. de los Campos, “Increased proportion of variance explained and prediction accuracy of survival of breast cancer patients with use of whole-genome multiomic profiles,” *Genetics*, vol. 203, no. 3, pp. 1425–1438, 2016.
- [37] T. Hothorn, P. Bühlmann, S. Dudoit, A. Molinaro, and M. J. Van Der Laan, “Survival ensembles,” *Biostatistics*, vol. 7, no. 3, pp. 355–373, 2005.
- [38] A. Obulkasim, G. A. Meijer, and M. A. van de Wiel, “Stepwise classification of cancer samples using clinical and molecular data,” *BMC bioinformatics*, vol. 12, no. 1, p. 422, 2011.
- [39] H. Binder and M. Schumacher, “Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models,” *BMC bioinformatics*, vol. 9, no. 1, p. 14, 2008.
- [40] A.-L. Boulesteix and W. Sauerbrei, “Added predictive value of high-throughput molecular data to clinical data and its validation,” *Briefings in bioinformatics*, vol. 12, no. 3, pp. 215–229, 2011.
- [41] J. Chiquet, Y. Grandvalet, and C. Charbonnier, “Sparsity with sign-coherent groups of variables via the cooperative-lasso,” *The Annals of Applied Statistics*, vol. 6, no. 2, pp. 795–830, 2012.
- [42] N. Simon and R. Tibshirani, “Standardization and the group lasso penalty,” *Statistica Sinica*, vol. 22, no. 3, p. 983, 2012.

- [43] Y. Lin and H. H. Zhang, “Component selection and smoothing in multivariate nonparametric regression,” *The Annals of Statistics*, vol. 34, no. 5, pp. 2272–2297, 2006.
- [44] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society: series B (statistical methodology)*, vol. 67, no. 2, pp. 301–320, 2005.
- [45] T. Hesterberg, N. H. Choi, L. Meier, and C. Fraley, “Least angle and L_1 penalized regression: A review,” *Statistics Surveys*, vol. 2, pp. 61–93, 2008.
- [46] X. Liao and M. C. Meyer, “cgam: An r package for the constrained generalized additive model,” *Journal of Statistical Software*, vol. 89, no. 5, 2018.
- [47] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics, Springer, 2009.
- [48] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
- [49] T. A. Stamey, J. N. Kabalin, M. Ferrari, and N. Yang, “Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. IV. anti-androgen treated patients,” *The Journal of urology*, vol. 141, no. 5, pp. 1088–1090, 1989.
- [50] S. Reppe, H. Refvem, V. T. Gautvik, O. K. Olstad, P. I. Høvring, F. P. Reinholt, M. Holden, A. Frigessi, R. Jemtland, and K. M. Gautvik, “Eight genes are highly associated with BMD variation in postmenopausal Caucasian women,” *Bone*, vol. 46, no. 3, pp. 604–612, 2010.
- [51] D. W. Huang, B. T. Sherman, and R. A. Lempicki, “Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources,” *Nature protocols*, vol. 4, no. 1, pp. 44–57, 2009.
- [52] D. Ruppert, “Selecting the number of knots for penalized splines,” *Journal of Computational and Graphical Statistics*, vol. 11, no. 4, pp. 735–757, 2002.
- [53] M. C. Meyer, “Inference using shape-restricted regression splines,” *The Annals of Applied Statistics*, vol. 2, pp. 1013–1033, Sep 2008.

- [54] A. Volkmann, R. De Bin, W. Sauerbrei, and A.-L. Boulesteix, “A plea for taking all available clinical information into account when assessing the predictive value of omics data,” *BMC medical research methodology*, vol. 19, no. 1, p. 162, 2019.
- [55] H. M. Bøvelstad, S. Nygård, and Ø. Borgan, “Survival prediction from clinico-genomic models-a comparative study,” *BMC bioinformatics*, vol. 10, no. 1, p. 413, 2009.
- [56] H. Lian, H. Liang, and D. Ruppert, “Separation of covariates into non-parametric and parametric parts in high-dimensional partially linear additive models,” *Statistica Sinica*, vol. 25, pp. 591–607, 2015.
- [57] H. H. Zhang, G. Cheng, and Y. Liu, “Linear or nonlinear? Automatic structure discovery for partially linear models,” *Journal of the American Statistical Association*, vol. 106, no. 495, pp. 1099–1112, 2011.
- [58] Y. Lou, J. Bien, R. Caruana, and J. Gehrke, “Sparse partially linear additive models,” *Journal of Computational and Graphical Statistics*, vol. 25, no. 4, pp. 1126–1140, 2016.
- [59] K. Halvorsen, “ElemStatLearn: Data sets, functions and examples from the book: The Elements of Statistical Learning, Data Mining, Inference, and Prediction by Trevor Hastie, Robert Tibshirani and Jerome Friedman,” *R package version*, vol. 26, 2015.
- [60] ArrayExpress, EMBL-EBI, “E-MEXP-1618 - Transcription profiling of bone biopsies from postmenopausal females identifies 8 genes highly associated with bone mineral density.” <https://www.ebi.ac.uk/arrayexpress/experiments/E-MEXP-1618/>.

Table 1: True positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear parameters, for $n = 80$, $d_1 = d_2 = 4$, $\text{SNR} \approx 4$ and $t = 0$. PLAMM-1 selected an average $\alpha = 1.32$ (0.48). Standard deviations are given in parentheses. "TPM" denotes true positives among the monotone covariates, and "TPL" denotes true positives among the linear covariates.

Selection:	TPM	TPL	PE	
PLAMM-1	4.0 (0)	4.0 (0)	0.27 (0.055)	
PLAMM-2	4.0 (0)	4.0 (0)	0.28 (0.068)	
APLAMM-1	4.0 (0)	4.0 (0)	0.28 (0.055)	
APLAMM-2	4.0 (0)	4.0 (0)	0.28 (0.068)	
Cgam	4.0 (0)	4.0 (0)	0.25 (0.068)	

Results provided correct monotonicity directions:				
Scam	–	–	0.25 (0.041)	
Scar	–	–	0.42 (0.24)	
Mboost	4.0 (0)	4.0 (0)	0.26 (0.048)	

MSE:	g_1	g_2	g_3	g_4
PLAMM-1	0.0099 (0.0074)	0.012 (0.0078)	0.015 (0.0094)	0.013 (0.0086)
PLAMM-2	0.010 (0.0074)	0.012 (0.0081)	0.015 (0.0093)	0.012 (0.0093)
APLAMM-1	0.011 (0.0090)	0.013 (0.0091)	0.015 (0.0092)	0.014 (0.0091)
APLAMM-2	0.011 (0.0096)	0.013 (0.0098)	0.015 (0.0097)	0.014 (0.0096)
Cgam	0.011 (0.0075)	0.013 (0.0087)	0.013 (0.0080)	0.0099 (0.0078)

Results provided correct monotonicity directions:				
Scam	0.011 (0.0091)	0.0099 (0.0086)	0.013 (0.0086)	0.013 (0.0096)
Scar	0.027 (0.014)	0.033 (0.016)	0.032 (0.014)	0.031 (0.015)
Mboost	0.0085 (0.0065)	0.011 (0.011)	0.017 (0.010)	0.013 (0.011)

	β_1	β_2	β_3	β_4
PLAMM-1	1.96 (0.20)	1.98 (0.20)	–1.95 (0.18)	–1.98 (0.20)
PLAMM-2	1.95 (0.21)	1.96 (0.21)	–1.94 (0.22)	–1.97 (0.23)
APLAMM-1	1.97 (0.20)	1.99 (0.20)	–1.96 (0.18)	–2.00 (0.20)
APLAMM-2	1.95 (0.22)	1.96 (0.22)	–1.94 (0.22)	–1.97 (0.24)
Cgam	2.00 (0.23)	2.01 (0.22)	–2.00 (0.19)	–2.02 (0.21)

Results provided correct monotonicity directions:				
Scam	2.00 (0.22)	2.02 (0.21)	–2.00 (0.19)	–2.02 (0.21)
Scar	1.94 (0.25)	1.99 (0.24)	–1.95 (0.24)	–1.96 (0.25)
Mboost	1.91 (0.24)	1.94 (0.22)	–1.93 (0.19)	–1.95 (0.21)

Table 2: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 0$. PLAMM-1 selected an average $\alpha = 1.12$ (0.44). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection:	TPM	TPL	FPM	FPL	PE
PLAMM-1	4.0 (0)	4.0 (0)	3.59 (1.69)	3.46 (1.95)	0.31 (0.068)
PLAMM-2	4.0 (0)	4.0 (0)	3.66 (1.72)	3.17 (1.88)	0.32 (0.087)
APLAMM-1	4.0 (0)	4.0 (0)	0.67 (1.05)	1.45 (1.83)	0.29 (0.064)
APLAMM-2	4.0 (0)	4.0 (0)	0.83 (1.16)	1.10 (1.48)	0.30 (0.086)
Cgam	4.0 (0)	4.0 (0)	2.71 (1.22)	1.74 (1.12)	0.33 (0.084)

Results provided correct monotonicity directions:					
Scam	–	–	–	–	0.35 (0.092)
Mboost	4.0 (0)	4.0 (0)	5.71 (0.59)	3.11 (1.56)	0.33 (0.069)

MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.010 (0.0089)	0.012 (0.0080)	0.019 (0.0097)	0.017 (0.013)	
PLAMM-2	0.010 (0.0088)	0.012 (0.0085)	0.019 (0.010)	0.017 (0.013)	
APLAMM-1	0.011 (0.0093)	0.013 (0.0079)	0.016 (0.0081)	0.015 (0.010)	
APLAMM-2	0.011 (0.0088)	0.013 (0.0088)	0.016 (0.0087)	0.015 (0.012)	
Cgam	0.012 (0.0081)	0.016 (0.0097)	0.018 (0.0099)	0.015 (0.011)	

Results provided correct monotonicity directions:					
Scam	0.011 (0.0096)	0.013 (0.010)	0.025 (0.013)	0.021 (0.015)	
Mboost	0.0092 (0.0078)	0.014 (0.0097)	0.019 (0.0092)	0.019 (0.014)	

	β_1	β_2	β_3	β_4	
PLAMM-1	1.89 (0.24)	1.91 (0.21)	–1.90 (0.20)	–1.89 (0.22)	
PLAMM-2	1.86 (0.25)	1.88 (0.24)	–1.87 (0.25)	–1.85 (0.26)	
APLAMM-1	1.96 (0.22)	1.98 (0.20)	–1.96 (0.18)	–1.95 (0.20)	
APLAMM-2	1.95 (0.25)	1.97 (0.23)	–1.95 (0.22)	–1.93 (0.27)	
Cgam	2.00 (0.25)	2.03 (0.22)	–2.00 (0.22)	–1.98 (0.23)	

Results provided correct monotonicity directions:					
Scam	2.01 (0.26)	2.03 (0.22)	–2.01 (0.22)	–1.99 (0.24)	
Mboost	1.78 (0.25)	1.80 (0.20)	–1.80 (0.21)	–1.78 (0.24)	

Table 3: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 50$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 0$. PLAMM-1 selected an average $\alpha = 1.09$ (0.41). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection:	TPM	TPL	FPM	FPL	PE
PLAMM-1	4.0 (0)	4.0 (0)	3.75 (1.54)	3.86 (1.69)	0.42 (0.11)
PLAMM-2	4.0 (0)	3.95 (0.36)	3.42 (1.58)	3.41 (2.03)	0.51 (0.39)
APLAMM-1	3.99 (0.10)	4.0 (0)	0.96 (1.23)	1.60 (1.68)	0.41 (0.15)
APLAMM-2	3.97 (0.22)	3.92 (0.44)	1.01 (1.22)	1.29 (1.57)	0.53 (0.46)
Cgam	4.0 (0)	4.0 (0)	2.99 (1.18)	1.93 (1.25)	0.51 (0.21)

Results provided correct monotonicity directions:					
Scam	–	–	–	–	0.57 (0.20)
Mboost	4.0 (0)	4.0 (0)	5.68 (0.72)	3.06 (1.65)	0.53 (0.14)

MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.021 (0.016)	0.020 (0.014)	0.034 (0.022)	0.029 (0.027)	
PLAMM-2	0.028 (0.042)	0.023 (0.017)	0.041 (0.034)	0.037 (0.036)	
APLAMM-1	0.025 (0.026)	0.023 (0.018)	0.029 (0.019)	0.028 (0.031)	
APLAMM-2	0.035 (0.059)	0.028 (0.029)	0.037 (0.039)	0.037 (0.045)	
Cgam	0.027 (0.024)	0.028 (0.021)	0.039 (0.022)	0.031 (0.044)	

Results provided correct monotonicity directions:					
Scam	0.023 (0.022)	0.024 (0.023)	0.055 (0.029)	0.046 (0.029)	
Mboost	0.027 (0.023)	0.026 (0.020)	0.035 (0.024)	0.037 (0.032)	

	β_1	β_2	β_3	β_4	
PLAMM-1	1.87 (0.30)	1.89 (0.30)	–1.84 (0.29)	–1.85 (0.26)	
PLAMM-2	1.74 (0.44)	1.74 (0.48)	–1.73 (0.42)	–1.72 (0.41)	
APLAMM-1	1.94 (0.29)	1.97 (0.29)	–1.92 (0.28)	–1.94 (0.26)	
APLAMM-2	1.78 (0.47)	1.78 (0.55)	–1.77 (0.47)	–1.76 (0.44)	
Cgam	2.05 (0.35)	2.05 (0.35)	–2.00 (0.34)	–1.98 (0.31)	

Results provided correct monotonicity directions:					
Scam	2.00 (0.36)	2.03 (0.37)	–1.98 (0.40)	–2.01 (0.31)	
Mboost	1.58 (0.40)	1.60 (0.34)	–1.63 (0.39)	–1.61 (0.36)	

Table 4: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 4$ and $t = 1$. PLAMM-1 selected an average $\alpha = 1.24$ (0.33). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection:	TPM	TPL	FPM	FPL	PE
PLAMM-1	4.0 (0)	4.0 (0)	3.70 (1.60)	3.51 (1.51)	0.094 (0.023)
PLAMM-2	4.0 (0)	3.94 (0.42)	3.62 (1.71)	2.40 (1.97)	0.12 (0.070)
APLAMM-1	4.0 (0)	4.0 (0)	0.56 (1.16)	1.09 (1.19)	0.087 (0.021)
APLAMM-2	3.99 (0.10)	3.91 (0.51)	0.76 (1.44)	0.85 (1.33)	0.13 (0.10)
Cgam	4.0 (0)	4.0 (0)	2.46 (1.18)	1.62 (1.25)	0.10 (0.027)

Results provided correct monotonicity directions:					
Scam	–	–	–	–	0.11 (0.030)
Mboost	4.0 (0)	4.0 (0)	5.94 (0.24)	2.59 (1.22)	0.11 (0.025)

MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.0048 (0.0041)	0.0053 (0.0046)	0.0079 (0.0064)	0.0075 (0.0059)	
PLAMM-2	0.0067 (0.0095)	0.0064 (0.0054)	0.0093 (0.0069)	0.010 (0.0089)	
APLAMM-1	0.0050 (0.0047)	0.0053 (0.0045)	0.0061 (0.0044)	0.0075 (0.0060)	
APLAMM-2	0.0082 (0.015)	0.0073 (0.0094)	0.0094 (0.015)	0.010 (0.014)	
Cgam	0.0056 (0.0043)	0.0071 (0.0051)	0.0065 (0.0049)	0.0064 (0.0060)	

Results provided correct monotonicity directions:					
Scam	0.0047 (0.0043)	0.0060 (0.0048)	0.0090 (0.0070)	0.0073 (0.0086)	
Mboost	0.0047 (0.0040)	0.0062 (0.0057)	0.0078 (0.0053)	0.0082 (0.0065)	

	β_1	β_2	β_3	β_4	
PLAMM-1	1.94 (0.21)	1.95 (0.20)	–1.85 (0.21)	–1.85 (0.21)	
PLAMM-2	1.70 (0.43)	1.70 (0.40)	–1.58 (0.48)	–1.59 (0.44)	
APLAMM-1	2.03 (0.21)	2.04 (0.20)	–1.92 (0.20)	–1.91 (0.19)	
APLAMM-2	1.78 (0.47)	1.78 (0.46)	–1.64 (0.55)	–1.66 (0.49)	
Cgam	2.02 (0.24)	2.06 (0.23)	–2.00 (0.21)	–1.99 (0.23)	

Results provided correct monotonicity directions:					
Scam	2.00 (0.26)	2.06 (0.21)	–2.01 (0.22)	–2.01 (0.24)	
Mboost	1.70 (0.25)	1.72 (0.22)	–1.72 (0.21)	–1.72 (0.24)	

Table 5: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 80$, $d_1 = d_2 = 10$, $\text{SNR} \approx 2$ and $t = 0$. PLAMM-1 selected an average $\alpha = 1.11$ (0.40). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection	TPM	TPL	FPM	FPL	PE
PLAMM-1	4.0 (0)	4.0 (0)	3.45 (1.75)	3.40 (1.87)	1.07 (0.22)
PLAMM-2	4.0 (0)	4.0 (0)	3.60 (1.68)	3.36 (1.90)	1.07 (0.22)
APLAMM-1	3.99 (0.10)	3.99 (0.10)	1.08 (1.24)	1.73 (1.95)	1.07 (0.25)
APLAMM-2	3.98 (0.14)	3.99 (0.10)	1.09 (1.23)	1.29 (1.37)	1.07 (0.23)
Cgam	4.0 (0)	3.99 (0.10)	2.53 (1.19)	1.66 (1.17)	1.24 (0.30)

Results provided correct monotonicity directions:					
Scam	–	–	–	–	1.23 (0.28)
Mboost	4.0 (0)	3.99 (0.10)	5.19 (1.11)	1.96 (1.46)	1.14 (0.23)

MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.027 (0.025)	0.033 (0.028)	0.046 (0.031)	0.048 (0.036)	
PLAMM-2	0.028 (0.026)	0.033 (0.028)	0.047 (0.032)	0.048 (0.038)	
APLAMM-1	0.034 (0.035)	0.039 (0.029)	0.048 (0.030)	0.045 (0.035)	
APLAMM-2	0.035 (0.035)	0.041 (0.033)	0.049 (0.032)	0.047 (0.038)	
Cgam	0.037 (0.027)	0.045 (0.033)	0.049 (0.032)	0.049 (0.034)	

Results provided correct monotonicity directions:					
Scam	0.032 (0.024)	0.034 (0.025)	0.057 (0.042)	0.043 (0.030)	
Mboost	0.026 (0.025)	0.037 (0.032)	0.042 (0.027)	0.045 (0.033)	

	β_1	β_2	β_3	β_4	
PLAMM-1	1.78 (0.45)	1.82 (0.37)	–1.79 (0.42)	–1.79 (0.42)	
PLAMM-2	1.70 (0.43)	1.70 (0.40)	–1.58 (0.48)	–1.59 (0.44)	
APLAMM-1	1.90 (0.46)	1.94 (0.39)	–1.92 (0.35)	–1.90 (0.41)	
APLAMM-2	1.94 (0.45)	1.95 (0.37)	–1.94 (0.35)	–1.90 (0.40)	
Cgam	1.98 (0.50)	2.04 (0.43)	–2.01 (0.43)	–1.97 (0.43)	

Results provided correct monotonicity directions:					
Scam	2.02 (0.51)	2.07 (0.40)	–2.02 (0.41)	–1.99 (0.44)	
Mboost	1.54 (0.49)	1.58 (0.38)	–1.56 (0.38)	–1.54 (0.43)	

Table 6: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 50$, $d_1 = d_2 = 100$, $\text{SNR} \approx 4$ and $t = 0$. PLAMM-1 selected an average $\alpha = 0.79$ (0.16). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection:	TPM	TPL	FPM	FPL	PE
PLAMM-1	3.62 (0.91)	3.73 (0.78)	12.91 (5.77)	12.81 (6.64)	1.27 (0.64)
PLAMM-2	3.67 (0.55)	3.33 (1.28)	18.94 (8.23)	9.98 (7.44)	1.63 (0.71)
APLAMM-1	3.49 (0.99)	3.67 (0.84)	4.54 (3.12)	6.06 (5.41)	1.12 (0.82)
APLAMM-2	3.42 (0.77)	2.92 (1.41)	7.61 (4.61)	1.75 (1.03)	1.75 (1.03)
Results provided correct monotonicity directions:					
Mboost	2.62 (1.26)	2.05 (1.48)	10.25 (15.28)	2.09 (4.00)	2.22 (0.62)
MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.098 (0.078)	0.10 (0.10)	0.13 (0.13)	0.13 (0.13)	
PLAMM-2	0.10 (0.080)	0.11 (0.10)	0.13 (0.11)	0.14 (0.11)	
APLAMM-1	0.082 (0.082)	0.085 (0.11)	0.097 (0.13)	0.094 (0.13)	
APLAMM-2	0.11 (0.087)	0.11 (0.11)	0.11 (0.10)	0.12 (0.11)	
Results provided correct monotonicity directions:					
Mboost	0.16 (0.071)	0.22 (0.12)	0.26 (0.17)	0.29 (0.19)	
	β_1	β_2	β_3	β_4	
PLAMM-1	1.26 (0.50)	1.16 (0.58)	-1.14 (0.59)	-1.24 (0.60)	
PLAMM-2	0.88 (0.62)	0.88 (0.65)	-0.83 (0.60)	-0.91 (0.69)	
APLAMM-1	1.56 (0.59)	1.45 (0.65)	-1.39 (0.66)	-1.54 (0.63)	
APLAMM-2	0.98 (0.75)	0.99 (0.81)	-0.93 (0.78)	-1.01 (0.82)	
Results provided correct monotonicity directions:					
Mboost	0.41 (0.48)	0.33 (0.47)	-0.37 (0.49)	-0.43 (0.57)	

Table 7: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 50$, $d_1 = 100$, $d_2 = 500$, $\text{SNR} \approx 4$ and $t = 0$. PLAMM-1 selected an average $\alpha = 0.89$ (0.23). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection:	TPM	TPL	FPM	FPL	PE
PLAMM-1	3.13 (1.24)	3.56 (0.97)	11.42 (9.83)	11.28 (9.56)	2.14 (0.72)
PLAMM-2	2.82 (1.34)	2.97 (1.31)	20.94 (12.43)	8.99 (7.78)	2.18 (0.79)
APLAMM-1	2.92 (1.31)	3.47 (1.08)	5.18 (4.60)	8.05 (6.98)	2.21 (1.02)
APLAMM-2	2.48 (1.34)	2.23 (1.40)	9.13 (6.01)	2.71 (3.42)	2.63 (1.18)
Results provided correct monotonicity directions:					
Mboost	1.07 (1.11)	0.52 (0.81)	5.30 (14.40)	0.14 (0.64)	2.96 (0.35)
MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.17 (0.076)	0.23 (0.14)	0.30 (0.17)	0.30 (0.20)	
PLAMM-2	0.16 (0.077)	0.20 (0.14)	0.25 (0.17)	0.27 (0.19)	
APLAMM-1	0.17 (0.086)	0.22 (0.16)	0.26 (0.22)	0.26 (0.22)	
APLAMM-2	0.16 (0.082)	0.19 (0.15)	0.23 (0.19)	0.25 (0.22)	
Results provided correct monotonicity directions:					
Mboost	0.21 (0.052)	0.32 (0.10)	0.43 (0.15)	0.45 (0.17)	
	β_1	β_2	β_3	β_4	
PLAMM-1	0.83 (0.75)	0.86 (0.68)	-0.87 (0.73)	-0.84 (0.71)	
PLAMM-2	0.66 (0.65)	0.67 (0.61)	-0.67 (0.66)	-0.74 (0.68)	
APLAMM-1	1.04 (0.90)	1.06 (0.81)	-1.11 (0.87)	-1.07 (0.87)	
APLAMM-2	0.64 (0.77)	0.61 (0.75)	-0.64 (0.81)	-0.70 (0.83)	
Results provided correct monotonicity directions:					
Mboost	0.053 (0.25)	0.074 (0.24)	-0.069 (0.21)	-0.10 (0.27)	

Table 8: True and false positives, mean squared prediction errors (PE), mean squared estimation errors (MSE), and estimated linear coefficients, for $n = 100$, $d_1 = 100$, $d_2 = 500$, $\text{SNR} \approx 4$ and $t = 0$. PLAMM-1 selected an average $\alpha = 0.91$ (0.13). Standard deviations are given in parentheses. "TPM" and "FPM" denote true and false positives among the monotone covariates, respectively, and "TPL" and "FPL" denote true and false positives among the linear covariates, respectively.

Selection:	TPM	TPL	FPM	FPL	PE
PLAMM-1	4.0 (0)	4.0 (0)	29.26 (13.09)	17.10 (10.44)	0.49 (0.11)
PLAMM-2	4.0 (0)	3.96 (0.40)	38.5 (18.32)	12.16 (10.46)	0.72 (0.57)
APLAMM-1	4.0 (0)	4.0 (0)	8.41 (5.43)	8.45 (6.62)	0.36 (0.089)
APLAMM-2	4.0 (0)	3.84 (0.68)	16.42 (10.41)	0.44 (0.98)	0.99 (0.93)

Results provided correct monotonicity directions:					
Mboost	3.91 (0.38)	3.90 (0.48)	35.84 (26.44)	1.04 (1.30)	0.90 (0.43)

MSE:	g_1	g_2	g_3	g_4	
PLAMM-1	0.032 (0.025)	0.028 (0.020)	0.050 (0.023)	0.043 (0.021)	
PLAMM-2	0.041 (0.040)	0.034 (0.028)	0.060 (0.048)	0.053 (0.043)	
APLAMM-1	0.014 (0.015)	0.013 (0.0086)	0.019 (0.010)	0.018 (0.013)	
APLAMM-2	0.040 (0.051)	0.035 (0.032)	0.051 (0.061)	0.045 (0.054)	

Results provided correct monotonicity directions:					
Mboost	0.071 (0.058)	0.069 (0.056)	0.094 (0.085)	0.085 (0.088)	

	β_1	β_2	β_3	β_4	
PLAMM-1	1.69 (0.25)	1.65 (0.25)	-1.66 (0.23)	-1.67 (0.24)	
PLAMM-2	1.42 (0.53)	1.41 (0.50)	-1.40 (0.51)	-1.41 (0.50)	
APLAMM-1	1.91 (0.20)	1.88 (0.17)	-1.90 (0.19)	-1.89 (0.20)	
APLAMM-2	1.28 (0.69)	1.28 (0.67)	-1.25 (0.68)	-1.28 (0.67)	

Results provided correct monotonicity directions:					
Mboost	1.12 (0.44)	1.11 (0.43)	-1.11 (0.40)	-1.13 (0.45)	

Table 9: Estimated linear parameters and prediction errors with the different methods for the prostate cancer data. The estimated α for PLAMM-1 on the full data was 1.5. The estimated α on the training data for PLAMM-1 was 1.33 (standard deviation 0.54).

Estimated linear coefficients.				
	PLAMM-1	PLAMM-2	APLAMM-1	APLAMM-2
Svi	0.75	0.78	0.61	0.72
Gleason	0.41	0.43	0.28	0.37

	Cgam	Mboost	Scam	
Svi	0.65	0.42	-0.56	
Gleason	0.42	0.067	0.59	
Estimated mean squared prediction errors				
	PLAMM-1	PLAMM-2	APLAMM-1	APLAMM-2
Prediction error	0.63	0.60	0.65	0.65
Standard deviation	0.17	0.13	0.16	0.14

	Cgam	Mboost	Scam	
Prediction error	0.79	0.61	0.65	
Standard deviation	0.17	0.13	0.18	

Table 10: Estimated linear parameters and prediction errors with the different methods for the bone mineral density data. The estimated α for PLAMM-1 on the full data set was 0.75. The estimated α for PLAMM-1 on the training data sets was 1.25 (standard deviation 0.48). "Std. dev." is the standard deviation and "Ad. Lasso" is the adaptive lasso.

Estimated linear coefficients							
	PLAMM-1	PLAMM-2	APLAMM-1	APLAMM-2	Mboost	Lasso	Ad. Lasso
Age	-0.23	-0.038	-0.27	-0.024	-	-0.012	-
BMI	0.38	0.15	0.49	0.15	0.16	0.074	0.070
Pth	-0.095	-0.088	-0.053	-0.036	-	-0.043	-
Vitamin D	-	-0.0041	-	-	-	-	-
Cpt	-0.054	-0.10	-	-0.089	-	-	-
Estimated prediction errors							
Estimate	1.82	1.95	2.11	2.15	1.97	2.06	2.02
Std. dev.	0.32	0.39	0.60	0.60	0.30	0.34	0.32