MONASH University

MONASH BUSINESS SCHOOL

ISSN 1440-771X

Department of Econometrics and Business Statistics

http://monash.edu/business/ebs/research/publications

Sparse Multiple Index Models for High-dimensional Nonparametric Forecasting

Nuwani Palihawadana, Rob J Hyndman, Xiaoqian Wang

March 2024

Working Paper no/yr

Sparse Multiple Index Models for High-dimensional Nonparametric Forecasting

Nuwani Palihawadana
Department of Econometrics & Business Statistics
Clayton VIC 3800
Australia
Email: nuwani.kodikarapalihawadana@monash.edu
Corresponding author

Rob J Hyndman
Department of Econometrics & Business Statistics
Clayton VIC 3800
Australia
Email: rob.hyndman@monash.edu

Xiaoqian Wang
Department of Econometrics & Business Statistics
Clayton VIC 3800
Australia
Email: xiaoqian.wang@monash.edu

7 March 2024

JEL classification: C10,C14,C22

Sparse Multiple Index Models for High-dimensional Nonparametric Forecasting

Abstract

High-dimensionality is a common phenomenon in real-world forecasting problems. Oftentimes, forecasts are contingent on a long history of predictors, while the relationships between some predictors and the response of interest exhibit complex nonlinear patterns. In such a situation, a nonlinear "transfer function" model, with additivity constraints to mitigate the issue of *curse of dimensionality*, is a conspicuous choice. Particularly, nonparametric *additive index models* greatly reduce the number of parameters to be estimated in comparison to a general additive model. In this paper, we present a novel algorithm for estimating high-dimensional nonparametric additive index models, with simultaneous variable selection, which we call ***SMI*** (***S***parse ***M***ultiple ***I***ndex) ***Model***. The SMI Modelling algorithm is based on an iterative procedure that applies mixed integer programming to solve an ℓ_0-regularised nonlinear least squares problem. We demonstrate the functionality and the characteristics of the proposed algorithm through a simple simulation exercise. We also illustrate the use of the SMI Modelling algorithm in two empirical applications related to forecasting heat exposure related daily mortality and daily solar intensity.

Keywords: Additive index models, Variable selection, Dimension reduction, Mixed integer programming

1 Introduction

Forecasts are often contingent on a very long history of predictors. For example, when forecasting half-hourly electricity demand, it is common to use at least a week of historical half-hourly temperatures and other weather observations (Hyndman & Fan 2010). Similarly, when forecasting bore levels, rainfall data from up to thousand days earlier can impact the result (Bakker & Schaars 2019) due to the complex flow dynamics of rainfall into aquifers.

On the other hand, in most of these applications, the relationships between the predictors and the response variable exhibit complex nonlinear patterns. For instance, the relationship between

electricity demand and temperature is often nonlinear (Hyndman & Fan 2010; Fan & Hyndman 2012).

These examples suggest a possible nonlinear "*transfer function*" model of the form

$$y_t = f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \ldots, \boldsymbol{x}_{t-p}, y_{t-1}, \ldots, y_{t-k}) + \varepsilon_t, \tag{1}$$

where y_t is the observation of the response variable at time t, $\boldsymbol{x}_t$ is a vector of predictors at time t, and ε_t is the random error. By including lagged values of y_t along with the lagged predictors, we allow for any serial correlation in the data. However, it makes the resulting function difficult to interpret. An alternative formulation is

$$y_t = f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \ldots, \boldsymbol{x}_{t-p}) + g(\varepsilon_t, \varepsilon_{t-1}, \ldots, \varepsilon_{t-k}),$$

which is more difficult to estimate, but makes it simpler to interpret the effect of the predictors on the response variable.

When applying the transfer function model to forecast lengthy time series with complex patterns, the form of f is nonlinear, involving complicated interactions, and with a high value of p.

Typically, the form of f involves many ad hoc model choices. It is essentially impossible to estimate a p-dimensional function for large p due to the curse of dimensionality (Bellman 1957; Stone 1982). Instead, we normally impose some form of additivity, along with some low-order interactions.

For example, Fan & Hyndman (2012) proposed a ***semi-parametric additive model*** to obtain short-term forecasts of the half-hourly electricity demand for power systems in the Australian National Electricity Market. In this model, f is assumed to be fully additive, and is used to capture the effects of recent predictor values on the demand. The main objective behind the use of this proposed semi-parametric model is to allow nonparametric components in a regression-based modelling framework with serially correlated errors (Fan & Hyndman 2012). The model fitted for each half-hourly period (q) can be written as

$$\log(y_{t,q}) = h_q(t) + f_q(w_{1,t}, w_{2,t}) + a_q(y_{t-,p}) + \varepsilon_t,$$

where the response variable is the logarithm of electricity demand at time t (measured in the half-hourly intervals) during period q. The term $h_q(t)$ models several calendar effects that are included as linear terms. The temperature effects are modelled using the nonparametric

component $f_q(w_{1,t}, w_{2,t})$, while the nonparametric term $a_q(y_{t-,p})$ captures the lagged effects of the response. It is important to notice here that the error term ε_t is serially uncorrelated in each half-hourly model, because the serial correlation is eliminated by the inclusion of the lagged responses in the model. However, there will still be some correlation between the residuals from the various half-hourly models (Fan & Hyndman 2012).

Similarly, a ***distributed lag model*** was proposed by Wood (2017) to forecast daily death rate in Chicago using measurements of several air pollutants. In this model, the response variable is modelled via a sum of smooth functions of lagged predictor variables, which is quite similar in nature to the semi-parametric additive model used by Fan & Hyndman (2012). However, unlike in Fan & Hyndman (2012), Wood (2017) suggested to allow the smooth functions for lags of the same covariate to vary smoothly over lags, preventing large differences in estimated effects between adjacent lags. Thus, the model is of the form

$$\log(y_t) = f_1(t) + \sum_{k=0}^{K} f_2(p_{t-k}, k) + \sum_{k=0}^{K} f_3(o_{t-k}, w_{t-k}, k),$$

where y_t is the death rate at day t, and f_1 is a nonparametric term to capture the *time* effect. The model incorporates the current value ($k = 0$) and several lagged values ($k = 1, \ldots, K$) of the predictors, where the *distributed lag effect* of a single predictor variable, and of an interaction of two predictor variables are captured by the sum of nonparametric terms $\sum_{k=0}^{K} f_2(p_{t-k}, k)$ and $\sum_{k=0}^{K} f_3(o_{t-k}, w_{t-k}, k)$ respectively. The smooth functions f_2 and f_3 are proposed to be estimated using *tensor product smooths*.

For more examples, Ho, Chen & Hwang (2020) used semi-paramteric additive models to estimate ground-level $PM_{2.5}$ concentrations in Taiwan, while nonparametric additive models were utilised by Ibrahim et al. (2022) for predicting census survey response rates. Furthermore, Ravindra et al. (2019) provided a comprehensive review of the applications of additive models for environmental data, with a special focus on air pollution, climate change, and human health related studies.

While such models have been used to address problems including electricity demand, air quality related mortality rate and groundwater level forecasting etc. (Fan & Hyndman 2012; Hyndman & Fan 2010; Wood 2017; Peterson & Western 2014; Rajaee, Ebrahimi & Nourani 2019), there are still a number of unresolved issues in their applications. In this paper, we attempt to address two of those issues. Firstly, even though nonparametric additive models act as a remedy to the curse of dimensionality as we discussed earlier, the estimation of the model is still challenging in a high-dimensional setting due to the large number of nonparametric components to be

estimated. Secondly, there is a noticeable subjectivity in the selection of predictor variables (from the available predictors) for the model, where in most of the applications of interest we discussed above, the predictor choices in the final model are mainly based on empirical explorations or domain expertise.

There are a number of previous studies that have attempted to address the issue of variable selection in nonparametric/semi-parametric additive models to some extend, using various techniques. For example, Huang, Horowitz & Wei (2010) used a *Least Absolute Shrinkage and Selection Operator (LASSO)* (Tibshirani 1996) based procedure for variable selection in nonparametric additive models, whereas Fan & Hyndman (2012) used a straightforward backward elimination technique to achieve selection. Moreover, Ibrahim et al. (2022) and Hazimeh, Mazumder & Radchenko (2023) used Mixed Integer Programming based methodologies to provide a solution to the *best subset selection* problem in nonparametric additive models. More details of these methods are discussed later in Section 2.

In this paper, however, we are interested in high-dimensional applications that exhibit complicated interactions among predictors (specially in the presence of large number of lagged variables), as well as correlated errors. In such a situation, *"index models"* (refer Section 2.2) seem to be useful for improving the flexibility of the broader class of nonparametric additive models (Radchenko 2015), while mitigating the difficulty of estimating a nonparametric component for each individual predictor.

To our knowledge, no previous research has been done to look at how the predictor choices can be made more objective and principled in nonparametric additive index models. Hence, our goal was to develop a methodology for optimal predictor selection in the context of high-dimensional nonparametric additive index models. Moreover, due to computational advancements in the field, the use of *Mathematical Optimisation* concepts in solving statistical problems has gained a lot of interest in the recent past (Theußl, Schwendinger & Hornik 2020). This motivated us to develop a variable selection algorithm based on mathematical optimisation techniques.

Additionally, it is crucial to point out that any such variable selection methodology naturally renders inferential statistics invalid, since we do not assume the resulting model obtained through the variable selection procedure to represent the true data generating process. Hence, our focus in this paper is only on improving forecasts, but not on making inferences on the resulting parameter estimates.

The rest of this paper is organised as follows. In Section 2, we provide a concise exposition of related ideas and previous work, while establishing the foundation for this paper. Section 3 presents our proposed model, *Sparse Multiple Index Model* (SMI Model), and describes the variable selection algorithm and estimation procedure. In Section 4, we demonstrate the functionality and the characteristics of the proposed algorithm through a simulation experiment. Section 5 illustrates two empirical applications of the proposed estimation and variable selection methodology, related to forecasting heat exposure related daily mortality and daily solar intensity. Concluding remarks are given in Section 6.

2 Background

2.1 Variable Selection in Nonparametric Additive Models

As discussed in Section 1, the estimation of nonparametric function f (Equation 1) becomes infeasible in high-dimensional settings (i.e. number of predictors is very large) due to curse of dimensionality. As a result, *nonparametric additive models* have been employed with growing popularity. Let $(y_i, \boldsymbol{x}_i), i = 1, \ldots, n$, be independent and identically distributed (i.i.d) observations, and $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T$ be a p-dimensional vector of predictor values. Then a nonparametric additive model can be written as

$$y_i = \sum_{j=1}^{p} f_j(x_{ij}) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{2}$$

where f_j's are unknown functions (probably non-linear and smooth), and ε_i is the random error (Lian 2012). Even such an additivity condition is imposed, estimating the optimal predictive model will still be troublesome when p is very large (probably even larger than the sample size n) due to over-fitting (Lian 2012). Thus, it is natural to bring in the sparsity assumption, and assume that some of f_j's are zero, which gives rise to the need of a variable selection method to differentiate between zero and non-zero components, while estimating the non-zero components (Huang, Horowitz & Wei 2010).

2.1.1 Backward Elimination

In the problem of forecasting long-term peak electricity demand, Hyndman & Fan (2010) used a stepwise procedure for variable selection through cross-validation. In the each half-hourly model fitted, the data is split into training and validation sets, and the predictors are selected into the model based on the Mean Squared Error (MSE) calculated for the validation set. Starting

from the full model, the predictive power of each variable is evaluated by dropping one at a time. A predictor, the removal of which contributed to a decrease in the validation MSE, is omitted from the model in subsequent steps (Hyndman & Fan 2010). Fan & Hyndman (2012) used a similar method except for the fact that they considered the Mean Absolute Percentage Error (MAPE) as the selection criterion. Therefore, both of these prior work use stepwise variable selection methodology based on out-of-sample forecasting performance.

2.1.2 Penalisation Methods

According to Huang, Horowitz & Wei (2010), there are numerous penalised methods for variable selection and parameter estimation in high-dimensional settings, including the *bridge estimator* proposed by Frank & Friedman (1993), the *Least Absolute Shrinkage and Selection Operator (LASSO)* by Tibshirani (1996), the *Smoothly Clipped Absolute Deviation Penalty* (SCAD) by Fan & Li (2001), and the *Minimum Concave Penalty* (MCP) by Zhang (2010). Among them, we observe that the LASSO and the SCAD penalties are appearing popularly in literature.

Tibshirani (1996) introduced the regularisation method, ***LASSO***, for estimating linear models, which minimises the sum of squared residuals subject to the ℓ_1 penalty on the coefficients. Assume the classical linear regression model $y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i$, fitted for the data $(y_i, \boldsymbol{x_i})$, $i = 1, \ldots, n$, where y_i is the response, $\boldsymbol{x_i} = (x_{i1}, \ldots, x_{ip})^T$ is a p-dimensional vector of predictors, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is the parameter vector corresponding to $\boldsymbol{x_i}$, and ε_i is the random error. Then, the LASSO estimator, $\hat{\boldsymbol{\beta}}_{LASSO}$, can be obtained by

$$\hat{\boldsymbol{\beta}}_{LASSO} = \min_{\boldsymbol{\beta}} \left\{ \left\| \boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x_j} \beta_j \right\|_2^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

where $\boldsymbol{x_j} = \left(x_{1j}, \ldots, x_{nj}\right)^T$, and λ is a non-negative tuning parameter. The LASSO estimator reduces to the Ordinary Least Squares (OLS) estimator if λ is equal to zero (Konzen & Ziegelmann 2016). Due to the nature of the penalty applied, LASSO shrinks some of the coefficients towards zero, and sets the others exactly to zero, where the estimation of coefficients and variable selection are performed simultaneously (Konzen & Ziegelmann 2016).

While showing that the LASSO is not consistent for variable selection in certain situations, Zou (2006) introduced ***Adaptive Lasso*** (popularly known as "adaLASSO"); an extension of the LASSO method, which uses adaptive weights to penalise coefficients using the LASSO (i.e. ℓ_1)

penalty. Thus, the adaLASSO objective function can be written as

$$\hat{\boldsymbol{\beta}}_{adaLASSO} = \min_{\boldsymbol{\beta}} \left\{ \left\| \boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x}_j \beta_j \right\|_2^2 + \lambda \sum_{j=1}^{p} w_j |\beta_j| \right\},$$

where the vector of weights $\boldsymbol{w} = \left(w_1, \ldots, w_p\right)^T$ is estimated by $\hat{\boldsymbol{w}} = 1/|\hat{\boldsymbol{\beta}}|^\gamma$ for $\gamma > 0$, which is a tuning parameter, and $\hat{\boldsymbol{\beta}}$ being any consistent estimator of $\boldsymbol{\beta}$ (Zou 2006).

Yuan & Lin (2006) considered the problem of selecting groups of variables, and discussed extensions of three variable selection and estimation methods namely, *LASSO* (Tibshirani 1996), *Least Angle Regression Selection* (LARS, Efron et al. 2004), and *Non-negative Garrotte* (Breiman 1995). Consider an n-dimensional response vector $\boldsymbol{y}$, and an $n \times p$ matrix of predictor values $\boldsymbol{X}$. Then the ***Group Lasso*** estimator of the coefficients vector $\boldsymbol{\beta}$ is obtained by minimising

$$\frac{1}{2} \left\| \boldsymbol{y} - \sum_{\ell=1}^{L} \boldsymbol{X}_\ell \boldsymbol{\beta}_\ell \right\|_2^2 + \lambda \sum_{\ell=1}^{L} \|\boldsymbol{\beta}_\ell\|_{\boldsymbol{K}_\ell},$$

where $\boldsymbol{X}_\ell$ is an $n \times p_\ell$ sub-matrix in $\boldsymbol{X}$ that corresponds to the ℓ^{th} group of predictors (p_ℓ is the number of predictors in ℓ^{th} group), $\boldsymbol{\beta}_\ell$ is the corresponding vector of coefficients, $\ell = 1, \ldots, L$, $\|\boldsymbol{\beta}_\ell\|_{\boldsymbol{K}_\ell} = (\boldsymbol{\beta}_\ell' \boldsymbol{K}_\ell \boldsymbol{\beta}_\ell)^{\frac{1}{2}}$ with $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_L$ being a set of given positive definite matrices, and λ is a non-negative tuning parameter. Moreover, Simon et al. (2013) proposed ***Sparse-Group Lasso***, which is a convex combination of general Lasso and Group Lasso methods, where the focus is on both "groupwise sparsity" (the number of groups with at least one nonzero coefficient), and "within group sparsity" (the number of nonzero coefficients within each nonzero group).

According to Fan & Li (2001), a penalty function used in penalised least squares approaches should have three properties. Firstly, it should be singular at origin to generate a solution that is sparse. Secondly, it should fulfill certain conditions to be stable in model selection. Finally, it should be able to generate unbiased estimates for large coefficients via being bounded by a constant. They argued that all those three conditions are not satisfied by the penalisation methods such as the bridge regression (Frank & Friedman 1993) and the LASSO (Tibshirani 1996). Hence they proposed the ***SCAD*** penalty function, which is defined in terms of its first derivative as

$$p_\lambda'(\theta) = \lambda \left\{ I(\theta \leq \lambda) + \frac{(a\lambda - \theta)_+}{(a-1)\lambda} I(\theta > \lambda) \right\},$$

for some $a > 2$, and $\theta > 0$ (Fan & Li 2001). According to Fan & Li (2001), the SCAD penalty function retains the favourable properties of both best subset selection and ridge regression, while having all three desired features, i.e., sparsity, stability, and unbiasedness.

Based on the above penalisation methods that are originally developed for linear models, Huang, Horowitz & Wei (2010) proposed a new penalisation method for variable selection in nonparametric additive model (Equation 2), named ***Adaptive Group Lasso***. They approximated f_j's using normalised B-spline bases, so that a linear combination of B-spline basis functions is used to represent an individual nonparametric component f_j. The proposed method is a generalisation of Adaptive Lasso method (Zou 2006) to the Group Lasso method (Yuan & Lin 2006).

When the nonparametric additive model in Equation 2 is considered, an obvious possibility is that some of the additive components (i.e. f_j's) are being linear. For example, recall the electricity demand forecasting problem (Hyndman & Fan 2010; Fan & Hyndman 2012), where some of the calendar effects are included into the model as linear variables, whereas lagged temperature and lagged demand variables are included using nonlinear additive components. Such situations suggest the use of *semi-parametric partially linear additive models* that can be mathematically represented as

$$y_i = \sum_{j=1}^{p} f_j(x_{ij}) + \sum_{k=1}^{q} w_{ik}\beta_k + \varepsilon_i, \quad i = 1, \ldots, n,$$

where $\boldsymbol{x}_j$'s, $j = 1, \ldots, p$, are a set of predictors that enter the model as nonparametric components, whereas $\boldsymbol{w}_k$'s, $k = 1, \ldots, q$ are another set of predictors that are included as linear components. While several studies have assumed that the number of nonparametric components are fixed, and performed variable selection only among the linear components of the model (Lian 2012; Guo et al. 2013; Liu, Wang & Liang 2011), Wang et al. (2014) introduced a methodology for selecting both linear and nonlinear components simultaneously, in the context of correlated, longitudinal data. They proposed the use of a ***Penalised Quadratic Inference Function (PQIF) with double SCAD penalties*** for variable selection and model estimation, where the correlation structure of the data was incorporated into the estimation method (see Wang et al. (2014) for details).

2.1.3 Time Series Aspect

It is worthwhile to briefly mention that there are extensions of the penalisation methods discussed above, which have specifically proposed to take the autocorrelation and lag structures in time series data into account.

Wang, Guodong & Tsai (2007) proposed an extension of the LASSO method for Regression with Autoregressive Error (REGAR) models. Park & Sakaori (2013) and Konzen & Ziegelmann (2016) proposed modifications to Adaptive Lasso method to incorporate the lag structures presented

in Autoregressive Distributed Lag (ADL) models into the variable selection and estimation methodology. The ***Ordered Lasso*** was introduced by Tibshirani & Suo (2016) to deal with time-lagged regression problems, where we forecast the response value at time t using the predictor values from K previous time points, assuming that the magnitude of regression coefficients decreases as the lagged predictor moves away from time t.

However, it is important to note that all the models considered in the above time series related work are linear; none of them include nonparametric terms.

2.2 Index Models

2.2.1 Single Index Model

The nonparametric additive model (Equation 2) estimates the relationship between the response and the predictors using a sum of univariate nonlinear functions corresponding to each individual predictor variable. Hence, it is incapable of handling the interactions among the predictors, which are ubiquitous in real-world problems (Zhang et al. 2008).

As a remedy, the ***Single Index Model***, a generalisation of the linear regression model where the linear predictor is replaced by a semi-parametric component, is popularly being used in the literature (Radchenko 2015). Let y_i be the response, and $\boldsymbol{x}_i$ be a p-dimensional predictor vector. Then the single index model can be written as

$$y_i = g\left(\boldsymbol{\alpha}^T \boldsymbol{x}_i\right) + \varepsilon_i, \quad i = 1, \ldots, n,$$

where $\boldsymbol{\alpha}$ is a p-dimensional vector of unknown coefficients (i.e. parameters), g is an unknown univariate function, and ε_i is the random error (Stoker 1986; Härdle, Hall & Ichimura 1993). The linear combination $\boldsymbol{\alpha}^T \boldsymbol{x}_i$ is called the *index*. Single index model is viewed as a viable alternative to the additive model since it offers more flexibility and interpretability (Radchenko 2015).

According to Radchenko (2015), single index models have widely been used in scenarios with fairly low and moderate dimensionality, where the corresponding estimation and variable selection techniques are not directly applicable to the high-dimensional setting. The error sum of squares of the model being non-convex with respect to index coefficients, is the main reason behind the existence of very limited number of methods in high-dimensional case (Radchenko 2015). For an extensive summary of available methods, we refer to Radchenko (2015).

2.2.2 Multiple Index Models

Projection Pursuit Regression

Friedman & Stuetzle (1981) introduced ***Projection Pursuit Regression (PPR)*** by extending the nonparametric additive model (Equation 2) to enable the modelling of interactions among predictor variables. On the other hand, PPR is an extension of the single index model to an ***Additive Index Model*** , given by

$$y_i = \sum_{j=1}^{q} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_i) + \varepsilon_i, \quad i = 1, \ldots, n,$$

where y_i is the response, $\boldsymbol{x}_i$ is a p-dimensional predictor vector, $\boldsymbol{\alpha}_j = \left(\alpha_{j1}, \ldots, \alpha_{jp}\right)^T, j = 1, \ldots, q$ are p-dimensional projection vectors (or vectors of *"index coefficients"*), g_j's are unknown univariate functions, and ε_i is the random error.

Instead of estimating a single index, PPR estimates multiple indices and connects them to the response through a sum of univariate nonlinear functions. These indices are constructed through a *Projection Pursuit (PP)* (Kruskal 1969; Friedman & Tukey 1974) algorithm, which is considered to be "interesting" low-dimensional projections of a high-dimensional feature space, obtained through the maximisation of an appropriate objective function or a "projection index" (Huber 1985).

According to Zhang et al. (2008), PPR increases the power of additive models in high-dimensional settings, but it has two major drawbacks. Firstly, since PP increases the freedom of the additive model, it tends to overfit in a situation, where there are a lot of unimportant predictors. Secondly, the interpretation of the model estimated by PPR will be troublesome as many non-zero elements will be present in each projection vector $\boldsymbol{\alpha}_j$. To overcome these issues, Zhang et al. (2008) introduced an ℓ_1 regularised projection pursuit algorithm, where the resultant regression model is named as ***Sparse Projection Pursuit Regression*** (SpPPR). In SpPPR, an ℓ_1 penalty (i.e. a LASSO penalty) on index coefficients is added to the cost function (the squared error) at each iteration of the PP, thereby performing variable selection and model estimation simultaneously. See Zhang et al. (2008) for more details.

Although Zhang et al. (2008) claimed that the SpPPR algorithm can detect important predictors even in a noisy data set, our experiments show that it is not particularly scalable for large data sets with both higher number of predictors and observations.

Group-wise Additive Index Model

Even though PPR introduces flexibility and the ability to model interactions among predictors into additive models, the indices obtained through PPR contain all the predictors at hand. Hence, even with a variable selection mechanism like SpPPR (Zhang et al. 2008), PPR creates indices possibly by mixing heterogeneous variables in a single linear combination, making very little sense in terms of interpretability (Masselot et al. 2022).

Typically, in many real-world problems, natural groupings can be identified in predictor variables. For example, naturally interacting variables can be grouped together, such as several lags of a predictor, weather related variables, and genes or proteins that are grouped by biological pathways in a biological study (Masselot et al. 2022; Wang, Xu & Zhu 2015).

This suggests the use of a ***Group-wise Additive Index Model (GAIM)***, which can be written as

$$y_i = \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}) + \varepsilon_i, \quad i = 1, \ldots, n,$$

where y_i is the univariate response, $\boldsymbol{x}_{ij} \in \mathbb{R}^{l_j}$, $j = 1, \ldots, p$ are naturally occurring p groups of predictors, which are p non-overlapping subsets of $\boldsymbol{x}_i$ - the vector of all predictors, $\boldsymbol{\alpha}_j$ is a l_j-dimensional vector of index coefficients corresponding to the index $h_{ij} = \boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}$, g_j is an unknown (possibly nonlinear) component function, and ε_i is the random error, which is independent of $\boldsymbol{x}_i$ (Wang et al. 2015; Masselot et al. 2022).

Since GAIM uses groups of predictors that are naturally or logically belonging together to construct indices, such derived indices will be more expressive and interpretable. However, at the same time, this introduces a certain level of subjectivity into the model formulation as different users can group the available predictors in different ways based on different logical reasoning.

In this paper, our aim is to reduce that subjectivity induced by personal judgment or domain expertise. Hence, we propose a methodology that injects more objectivity into the estimation of multiple index models by algorithmically grouping predictors into indices, resulting in a model with a higher predictive accuracy.

Constrained Group-wise Additive Index Model

The ***Constrained Group-wise Additive Index Model (CGAIM)*** was proposed by Masselot et al. (2022) for constructing comprehensive and easily interpretable indices from a large set of

explanatory variables. The model of interest is a *semi-parametric group-wise additive index model* given by

$$y_i = \beta_0 + \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}) + \sum_{k=1}^{d} f_k(w_{ik}) + \boldsymbol{\theta}^T \boldsymbol{u}_i + \varepsilon_i, \quad i = 1, \ldots, n,$$

where y_i is the univariate response, β_0 is the model intercept, $\boldsymbol{x}_{ij} \in \mathbb{R}^{l_j}$, $j = 1, \ldots, p$ are naturally occurring p groups of predictor vectors (i.e. it is assumed that the predictor groupings are known in advance), which are p subsets of $\boldsymbol{x}_i$ - the q-dimensional vector of all predictors entering indices, $\boldsymbol{\alpha}_j$ is the vector of index coefficients corresponding to the index $h_{ij} = \boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}$, and g_j is the corresponding nonlinear link function (possibly estimated by a spline). The additional predictor variables that are helpful in predicting y_i, but do not enter any of the indices are two-fold: a covariate that relates to the response through a nonlinear function f_k, denoted by w_{ik}, and the vector of linear covariates denoted by $\boldsymbol{u}_i$.

This is an extension of the GAIM that allows to impose constraints on the index coefficients as well as on the nonlinear link functions. In CGAIM, linear constraints of the form $\boldsymbol{C}_j \boldsymbol{\alpha}_j \geq 0$ can be imposed on the index coefficients, where $\boldsymbol{C}_j \in \mathbb{R}^{d_j \times l_j}$, and d_j is the number of constraints. Moreover, shape constraints such as monotonicity, convexity or concavity can be imposed on the nonparametric functions. This modification allows to incorporate prior knowledge or operational requirements into the model estimation.

First, considering only the additive index part of the model, and given $(y_i, x_{i1}, \ldots, x_{iq}), \quad i = 1, \ldots, n$ be the observed data, where the q predictors are grouped into p groups, the estimation problem of the CGAIM can be formulated as

$$\begin{aligned} \min_{\boldsymbol{\alpha}, \beta_0} \quad & \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}) \right]^2, \\ \text{s.t.} \quad & \boldsymbol{C}\boldsymbol{\alpha} \geq 0, \quad g_j \in m, \end{aligned} \tag{3}$$

where $\boldsymbol{\alpha} = \left[\boldsymbol{\alpha}_1^T, \ldots, \boldsymbol{\alpha}_p^T \right]^T$, β_0 is the model intercept, $\boldsymbol{C} \in \mathbb{R}^{d \times q}$, d is the number of constraints on the index coefficients vector $\boldsymbol{\alpha}$, and m is a shape constraint imposed on g_j (Masselot et al. 2022).

Notice that $\boldsymbol{\alpha}_j$s behave non-linearly in Equation 3, and hence, this is a non-linear least squares problem. Accordingly, Masselot et al. (2022) introduced an efficient iterative algorithm for estimating the CGAIM based on ***Sequential Quadratic Programming*** (SQP), one of the most successful techniques for solving nonlinear constrained optimisation problems (Boggs & Tolle 1995). For details of the CGAIM algorithm refer to Masselot et al. (2022).

2.3 Mathematical Optimisation for Variable Selection

2.3.1 Mathematical Optimisation

Optimisation plays a major role in both decision science and physical systems evaluation. ***Mathematical Optimisation*** or ***Mathematical Programming*** can be defined as the minimisation (or maximisation) of a function subject to restrictions on the unknowns/parameters of that function (Nocedal & Wright 2006). Hence, a mathematical optimisation problem can be written as

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f_0(\boldsymbol{x}) \\ \text{s.t.} \quad & f_i(\boldsymbol{x}) \leq b_i, \quad i = 1, \ldots, m \end{aligned} \tag{4}$$

where the vector of unknowns or parameters of the problem is given by $\boldsymbol{x} = (x_1, \ldots, x_n)^T$, the *objective function* is denoted by $f_0 : \mathbb{R}^n \to \mathbb{R}$, the *constraint functions* are given by $f_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, m$, and the bounds of the constraints are denoted by $\boldsymbol{b} = (b_1, \ldots, b_m)^T$. A vector of values $\boldsymbol{x}^*$ that results in the smallest value for the objective function among all vectors that satisfy the stated constraints, is called the *optimal* value or the *solution* to the problem (Boyd & Vandenberghe 2004). After mathematically formulating the optimisation problem as above (Equation 4), an appropriate *optimisation algorithm* is used to obtain the solution $\boldsymbol{x}^*$ (Nocedal & Wright 2006).

Based on the form of the objective function and the constraints, various types of optimisation problems are identified.

An optimisation problem is known as a ***Linear Programming*** (LP) when both the objective function and the constraints in Equation 4 (i.e. all f_i, $i = 0, \ldots, m$) are linear. Hence, a LP can be written as

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \boldsymbol{a_0}^T \boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \end{aligned} \tag{5}$$

where $\boldsymbol{x}$ is the vector that contains the parameters to be optimised, and $\boldsymbol{a_0} \in \mathbb{R}^n$ is the vector of coefficients of the objective function. The matrix of coefficients in the constraints is denoted by $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, and $\boldsymbol{b}$ is the vector containing the upper bounds of the constraints. All LPs are *convex* optimisation problems (Theußl, Schwendinger & Hornik 2020).

The LP problem given in Equation 5 can be generalised to involve a quadratic term in the objective function, in which case it is called a ***Quadratic Programming*** (QP). A QP can be written

as

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q_0}\boldsymbol{x} + \boldsymbol{a_0}^T\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \end{aligned}$$

where $\boldsymbol{Q_0} \in \mathbb{R}^{n \times n}$. Unless the matrix $\boldsymbol{Q_0}$ is positive semi-definite, a QP is non-convex (Theußl, Schwendinger & Hornik 2020).

If a linear objective function is minimised over a *convex cone*, such an optimisation problem is called a ***Conic Programming*** (CP), which can be written as

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \boldsymbol{a_0}^T\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} + s = \boldsymbol{b}, \quad s \in \mathcal{K}, \end{aligned}$$

where $\mathcal{K}$ denotes a nonempty closed convex cone. CPs are designed to model convex optimisation problems (Theußl, Schwendinger & Hornik 2020).

If we restrict some of the unknowns/parameters in an optimisation problem to take only integer values, then that optimisation problem is called a ***Mixed Integer Programming*** (MIP). For example, if we constraint $x_k \in \mathbb{Z}$ for at least one k, $k \in \{1, \ldots, n\}$ in the optimisation problem given by Equation 4, then the optimisation problem becomes a MIP. If all the unknowns of an optimisation problem are constrained to be integers, such a problem is referred to as a pure ***Integer Programming*** (IP), whereas if all the unknowns are bounded between zero and one (i.e. $\boldsymbol{x} \in \{0,1\}^n$), the optimisation problem is referred to as a ***Binary (Integer) Programming*** (Theußl, Schwendinger & Hornik 2020). MIPs are hard to solve as they are non-convex due to the integer constraints. However, a growth in the number of commercial as well as non-commercial MIP solvers has made it possible to solve MIP problems conveniently and directly.

2.3.2 Variable Selection

Mathematical optimisation is fundamentally important in statistics, as many statistical problems including regression, classification, and other types of estimation/approximation problems can be re-interpreted as optimisation problems (Theußl, Schwendinger & Hornik 2020). Thus, the problem of variable selection - one of the prolonged interests of statisticians, has also benefited from using optimisation concepts, particularly MIP and convex optimisation, in the recent past.

For example, Bertsimas, King & Mazumder (2016) used a mixed integer optimisation procedure to solve the classical best subset selection problem in a linear regression. They developed a

discrete optimisation method by extending modern first-order continuous optimisation techniques. The method can produce near-optimal solutions that would serve as warm starts for a MIP algorithm, which would choose the best k features out of p predictors. Similarly, Hazimeh & Mazumder (2020) developed fast and efficient algorithms based on coordinate descent and local combinatorial optimisation to solve the same best subset selection (or ℓ_0-regularised least squares) problem through re-formulating local combinatorial search problems as structured MIPs.

Furthermore, Hazimeh, Mazumder & Radchenko (2023) proposed a group-wise variable selection methodology, based on discrete mathematical optimisation, which is applicable to both ℓ_0-regularised linear regression and nonparametric additive models in a high-dimensional setting. They formulated the group ℓ_0-based estimation problem as a ***Mixed Integer Second Order Cone Programming*** (MISOCP), and proposed a new customised Branch-and-Bound (BnB) algorithm (Land & Doig 1960; Little et al. 1963) to obtain the global optimal solution to the MISOCP.

Through the study of above literature, we noticed that the mathematical optimisation based algorithms reduce computational cost of variable selection procedures in high-dimensional settings. This is largely due to the availability of efficient commercial solvers such as *Gurobi* and *CPLEX*. This motivated us to focus on a mathematical optimisation based procedure for developing our variable selection methodology.

3 Sparse Multiple Index Model

In this section, we develop a ***Sparse Multiple Index Model*** (hereafter referred to as SMI Model) to establish an objective and a principled methodology for estimating high-dimensional nonparametric additive index models, highlighting predictor grouping along with optimal predictor selection.

3.1 The SMI Model

The model of interest is a semi-parametric additive index model, which can be written as

$$y_i = \beta_0 + \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}) + \sum_{k=1}^{d} f_k(w_{ik}) + \boldsymbol{\theta}^T \boldsymbol{u}_i + \varepsilon_i, \quad i = 1, \ldots, n, \tag{6}$$

where y_i is the univariate response, β_0 is the model intercept, $\boldsymbol{x}_{ij} \in \mathbb{R}^{l_j}$, $j = 1, \ldots, p$ are p subsets of all the predictors entering indices, $\boldsymbol{\alpha}_j$ is a vector of index coefficients corresponding to the index $h_{ij} = \boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}$, g_j is a nonlinear link function (possibly estimated by a spline). Note that we also allow for the inclusion of predictors that do not enter any of the indices. These additional predictors are two-fold: a covariate w_{ik} that relates to the response through the nonlinear function f_k, $k = 1, \ldots, d$, and linear covariates denoted by $\boldsymbol{u}_i$.

We make three main assumptions as follows to define the **SMI Model**:

1. The number of indices (i.e. the number of subsets of predictors) p is unknown;
2. The predictor grouping among indices is unknown; and
3. No predictor enters more than one indices (i.e. overlapping of predictors among indices is not allowed).

These assumptions further imply that the index coefficients $\boldsymbol{\alpha}_j$s and the nonlinear link functions g_js are also unknown, and will need to be estimated.

One of the key features of the proposed SMI model is to allow for zero index coefficients for predictors, so that the predictors with zero coefficients are dropped out from the model, achieving variable selection. The other key feature of the SMI model is its flexibility in allowing a variable number of indices, ranging from 1 (i.e. all q predictors are paased to a single index) to q (i.e. each predictor goes into a separate index). Hence, both the Single Index Model and the Additive Model are special cases of the proposed SMI Model.

3.2 Optimisation Problem Formulation

Let q be the total number of predictors entering p non-overlapping subsets of size l_j, $j = 1, \ldots, p$ (i.e. $\sum_{j=1}^{p} l_j = q$). The optimisation problem we seek to address is of the form below, where the sum of the squared error of the model (Equation 6) is minimised together with an ℓ_0 penalty term and an ℓ_2 (ridge) penalty term:

$$\begin{aligned} \min_{\beta_0, p, \boldsymbol{\alpha}, \boldsymbol{g}, \boldsymbol{f}, \boldsymbol{\theta}} \quad & \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_{ij}) - \sum_{k=1}^{d} f_k(w_{ik}) - \boldsymbol{\theta}^T \boldsymbol{u}_i \right]^2 \\ & + \lambda_0 \sum_{j=1}^{p} \sum_{m=1}^{l_j} \mathbf{1}\left(\alpha_{jm} \neq 0\right) + \lambda_2 \sum_{j=1}^{p} \|\boldsymbol{\alpha}_j\|_2^2 \end{aligned} \tag{7}$$

where $\boldsymbol{\alpha} = \left[\boldsymbol{\alpha}_1^T, \ldots, \boldsymbol{\alpha}_p^T\right]^T$, $\boldsymbol{g} = \{g_1, g_2, \ldots, g_p\}$, $\boldsymbol{f} = \{f_1, f_2, \ldots, f_d\}$, $\mathbf{1}(\cdot)$ is the indicator function, $\lambda_0 > 0$ is a tuning parameter that controls the number of selected predictors entering indices, and $\lambda_2 \geq 0$ is another tuning parameter that controls the strength of the additional shrinkage imposed on the estimated index coefficients.

Applying an ℓ_2-penalty in addition to the ℓ_0-penalty is motivated by related literature (Hazimeh & Mazumder 2020; Mazumder, Radchenko & Dedieuc 2022; Hazimeh, Mazumder & Radchenko 2023), where it is suggested that the prediction performance of best-subset selection is enhanced by the inclusion of an additional ridge penalty, especially when a low signal-to-noise ratio (SNR) is present.

3.3 MIQP Formulation

To solve the optimisation problem in Equation 7, we present a big-M based ***Mixed Integer Quadratic Programming*** (MIQP) formulation:

$$
\begin{aligned}
\min_{\beta_0, p, \boldsymbol{\alpha}, \boldsymbol{g}, \boldsymbol{f}, \boldsymbol{\theta}, \boldsymbol{z}} \quad & \sum_{i=1}^{n} \left[y_i - \beta_0 - \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^T \boldsymbol{x}_i) - \sum_{k=1}^{d} f_k(w_{ik}) - \boldsymbol{\theta}^T \boldsymbol{u}_i \right]^2 + \lambda_0 \sum_{j=1}^{p} \sum_{m=1}^{q} z_{jm} + \lambda_2 \sum_{j=1}^{p} \sum_{m=1}^{q} \alpha_{jm}^2 \\
\text{s.t.} \quad & \left|\alpha_{jm}\right| \leq M z_{jm} \quad \forall j, \forall m, \\
& \sum_{j=1}^{p} z_{jm} \leq 1 \quad \forall m, \\
& z_{jm} \in \{0, 1\}, \\
& j = 1, \ldots, p, \quad m = 1, \ldots, q,
\end{aligned}
\tag{8}
$$

where p is the (unknown) number of indices, $\boldsymbol{x}_i$ is the q-dimensional vector of all predictors entering indices, and $\boldsymbol{z} = \left(\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_p^T\right)^T$, $\boldsymbol{z}_j = \left(z_{j1}, \ldots, z_{jq}\right)^T$, $j = 1, \ldots, p$ such that $z_{jm} \in \{0, 1\}, m = 1, \ldots, q$ for all j. In other words, we introduce a binary (i.e. indicator) variable corresponding to each predictor in each index. A pre-specified *big-M parameter* is denoted by $M < \infty$, and it should be sufficiently large. If $\boldsymbol{\alpha}^*$ is the optimal solution to the problem given in Equation 8, then the big-M parameter should satisfy $\max\left(\left|\alpha_{jm}^*\right|\right) \leq M$, where $j \in \{1, \ldots, p\}$, and $m \in \{1, \ldots, q\}$.

Notice that, here we formulate the MIQP to include all q predictors in each index so that in this case, $\boldsymbol{\alpha}_j$ is a q-dimensional vector of index coefficients. However at the same time, as mentioned earlier, we introduce a set of binary variables corresponding to each predictor in each index, which serves two main purposes: firstly, these binary variables are used to make the *"on-or-off"*

decisions of the predictors in the model; secondly, they contribute to decide which predictors belong to which index.

To further elaborate, first, the big-M constraints ensure that α_{jm} is zero if and only if z_{jm} is zero, and if $z_{jm} = 1$, then $|\alpha_{jm}| \leq M$. At the same time, the ℓ_0-penalty term $\lambda_0 \sum_{j=1}^{p} \sum_{m=1}^{q} z_{jm}$ influences some of the binary variables z_{jm} to be zero, while the ℓ_2-penalty term $\lambda_2 \sum_{j=1}^{p} \sum_{m=1}^{q} \alpha_{jm}^2$ enforces additional shrinkage on the estimated coefficients. Therefore, these components together perform a variable selection.

Next, when a set of binary variables $\{z_{1m}, z_{2m}, \ldots, z_{pm}\}$ corresponding to the m^{th}, $m = 1, \ldots, q$, predictor in all p indices is considered, according to the constraint $\sum_{j=1}^{p} z_{jm} \leq 1$, only one or no binary variables in the set can take the value one, ensuring that the m^{th} predictor does not repeat in more than one index. If all the elements of the set are zero, then the m^{th} predictor will be dropped out from the model.

Thus, our main contribution in this paper is two-fold. Firstly, we propose a novel algorithm to objectively estimate a semi-parametric additive index model, while contributing towards an estimated model with a higher forecasting accuracy. Secondly, the proposed methodology will contribute towards estimating a parsimonious model in a high-dimensional setting, a crucial aspect of interpretability, even if the required domain knowledge for selecting the optimal set of predictors is unavailable.

3.4 Estimation Algorithm

In this section, we show how to efficiently find a minimiser for the problem given in Equation 8. Since the number of indices p, the vector of index coefficients $\boldsymbol{\alpha}$, as well as the set of nonparametric functions $\boldsymbol{g}$ are unknown, it is mathematically impossible to solve the above MIQP given in Equation 8 directly. Hence, we propose an iterative algorithm to solve it.

3.4.1 Initialising the Index Structure and Index Coefficients

First, we need to provide an initialisation for the index structure (i.e. number of indices p and the grouping of predictors among indices) and the index coefficients ($\boldsymbol{\alpha}$) of the model in order to start solving the MIQP given in Equation 8.

Based on several pre-experiments on the new algorithm, we propose five alternative methods for initialising the SMI Model as follows.

1. "PPR" - Projection Pursuit Regression Based Initialisation:

As discussed in Section 2.2, Projection Pursuit Regression model is a multiple index model, where each index consists of all the available predictors. Since in SMI Model we assume that there are no overlapping indices, it is impossible to use an estimated PPR model directly as a starting model for the algorithm. Thus, we follow the steps presented below to come up with a feasible initialisation for the index structure and the index coefficients.

i. Scale all the variables of the data set by dividing each variable by its standard deviation (so that it is possible to compare the estimated coefficients among predictors).

ii. Fit a PPR model and obtain estimated index coefficients. (The user can decide the number of indices to be estimated through *num_ind*; we use *num_ind* = 5 as the default value.)

iii. Calculate a threshold as $threshold = 0.1 \times \max(\text{PPR coefficients})$.

iv. Set to zero all coefficients that fall below the calculated threshold.

v. For predictors appearing in multiple indices, assign them to the index with the maximum coefficient and zero out their coefficients in other indices.

vi. After performing the above steps i-v, if any originally estimated index has all zero coefficients, it will be excluded from the model.

Now, the index structure and the index coefficients obtained through the above steps are considered to be a feasible initialisation for the proposed algorithm. Once the optimal SMI Model is obtained through the algorithm, each index coefficient will be back-transformed into the original scale of the respective predictor variable, reversing the scaling effect applied at the beginning.

2. "Additive" - Nonparametric Additive Model Based Initialisation:

As mentioned previously in Section 3.1, the nonparametric additive model is a special case of the SMI Model, where the number of indices equals the number of predictors entering indices ($p = q$) (i.e. each index contains only one predictor). Hence, it is a feasible starting point for our algorithm.

3. "Linear" - Linear Regression Based Initialisation:

In this option, we first regress the response variable on the predictors using a multiple linear regression. Then, we construct a single index (i.e. $p = 1$) using the estimated regression coefficients as the index coefficients of the predictors. Since single index model is also a special case of the SMI Model, this will be a candidate starting point.

4. "Multiple" - Selecting an Initial Model by Comparing Multiple Models:

Through our pre-experiments on the new algorithm, we identified that in some situations, the final optimised SMI Model often changes based on the initialisation provided to the algorithm. Hence, for this initialisation option, we consider several different models as initialisations, optimise the SMI Model for each of them, and pick the initial model that results in the lowest loss for the MIQP problem.

Here, users can decide on the number of models to be considered (*num_models*) as well as the the number of indices for all models (*num_ind* - same for all models). We use *num_models = 5* and *num_ind = 5* as default values.

5. "User Input" - User Specified Initialisation:

This option allows users to provide their desired initialisation for the algorithm by specifying the number of indices, the grouping of predictors among indices and the initial index coefficients. This option provides the freedom for users to utilise their domain expertise or prior knowledge in initialising the algorithm.

In all of the above initialisation options, once the estimate for $\boldsymbol{\alpha}$ is obtained, the estimated initial index coefficients for each index ($\hat{\boldsymbol{\alpha}}_j = \boldsymbol{\alpha}_{j,init}$) are scaled to have unit norm to ensure identifiability.

The characteristics and the performance of the proposed algorithm differ based on the chosen initialisation options, depending on the application scenario. Further insights into these aspects will be discussed in Section 4.

3.4.2 Estimating Nonlinear Functions

Once we have an estimate for $\boldsymbol{\alpha}$, estimating the SMI Model is equivalent to estimating a GAM as

$$y_i = \beta_0 + \sum_{j=1}^{p} g_j(\hat{h}_{ij}) + \sum_{k=1}^{d} f_k(w_{ik}) + \boldsymbol{\theta}^T \boldsymbol{u}_i + \varepsilon_i, \quad i = 1, \ldots, n,$$

where y_i is the response, and $\hat{h}_{ij} = \hat{\boldsymbol{\alpha}}_j^T \boldsymbol{x}_i$ is the estimated index.

The R packages ***mgcv*** (Wood 2011) and ***gam*** (Hastie 2023), for example, can be used to fit GAMs.

3.4.3 Updating the Index Structure and Index Coefficients

We update index coefficients $\boldsymbol{\alpha}^{new}$ through a MIQP:

$$\begin{aligned}
&\min_{\boldsymbol{\alpha}^{new}, \boldsymbol{z}^{new}} (\boldsymbol{\alpha}^{new} - \boldsymbol{\alpha}^{old})^T \boldsymbol{V}^T \boldsymbol{V} (\boldsymbol{\alpha}^{new} - \boldsymbol{\alpha}^{old}) - 2(\boldsymbol{\alpha}^{new} - \boldsymbol{\alpha}^{old})^T \boldsymbol{V}^T \boldsymbol{r} + \lambda_0 \sum_{j=1}^{p} \sum_{m=1}^{q} z_{jm}^{new} + \lambda_2 \sum_{j=1}^{p} \sum_{m=1}^{q} \alpha_{jm}^{(new)2} \\
&\text{s.t.} \quad \left|\alpha_{jm}^{new}\right| \leq M z_{jm}^{new} \quad \forall j, \forall m, \\
&\quad z_{jm}^{new} \in \{0, 1\}, \\
&\quad \sum_{j=1}^{p} z_{jm}^{new} \leq 1 \quad \forall m, \\
&\quad j = 1, \ldots, p, \quad m = 1, \ldots, q,
\end{aligned} \tag{9}$$

where $\boldsymbol{\alpha}^{old}$ is the current value of $\boldsymbol{\alpha}$, and z_{jm}^{new} are the updated set of binary variables to be estimated. $\boldsymbol{V}$ is the matrix of partial derivatives of the right hand side of Equation 6, with respect to $\boldsymbol{\alpha}_j$. The i^{th} line of $\boldsymbol{V}$ contains $\left[v_{i1}, \ldots, v_{ip}\right]$, where $v_{ij} = x_i g_j^{'}(h_{ij})$. The current residual vector, which contains $r_i = y_i - \beta_0 - \sum_{j=1}^{p} g_j(\boldsymbol{\alpha}_j^{(old)T} \boldsymbol{x}_i)$, is denoted by $\boldsymbol{r}$. It is important to note that the additional covariates w_{ik} and $\boldsymbol{u}_i$ do not step in to the process of updating $\boldsymbol{\alpha}_j$, because they are constants with respect to $\boldsymbol{\alpha}_j$, and thus they disappear from $\boldsymbol{V}$.

Similar to the explanation given by Masselot et al. (2022), the MIQP objective function in above Equation 9 ignores the Hessian (or the matrix of second derivatives of Equation 6, with respect to $\boldsymbol{\alpha}_j$), and considers only the matrix of first derivatives, which is a quasi-Newton step. The quasi-Newton Method is an alternative to the Newton's Method, avoiding the calculation of the Hessian to circumvent its computational burden (Peng 2022). Therefore, the $\boldsymbol{\alpha}$ updating step given in above Equation 9 is assured to be in a descent direction.

When $\boldsymbol{\alpha}^{new}$ is obtained, if any of the estimated individual index coefficient vectors $\boldsymbol{\alpha}_j^{new}$ contains all zeros (i.e. zero index), such indices will be dropped out from the model. Furthermore, similar to Section 3.4.1, once the new estimate $\boldsymbol{\alpha}^{new}$ is obtained, we scale each estimated index coefficient vector $\hat{\boldsymbol{\alpha}}_j = \boldsymbol{\alpha}_j^{new}$ to have unit norm.

The algorithm alternates updating the index coefficients $\boldsymbol{\alpha}$ and estimating nonlinear functions g with the updated $\boldsymbol{\alpha}$ until meeting one of the three criteria: (i) the reduction ratio of the objective (loss) function value in Equation 8, calculated between consecutive iterations, reaches a pre-specified convergence tolerance; (ii) the loss increases consecutively for three iterations; or (iii) the maximum number of iterations is reached. The selection of convergence tolerance and maximum iterations depends on the specific problem or data. In the empirical applications in

Section Section 5, we used a convergence tolerance of 0.001 and a maximum of 50 iterations, stopping at the first reached criterion.

Next, we consider changing the index structure of the model to exploit any benefits in terms of further minimising the loss function in Equation 8. As indices can be automatically reduced by dropping zero indices in each optimisation iteration, this step focuses on potential index additions to the current model. Specifically, we consider adding a new index to the current model by identifying dropped predictors. If applicable, a new index is constructed with these dropped predictors and the alternating updating process in the previous step is repeated. This increment step continues until one of these termination criteria is met: (i) the number of indices reaches q, selecting the final model as output; (ii) loss increases after the increment, selecting the previous iteration model as the final SMI model; or (iii) The solution maintains the same number of indices as the previous iteration, and the absolute difference between two successive iterations is not larger than a pre-specified tolerance, choosing the model with the smaller loss as the final SMI model in this case.

Note that, to obtain an estimated model with the best possible forecasting accuracy, it is important to select appropriate values for the non-negative penalty parameters λ_0 and λ_2. One possible way to do this is to estimate the model over a grid of possible values for λ_0 and λ_2, and then select the combination that yields the lowest loss function value. Moreover, it is also crucial to choose a suitable value for the big-M parameter, as the strength of the MIP formulation depends on the choice of a good lower bound (Bertsimas, King & Mazumder 2016). According to Hazimeh, Mazumder & Radchenko (2023), several methods have been used to select M in practice. For a description on estimating M in a linear regression setting, refer to Bertsimas, King & Mazumder (2016).

The following ***Algorithm 1*** summarises the key steps of the SMI Modelling algorithm.

Algorithm 1: SMI Modelling Algorithm

1. Initialise index structure and index coefficients $\boldsymbol{\alpha}$:
   Initialise p, predictors grouping among indices, and obtain $\boldsymbol{\alpha}^{init}$ using one of the five options in Section 3.4.1. Then scale each $\hat{\boldsymbol{\alpha}}_j = \boldsymbol{\alpha}_j^{init}$ to have unit norm.

2. Estimate nonlinear functions g_js:
   Estimate g_js using a GAM taking y_i as the response, $\hat{h}_{ij} = \hat{\boldsymbol{\alpha}}_j^T \boldsymbol{x}_i$s as predictors.

3. Update index coefficients $\boldsymbol{\alpha}$:
   Estimate updated value $\boldsymbol{\alpha}^{new}$ through the MIQP in Equation 9, and scale each $\hat{\boldsymbol{\alpha}}_j = \boldsymbol{\alpha}_j^{new}$ to have unit norm.

4. Iterate steps 2 and 3 until convergence, loss increase for three consecutive iterations, or reaching the maximum iterations.

5. Update index structure:
   Include a new index consisting of dropped predictors if applicable, and proceed to step 4. Otherwise, terminate the algorithm.

6. Iterate step 5 with increased number of indices p:
   Increase p by one in each iteration of step 5 until meeting one of the termination criteria below.

   - The number of indices in the iteration reaches q; select the final fitted model as output.
   - Loss increases after the increment; select previous iteration model as the final SMI model.
   - The solution maintains the same number of indices as the previous iteration, and the absolute difference of index coefficients between two successive iterations is not larger than a pre-specified tolerance; select the model with smaller loss as the final SMI model.

Throughout the experiments in the paper, we use $M = 10$, a convergence tolerance of 0.001, and a maximum of 50 iterations in step 4 of Algorithm 1, and a convergence tolerance of 0.001 for coefficients in estimating all the SMI Models.

4 Simulation Experiment

This section presents the results of a simple simulation experiment designed to demonstrate the performance and characteristics of the proposed SMI Modelling algorithm. Particularly, we try to investigate how the estimated SMI Model varies depending on the initialisation (as discussed in Section 3.4.1) used.

4.1 Data Generation

Generating predictor variables:

First, we generate two series each of length 1205: x_0, from a uniform distribution on the interval $[0, 1]$, and z_0, from random normal distribution $N(5, 4)$. Next, we construct lagged series up to 5^{th} lag of both x_0 and z_0. These current and lagged series of x_0 and z_0 (i.e. $\boldsymbol{x} = \{x_0, x_1, \ldots, x_5\}$, and $\boldsymbol{z} = \{z_0, z_1, \ldots, z_5\}$) were taken as predictors in the simulation experiment.

Generating response variables:

We generated two response variables y_1 and y_2, with two different index structures: single-index and 2-index, and added a random normal noise component with two different strengths as follows:

- Low noise level - $N(\mu = 0, \sigma = 0.1)$:
  $y_1 = (0.9 * x_0 + 0.6 * x_1 + 0.45 * x_3)^3 + \epsilon, \quad \epsilon \sim N(0, 0.01)$
  $y_2 = (0.9 * x_0 + 0.6 * x_1 + 0.45 * x_3)^3 + (0.35 * x_2 + 0.7 * x_5)^2 + \epsilon, \quad \epsilon \sim N(0, 0.01)$
- High noise level - $N(\mu = 0, \sigma = 0.5)$:
  $y_1 = (0.9 * x_0 + 0.6 * x_1 + 0.45 * x_3)^3 + \epsilon, \quad \epsilon \sim N(0, 0.25)$
  $y_2 = (0.9 * x_0 + 0.6 * x_1 + 0.45 * x_3)^3 + (0.35 * x_2 + 0.7 * x_5)^2 + \epsilon, \quad \epsilon \sim N(0, 0.25)$

Hence, the response y_1 is constructed using a single index consisting of the predictor variables x_0, x_1, and x_3, whereas the other response y_2 is constructed using two indices, where the first index consists of the predictors x_0, x_1, and x_3, and the second index consists of x_2 and x_5. Neither the variable x_4 nor any of the z variables were used in generating y_1 and y_2.

Once the data set is generated, the first five observations are discarded due to the missing values introduced by lagged variables, leaving a data set of 1200 observations. We use the first 1000 observations as the training set, while the remaining 200 observations are kept aside as the test set for evaluating the estimated models.

4.2 Experiment Setup

We estimated SMI Models through the proposed algorithm for each of the two response variables (the two "true models"), using three different sets of predictors as inputs. Our aim was to assess the algorithm's capability to correctly pick the relevant predictor variables (and drop the irrelevant predictors), and to estimate the correct index structure of the true model.

The three different sets of predictors considered are as follows:

1. All $\boldsymbol{x}$ variables (denoted as all $\boldsymbol{x}$);

2. All $\boldsymbol{x}$ variables and all $\boldsymbol{z}$ variables (denoted as all $\boldsymbol{x}$ + all $\boldsymbol{z}$);

3. A part of $\boldsymbol{x}$ variables (i.e. x_0, x_1 and x_2) and all $\boldsymbol{z}$ variables (denoted as some $\boldsymbol{x}$ + all $\boldsymbol{z}$).

We applied the proposed SMI Modelling algorithm with each of the above predictor combinations, for both variations of the responses concerning the noise level. Moreover, we considered each of the first four initialisation options that we discussed in Section 3.4.1, for each of the two responses.

4.3 Results

We summarise the results of the simulation experiment in Table 1. In the columns, we indicate the index structure (i.e. the number of indices and the predictor grouping among indices) estimated by the proposed algorithm under each of the four initialisation options. This is detailed for each combination explored, considering response, input predictors, and noise levels.

In the simulation experiment, we did not perform any tuning for the penalty parameters λ_0 and λ_2. Our experiments indicated that, for this simple example, different values of penalty parameters have a negligible impact on the estimated models. The default values $\lambda_0 = 1$ and $\lambda_2 = 1$ were used in estimating all the models presented in Table 1.

XQ: Please replace x and z in the Predictors column with bold $\boldsymbol{x}$ and $\boldsymbol{z}$. It's

XQ: I think it's too lengthy to explain table 1 with 14 paragraphs. Please simplify the results analysis using no more than five paragraphs. Use one paragraph each to interpret results under high and low noise levels respectively, with other two paragraphs discussing the insights, algorithm benefits and application scenarios based on the results. I've made a few changes, but you need to do some further work on it. I provided more detailed comments below.

Firstly, when the low noise level is considered, for the first true model y_1, when either "all $\boldsymbol{x}$" or "all $\boldsymbol{x}$ + all $\boldsymbol{z}$" predictors are provided, all four initialisation options enable the algorithm to output the correct index structure of the model, while selecting the correct set of predictors and dropping out the irrelevant ones. However, when only some $\boldsymbol{x}$ variables are provided, the models estimated under all four initialisations include some noise variables. This indicates that when the available predictors are insufficient to capture the data signal, the algorithm might select irrelevant variables to make up for the missing signal.

In the case of y_2 with low noise level, for both "all $\boldsymbol{x}$" and "all $\boldsymbol{x}$ + all $\boldsymbol{z}$" cases, the algorithm correctly estimates the index structure with all initialization options, except for the "Linear" option. The "Linear" option includes the correct variables in the model, however it fails to

Table 1: *Simulation experiment results.*

True Model	Predictors	PPR	Additive	Linear	Multiple
Low noise level					
y_1	all $\boldsymbol{x}$	1 index	1 index	1 index	1 index
		(x_0, x_1, x_3)	(x_0, x_1, x_3)	(x_0, x_1, x_3)	(x_0, x_1, x_3)
	all $\boldsymbol{x}$ + all $\boldsymbol{z}$	1 index	1 index	1 index	1 index
		(x_0, x_1, x_3)	(x_0, x_1, x_3)	(x_0, x_1, x_3)	(x_0, x_1, x_3)
	some $\boldsymbol{x}$ + all $\boldsymbol{z}$	1 index	3 indices	1 index	1 index
		(x_0, x_1, z_2, z_4)	(x_0, x_1) (z_4) (z_1)	(x_0, x_1, z_2, z_4)	(x_0, x_1, z_2, z_4)
y_2	all $\boldsymbol{x}$	2 indices	2 indices	1 index	2 indices
		(x_0, x_1, x_3) (x_2, x_5)	(x_0, x_1, x_3) (x_2, x_5)	$(x_0, x_1, x_2, x_3, x_5)$	(x_0, x_1, x_3) (x_2, x_5)
	all $\boldsymbol{x}$ + all $\boldsymbol{z}$	2 indices	2 indices	1 index	2 indices
		(x_0, x_1, x_3) (x_2, x_5)	(x_0, x_1, x_3) (x_2, x_5)	$(x_0, x_1, x_2, x_3, x_5)$	(x_0, x_1, x_3) (x_2, x_5)
	some $\boldsymbol{x}$ + all $\boldsymbol{z}$	3 indices	2 indices	1 index	2 indices
		(x_0, x_1, z_2) (x_2) (z_3, z_4)	(x_0, x_1, z_4) (x_2)	(x_0, x_1, x_2, z_2)	(x_0, x_1) (x_2, z_2, z_3)
High noise level					
y_1	all $\boldsymbol{x}$	1 index	2 indices	1 index	1 index
		(x_0, x_1, x_3)	(x_0, x_1) (x_3)	(x_0, x_1, x_3)	(x_0, x_1, x_3)
	all $\boldsymbol{x}$ + all $\boldsymbol{z}$	1 index	2 indices	1 index	2 indices
		(x_0, x_1, x_3)	(x_0, x_1, x_3) (z_0)	(x_0, x_1, x_3)	(x_0, x_1, x_3) (z_0)
	some $\boldsymbol{x}$ + all $\boldsymbol{z}$	3 indices	3 indices	1 index	3 indices
		(x_0, x_1) (z_1) (z_4)	(x_0, x_1) (z_1) (z_4)	(x_0, x_1, z_2, z_4)	(x_0, x_1) (z_0, z_4) (z_1)
y_2	all $\boldsymbol{x}$	3 indices	3 indices	2 indices	3 indices
		(x_0, x_1, x_3) (x_2, x_5) (x_4)	(x_0, x_1, x_3) (x_2, x_5) (x_4)	$(x_0, x_1, x_2, x_3, x_5)$ (x_4)	(x_0, x_1, x_3) (x_2, x_5) (x_4)
	all $\boldsymbol{x}$ + all $\boldsymbol{z}$	2 indices	2 indices	1 index	2 indices
		(x_0, x_1, x_3) (x_2, x_5, z_1)	(x_0, x_1, x_3) (x_2, x_5, z_1)	$(x_0, x_1, x_2, x_3, x_5, z_0)$	(x_0, x_1, x_3) (x_2, x_5)
	some $\boldsymbol{x}$ + all $\boldsymbol{z}$	3 indices	2 indices	1 index	2 indices
		(x_0, x_1, z_0, z_3) (x_2) (z_1, z_4, z_5)	$(x_0, x_1, z_0, z_1, z_3, z_4)$ (x_2)	$(x_0, x_1, x_2, z_0, z_3, z_4)$	$(x_0, x_1, z_0, z_1, z_3, z_4)$ (x_2)

identify the 2-index structure. This suggests that initializing the algorithm with a higher number of indices might be more effective than a lower number. When only some $\boldsymbol{x}$ variables are provided, similar to the case of y_1, models estimated for y_2 under all four initialisations, include noise variables.

Furthermore, we evaluated the forecasting error of the fitted models on the test set using Mean Squared Error (MSE). For y_1, in both the cases "all $\boldsymbol{x}$" and "all $\boldsymbol{x}$ + all $\boldsymbol{z}$", the models estimated using all four initialisations resulted in a test MSE of ≈ 0.01, which is the random squared error of the true model. This confirms the accuracy with which the SMI Modelling algorithm estimated the index structure for y_1. When only a part of $\boldsymbol{x}$ variables are provided, the test MSE increased to ≈ 0.23, as the estimated models contain noise variables.

For y_2, in both the cases "all $\boldsymbol{x}$" and "all $\boldsymbol{x}$ + all $\boldsymbol{z}$", all the models estimated resulted in a test MSE of ≈ 0.16. This is an interesting result as the test MSEs of the estimated models with correct index structure: "PPR", "Additive", and "Multiple", are not very different to the test MSE of an estimated model with incorrect index structure, but with correct predictors: "Linear". This

suggests that the selection of the predictor variables is more important than determining the index structure of the model.

Similar to the case of y_1, when only a part of $\boldsymbol{x}$ variables are provided, the test MSE of the estimated models for y_2 increased to ≈ 0.34, probably due to the inclusion of noise variables.

Moreover, when the above test MSE values of the models estimated for y_2 are considered, in contrast to the case of y_1, those values are higher than the random squared error of the true model. Intuitively, this might be due to the increased complexity of the model y_2 in comparison to y_1, where the total estimation error of two nonlinear link functions (corresponding to the two indices) for y_2, might be higher than the error of estimating a single nonlinear function for y_1.

XQ: The above six paragraphs should be merged into one short paragraph, comparing the index structure and forecast errors in the case of low noise level.

As expected, the accuracy with which the SMI Modelling algorithm estimates the index structure is in general lower with the high noise level, in comparison to the low noise level. In the case of y_1, when "all $\boldsymbol{x}$" variables are provided, except for "Additive" option, all the other initialisations have correctly estimated the model. When "all $\boldsymbol{x}$ + all $\boldsymbol{z}$" variables are provided, only the "PPR" and "Linear" options have estimated the correct model, whereas "Additive" and "Multiple" options have included a noise variable. Similar to the low noise level case, when only a part of $\boldsymbol{x}$ variables are provided, all four initialisations have led the algorithm to estimate models with noise variables.

Next, when y_2 is considered at high noise level, when "all $\boldsymbol{x}$" variables are provided, the estimated models using all four initialisation options have included the variable x_4 (as an additional index), which should have been omitted. When "all $\boldsymbol{x}$ + all $\boldsymbol{z}$" variables are provided, the option "Multiple" has led the algorithm to the correct model, while all the other three options have included a noise variable. Similar to the previous cases, when only a part of $\boldsymbol{x}$ variables are provided, the models estimated using all four initialisation options have included multiple noise variables, probably to cover up for the missing signal variables.

In both the cases "all $\boldsymbol{x}$" and "all $\boldsymbol{x}$ + all $\boldsymbol{z}$", all the models estimated for y_1, except for "Additive" option in "all $\boldsymbol{x}$" case, resulted in a test MSE of ≈ 0.23 (which is slightly lower than the random squared error of the true model; this probably indicates a slight level of over-fitting), irrespective of the fact that in "all $\boldsymbol{x}$ + all $\boldsymbol{z}$" case, "Additive" and "Multiple" options included a noise variable. This is an indication of the effect of the low signal-to-noise ratio in the data. The model estimated using "Additive" option in "all $\boldsymbol{x}$" case has resulted in a slightly higher test MSE

(≈ 0.27), probably due to the incorrect index structure. When only a part of $\boldsymbol{x}$ variables are provided, the test MSE has increased (≈ 0.47), which is expected.

Similar to previously discussed low noise level case, the estimated models for y_2 have resulted in higher test MSE values in comparison to the models for y_1. When "all $\boldsymbol{x}$" and "all $\boldsymbol{x}$ + all $\boldsymbol{z}$" variables are provided, all the estimated models resulted in test MSE values in the range $\approx (0.35, 0.37)$, irrespective of the different index structure and predictor choices, which is again an indication of the low signal-to-noise ratio in the data. Due to missing relevant predictors (and the inclusion of noise predictors in the model), the estimated models in "some x + all $\boldsymbol{z}$" case resulted in test MSEs ≈ 0.57.

XQ: The above four paragraphs should be merged into one short paragraph, comparing the index structure and forecast errors in the case of high noise level.

It is worth mentioning here that in real-world forecasting problems, the true data generating process (DGP) is unknown, and we do not expect an estimated model to precisely capture the true DGP. Therefore, as long as the estimated model demonstrates good forecasting accuracy, the index structure of the estimated model is less important.

Finally, the simulation study indicates that the choice of the initialisation depends on the data and application. Thus, the users are encouraged to follow a trial-and-error procedure to determine the most suitable initial model for a given application.

5 Empirical Applications

5.1 Forecasting Daily Mortality

We apply the SMI Modelling algorithm to a data set from Masselot et al. (2022), to forecast daily mortality based on heat exposure. Studying the effects of various environmental exposures such as weather related variables, pollutants and man-made environmental conditions etc. on human health, is of significant importance in environmental epidemiology. Therefore, forecasting daily deaths taking heat related variables as predictors is an interesting application.

5.1.1 Description of the Data

For this analysis, we consider daily mortality and heat exposure data for the Metropolitan Area of Montreal, Province of Quebec, Canada, from 1990 to 2014, for the months June, July, and August (i.e. summer season). The daily all-cause mortality data were obtained from the National

Institute of Public Health, Province of Quebec, while *DayMet* — a 1 km × 1 km grid data set (Thornton et al. 2021) was used to extract daily temperature and humidity data (Masselot et al. 2022).

Figure 1 shows the time plots of daily deaths during the summer for the years from 1990 to 1993. The series for each of the four years are presented separately in a faceted grid for visual clarity.

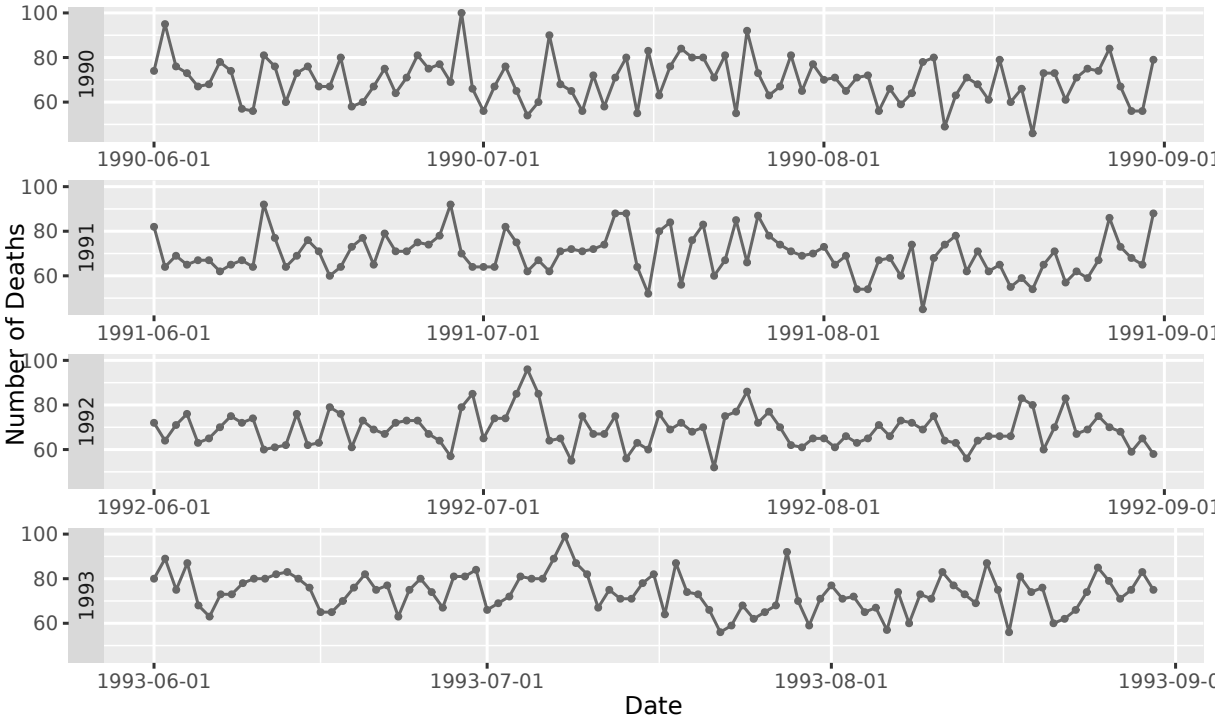

Figure 1: *Daily mortality in summer in Montreal, Canada from 1990 to 1993.*

The three main predictors considered in this empirical study are maximum temperature, minimum temperature, and vapour pressure (to represent the level of humidity). The number of daily deaths are plotted against each of these predictors in Figure 2, Figure 3, and Figure 4, respectively, where we can observe that the relationships between these predictors and the response are slightly non-linear.

5.1.2 Predictors Considered

1) Current maximum/minimum temperatures and lags:

In addition to current maximum and minimum temperatures, the temperature measurements up to 14 days prior (i.e. 0^{th} to 14^{th} lag) are considered as predictors in the forecasting model. This accounts for the cumulative impact of both current and recent past temperatures on a person's heat exposure.

2) Current vapour pressure and lags:

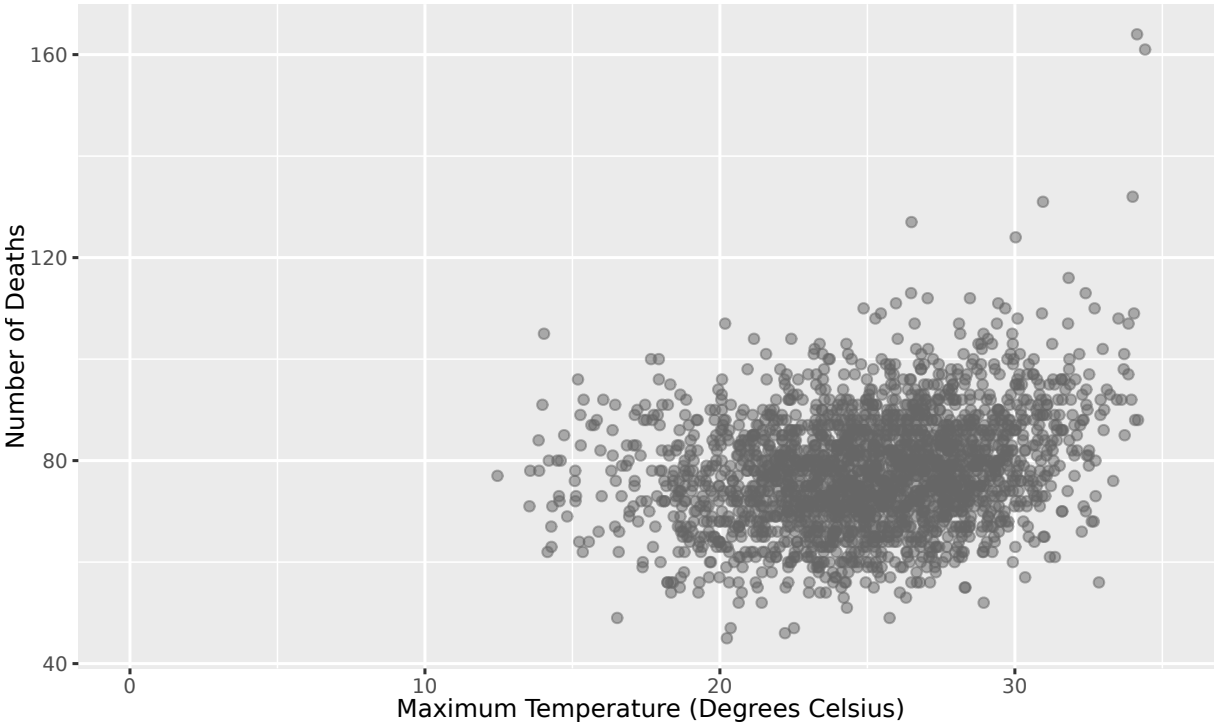

Figure 2: *Daily mortality in summer (from 1990 to 2014) plotted against maximum temperature.*

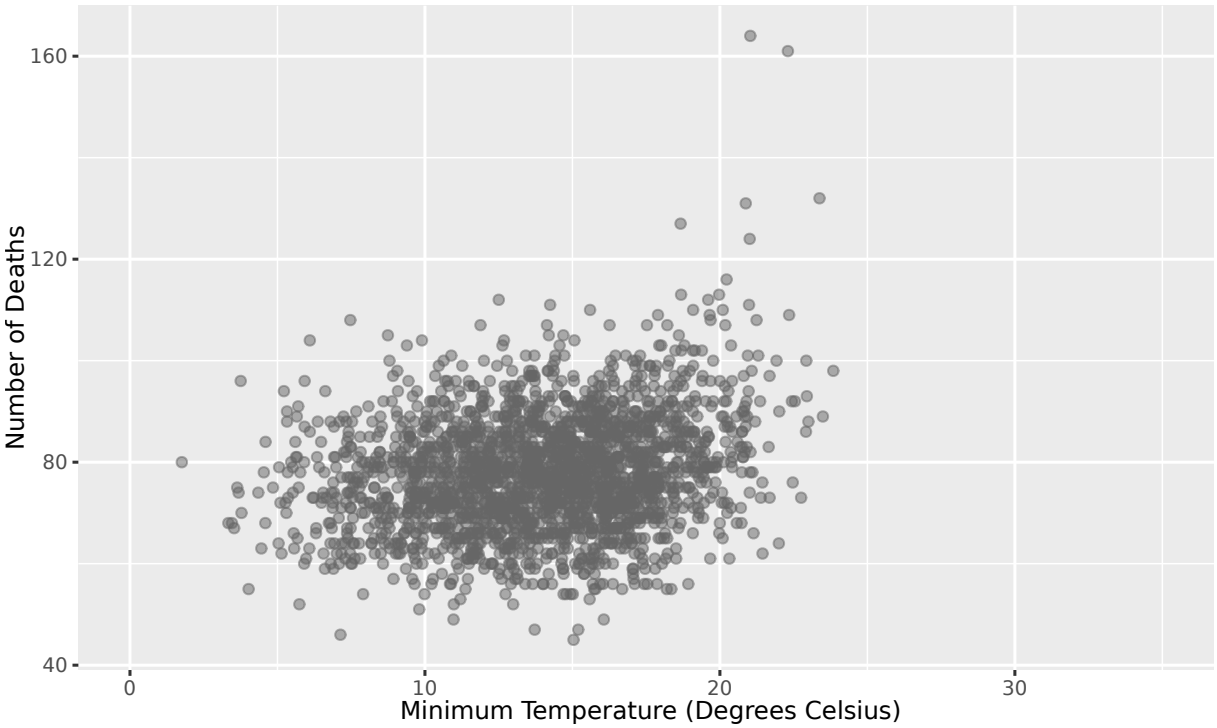

Figure 3: *Daily mortality in summer (from 1990 to 2014) plotted against minimum temperature.*

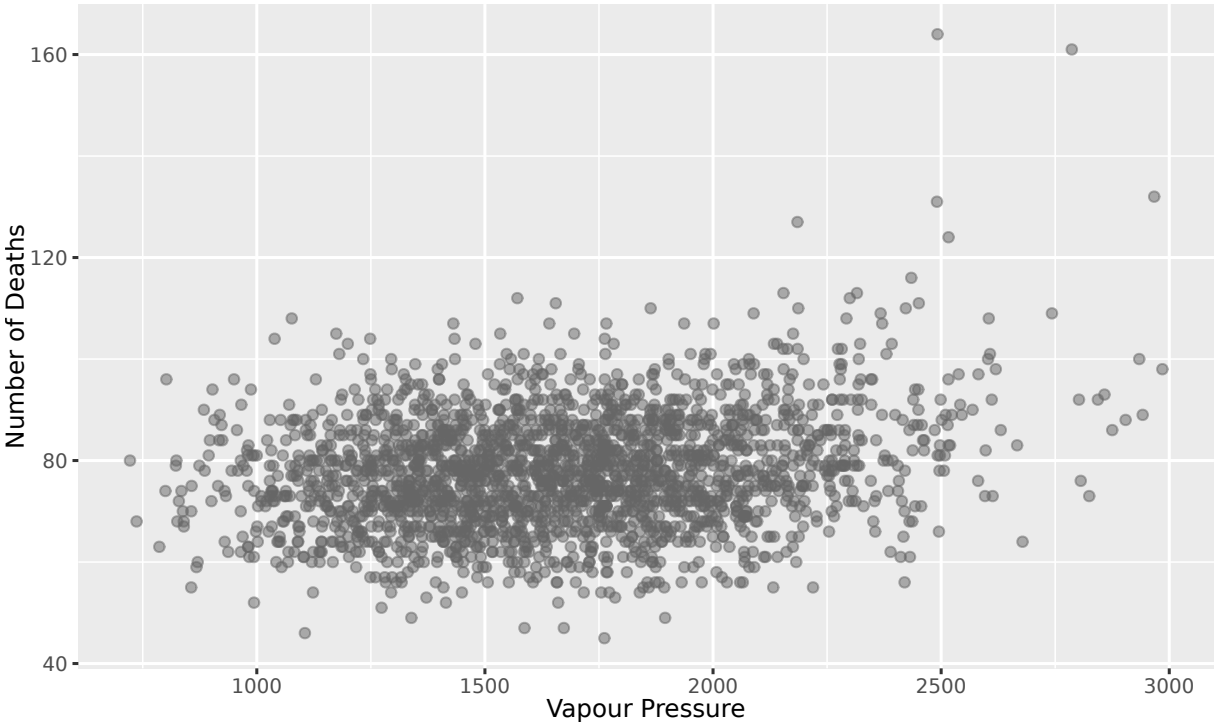

Figure 4: *Daily mortality in summer (from 1990 to 2014) plotted against vapour pressure.*

Similar to temperature variables, the current value and 14 lags of vapour pressure are considered as predictors, as a proxy to the level of humidity.

3) Calendar effects:

Finally, a couple of calendar variables; *day of the season (DOS)* and *Year*, are incorporated into the model to capture annual trend and seasonality, and also to control the autocorrelation in residuals, which is a common practice in environmental epidemiology (Masselot et al. 2022).

5.1.3 Modelling Framework

Maximum temperature lags, minimum temperature lags, and vapour pressure lags are considered as predictors entering indices. The two calendar variables, *DOS* and *Year*, are included into the model as separate nonparametric components that do not enter any of the indices.

Hence, the relevant SMI Model can be written as

$$\textbf{Deaths} = \beta_0 + \sum_{j=1}^{p} g_j(\boldsymbol{X}\boldsymbol{\alpha}_j) + f_1(\textbf{DOS}) + f_2(\textbf{Year}) + \boldsymbol{\varepsilon}, \tag{10}$$

where

- **Deaths** is the vector containing daily deaths observations;

- β_0 is the model intercept;
- p is the unknown number of indices that need to be estimated through the algorithm;
- $\boldsymbol{X}$ is the matrix containing the predictor variables that are entering indices (i.e. maximum temperature lags, minimum temperature lags, and vapour pressure lags);
- $\boldsymbol{\alpha}_j, j = 1, \ldots, p$ are the index coefficient vectors, each with a length equal to the number of predictors entering indices ($q = 45$);
- $g_j, j = 1, \ldots, p$, f_1, and f_2 are unknown nonparametric functions; and
- ε is the error term.

The data from 1990 to 2012 are used as the training set to estimate the model, while the data of year 2014 are separated to be the test set for evaluating forecasting performance. The data from the three summer months of year 2013 are kept aside as a validation set, which is used to estimate benchmark models for comparison purposes.

Then we apply the proposed SMI Modelling algorithm to the training set to estimate the model. Finally, the forecasting accuracy on the test set is evaluated using MSE and Mean Absolute Error (MAE).

5.1.4 Results

We estimated SMI Models for the mortality data using three different initialisation options: "PPR", "Additive" and "Linear", for comparison purposes. Through our pre-experiments on the new algorithm, we identified that the "PPR" initialisation option has a higher probability of better performance, whereas "Additive" (i.e. Additive Model) and "Linear" (i.e. Single Index Model) are two special cases of the SMI Model. We did not consider "Multiple" and "User Input" initialisations here as both of these two options require user specific inputs to some extend. Further, we tuned the penalty parameters λ_0 and λ_2, over ranges of integers from 1 to 12, and 0 to 12 respectively, through a greedy search based on in-sample MSE. Here, a greedy search is used instead of a grid search to reduce computational time.

The penalty parameter combination $(\lambda_0 = 12, \lambda_2 = 0)$ was selected for the model fitted with "PPR" initialisation. The estimated model, ***SMI Model (12, 0) - PPR*** , resulted in five indices wihout dropping any of the index variables. The optimal penalty parameter combination for the model initiated with "Additive" was $(\lambda_0 = 1, \lambda_2 = 0)$, resulting in the ***SMI Model (1, 0) - Additive*** , equivalent to a nonparametric additive model (no index variables or indices were

dropped). The model estimated with "Linear" initialisation selected $(\lambda_0 = 12, \lambda_2 = 5)$ (***SMI Model (12, 5) - Linear***), and resulted in two indices, without dropping any of the index variables.

We evaluated forecasting errors of the estimated models using two subsets of the original test set:

1. ***Test Set 1:*** original test set spanning 3 months (June, July and August 2014); and
2. ***Test Set 2:*** a test set covering 1 month (June 2014).

Note that in this application, we assumed that the future values of the maximum/minimum temperatures and vapour pressure are known to use in the forecasting model.

The MSE and MAE values for the estimated SMI Models on two different test sets are presented in Table 2. We observe that the SMI Model estimated with "PPR" initialisation, ***SMI Model (12, 0) - PPR*** , shows the best forecasting performance on both test sets, compared to the other two estimated SMI models.

Furthermore, we present the forecasting errors of three benchmark models in Table 2 for comparison with the estimated SMI Models. The first benchmark is a nonparametric additive model formulated through backward elimination, as proposed by Fan & Hyndman (2012) (Section 2.1.1). Next, a GAIM (Section 2.2.2) is also presented. In the case of GAIM, maximum temperature lags, minimum temperature lags, and vapour pressure lags are categorised into three groups, where an index estimated for each group. Finally, we present the forecasting errors of a PPR model. The number of indices in the PPR model was taken as 3, matching the number of indices estimated by the GAIM.

Table 2: *Daily mortality forecasting - Out-of-sample point forecast results.*

			Test Set 1		Test Set 2	
Model	Predictors	Indices	MSE	MAE	MSE	MAE
smimodel(12, 0) - PPR	47	5	**80.334**	**6.841**	**99.926**	**7.643**
smimodel(1, 0) - Additive	47	45	151.408	9.816	190.880	11.107
smimodel(12, 5) - Linear	47	2	164.629	10.153	207.040	11.141
Backward Elimination	36	NA	148.387	9.808	162.608	10.034
GAIM	47	3	85.145	7.257	103.494	8.480
PPR	47	3	82.877	7.202	104.217	8.404

Table 2 shows that ***SMI Model (12, 0) - PPR*** outperforms all three benchmark models in terms of forecasting accuracy, for both *Test Set 1* and *Test Set 2*. However, the SMI Models estimated using "Additive" or "Linear" initialisations have inferior forecasting performance compared to

all benchmark models considered. The actual number of deaths and the predicted values from the ***SMI Model (12, 0) - PPR*** and benchmark models on *Test Set 2* are plotted in Figure 5 for further comparison.

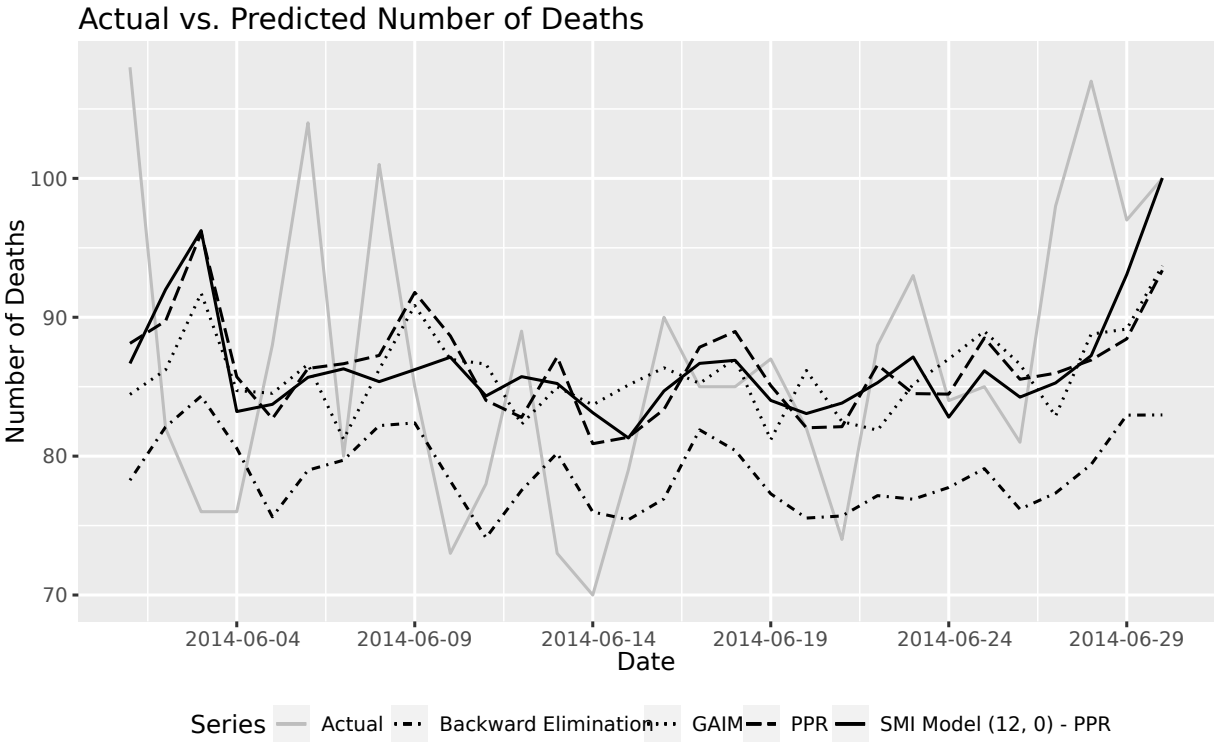

Figure 5: *Actual number of deaths vs. predicted number of deaths from "SMI Model (12, 0) - PPR" and benchmark models for Test Set 2.*

5.2 Forecasting Daily Solar Intensity

Next, we utilise the SMI Modelling algorithm to forecast daily solar intensity, using other weather conditions. As reported by Energy Institute (2023), renewable energy (excluding hydroelectricity) contributed to 7.5% of the world's primary energy consumption in 2022. Solar and wind power saw a combined capacity addition of 266 GW, with solar energy accounting for 72% of the increase. Given this, accurate forecasting of solar power generation, closely linked to solar intensity, is crucial for effective power system planning and management.

5.2.1 Description of the Data

We use solar intensity and other weather variables measured at a Davis weather station in Amherst, Massachusetts, obtained from the *UMass Trace Repository* (University of Massachusetts 2023). The data was recorded at every five minutes, from 21th February 2006 to 27th February 2013, using sensors for measuring temperature, wind chill, humidity, dew point, wind speed, wind direction, rain, pressure, solar intensity, and UV.

However, the data contained missing entries recorded as "-100000", which we removed from the data set. Moreover, for this analysis, we converted the five minutes data to daily data by averaging each variable over days.

Figure 6 shows the time plot of daily solar intensity for the entire period, which clearly depicts the annual seasonality in the data. As observed in Figure 6, there are days for which the observations were missing. We excluded those days from the analysis, and used only the days for which the data are available.

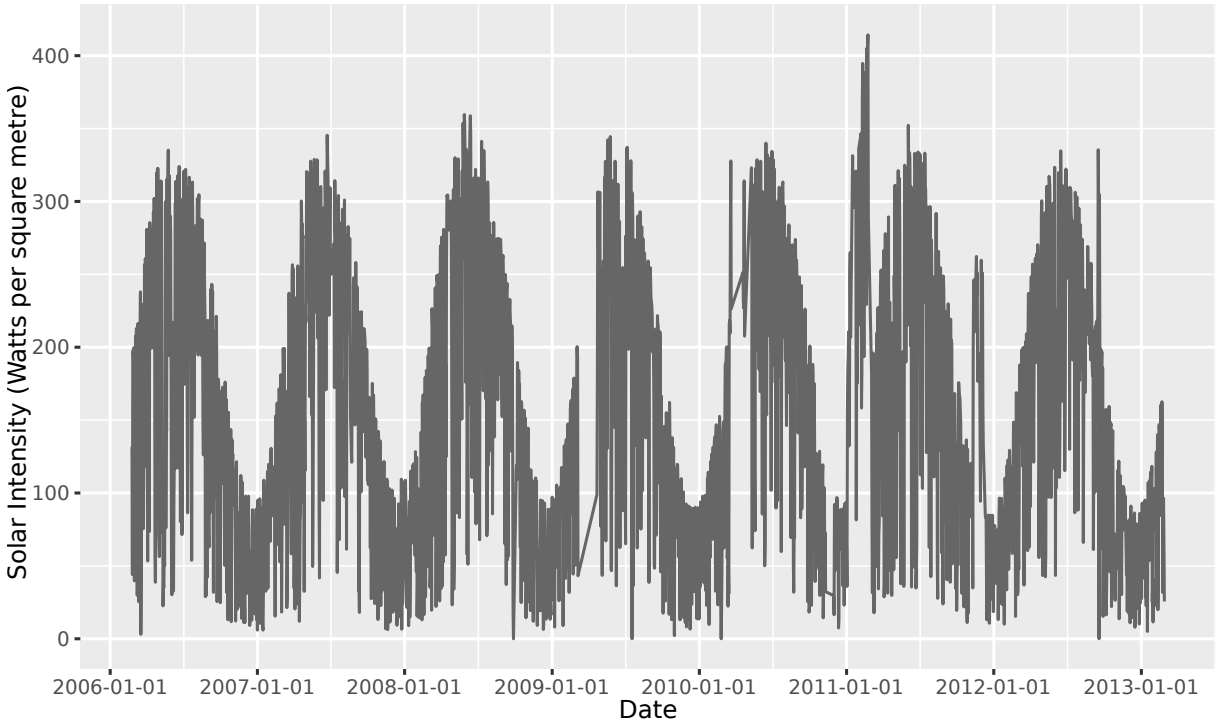

Figure 6: *Daily solar intensity in Amherst, Massachusetts - from February 2006 to February 2013.*

The variables temperature, dew point, wind, rain and humidity were considered to be the main set of predictors in the model. The daily solar intensity is plotted against each of these predictors in Figure 7, where we can observe that the relationships between these predictors and the response are non-linear.

5.2.2 Predictors Considered

1) Solar intensity lags:

Three lags of the daily solar intensity itself are used as predictors to incorporate the serial correlations presented in the data into the modelling process. Intuitively, the solar intensity of a particular day would have a relationship to the solar intensity of adjacent days.

2) Current weather variables and lags:

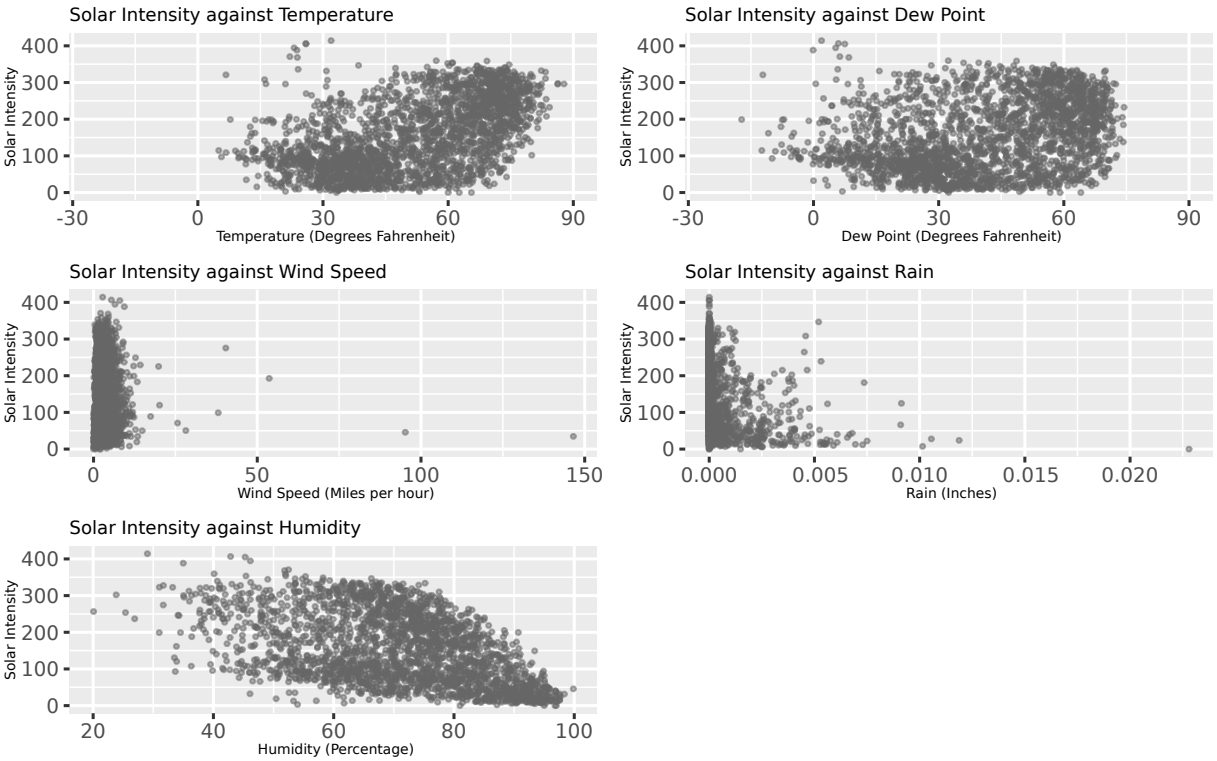

Figure 7: *Daily solar intensity against other weather variables.*

In addition to current temperature, dew point, wind speed, rain and humidity, the measurements of three previous days (i.e. 0^{th} to 3^{rd} lag) for each of the these weather variables are also included as predictors in the forecasting model.

3) Calendar effects:

Finally, a couple of calendar variables; *Month* (12 months of the year) and *Season* (the four seasons: Spring, Summer, Autumn and Winter), are incorporated into the model to capture annual seasonality, and control for autocorrelation in residuals.

5.2.3 Modelling Framework

The lags of solar intensity, and the lags of weather variables are considered as predictors that are entering indices. The two calendar variables, *Month* and *Season*, are included into the model as linear (categorical) predictor variables.

Hence, the relevant SMI Model can be written as

$$\textbf{Solar} = \beta_0 + \sum_{j=1}^{p} g_j(\boldsymbol{X}\boldsymbol{\alpha}_j) + \theta_1 \textbf{Month} + \theta_2 \textbf{Season} + \varepsilon, \tag{11}$$

where

- **Solar** is the vector containing daily observations of solar intensity;
- β_0 is the model intercept;
- p is the unknown number of indices that will be estimated through the algorithm;
- $\boldsymbol{X}$ is the matrix containing the predictor variables that are entering indices (i.e. solar intensity, temperature, dew point, wind speed, rain and humidity lags);
- $\boldsymbol{\alpha}_j, j = 1, \ldots, p$ are the index coefficient vectors, each of lenth equal to the number of predictors entering indices ($q = 23$);
- $g_j, j = 1, \ldots, p$ are unknown nonparametric functions;
- θ_1 and θ_2 are the two coefficients corresponding to the two linear predictor variables; and
- $\boldsymbol{\varepsilon}$ is the error term.

The data from February 2006 to October 2012 are used as the training set to estimate the model, while the data of the months January and February 2013 are separated to be the test set to evaluate the forecasting performance. The data from the months November and December 2013 are kept aside as a validation set, which is required to estimate some of the benchmark models for comparison.

Then we apply the proposed SMI Modelling algorithm to the training set to estimate the model, and the forecasting accuracy on the test set is evaluated using MSE and MAE.

5.2.4 Results

Similar to the previous empirical application, we estimated SMI Models for the solar intensity data using three different initialisation options: "PPR", "Additive" and "Linear", for comparison purposes. We also tuned the penalty parameters λ_0 and λ_2, over ranges of integers from 1 to 12, and 0 to 12 respectively.

The penalty parameter combination $(\lambda_0 = 12, \lambda_2 = 0)$ was selected for the model fitted with "PPR" initialisation. The estimated model, ***SMI Model (12, 0) - PPR*** , resulted in five indices without dropping any of the index variables. The optimal penalty parameter combination for the model estimated taking "Additive" model as the starting point was $(\lambda_0 = 1, \lambda_2 = 0)$. The estimated SMI Model did not drop any index variables or indices, and thus the final model, ***SMI Model (1, 0) - Additive*** , is equivalent to a nonparametric additive model. The model estimated with "Linear" initialisation also selected $(\lambda_0 = 1, \lambda_2 = 0)$. Unlike the above models, this SMI

Model dropped all index variables and resulted in null indices, and hence, the final model, ***SMI Model (1, 0) - Linear*** , is just a linear model with the two linear variables *Month* and *Season*. Notice that all three estimated SMI Models have $\lambda_2 = 0$, indicating that all three models have omitted the ℓ_2-penalty in the estimation process.

Note that similar to the previous application of heat related mortality forecasting, we assumed that the future values of the weather variables are known to use in the forecasting model.

Table 3 presents the MSE and MAE values for the estimated SMI Models on the test set. The results indicate that the SMI Model estimated with "Additive" initialisation, ***SMI Model (1, 0) - Additive***, shows the best forecasting performance among the three estimated SMI Models.

Similar to Section 5.1, we also present forecasting errors of three benchmark models in Table 3, to compare with the estimated SMI Models. Here, the GAIM is fitted by grouping the lags of each weather variable into a different group, resulting in six indices. The number of indices of the PPR model was taken as six, matching the number of indices estimated by the GAIM. Note that here, the two categorical calendar variables were excluded when estimating the PPR model.

Table 3: *Daily solar intensity forecasting - Out-of-sample point forecast results.*

			Test Set	
Model	Predictors	Indices	MSE	MAE
SMI Model (12, 0) - PPR	25	5	1745.030	33.246
SMI Model (1, 0) - Additive	25	23	1112.181	26.975
SMI Model (1, 0) - Linear	2	0	2009.847	35.346
Backward Elimination	16	NA	911.570	25.035
GAIM	25	6	2203.530	37.788
PPR	23	6	**796.779**	**22.455**

According to Table 3, the forecasting errors of ***SMI Model (1, 0) - Additive*** is lower than the GAIM. However, the ***SMI Model (1, 0) - Additive*** is unable to outperform both the semi-parametric additive model with backward elimination and the PPR model, where in this case, the estimated PPR model has resulted in the best forecasting accuracy.

Here, it is worth considering the differences between the SMI Model and the benchmark models that show superior forecasting performance. The method proposed by Fan & Hyndman (2012) formulates a semi-parametric additive model using a backward elimination of predictors. When estimating a PPR model, both the number of indices and the predictors within each index (each index includes all provided predictors that are entering indices) are pre-determined. In contrast, the SMI Model takes a more general and objective approach, where the number of indices as well as predictors within each index are automatically determined through the proposed

algorithm. Thus, the SMI Model faces a more challenging estimation task due to the limited prior information provided.

The actual solar intensity and the predicted values from the SMI Model (1, 0) - Additive and benchmark models are plotted in Figure 8 for further comparison.

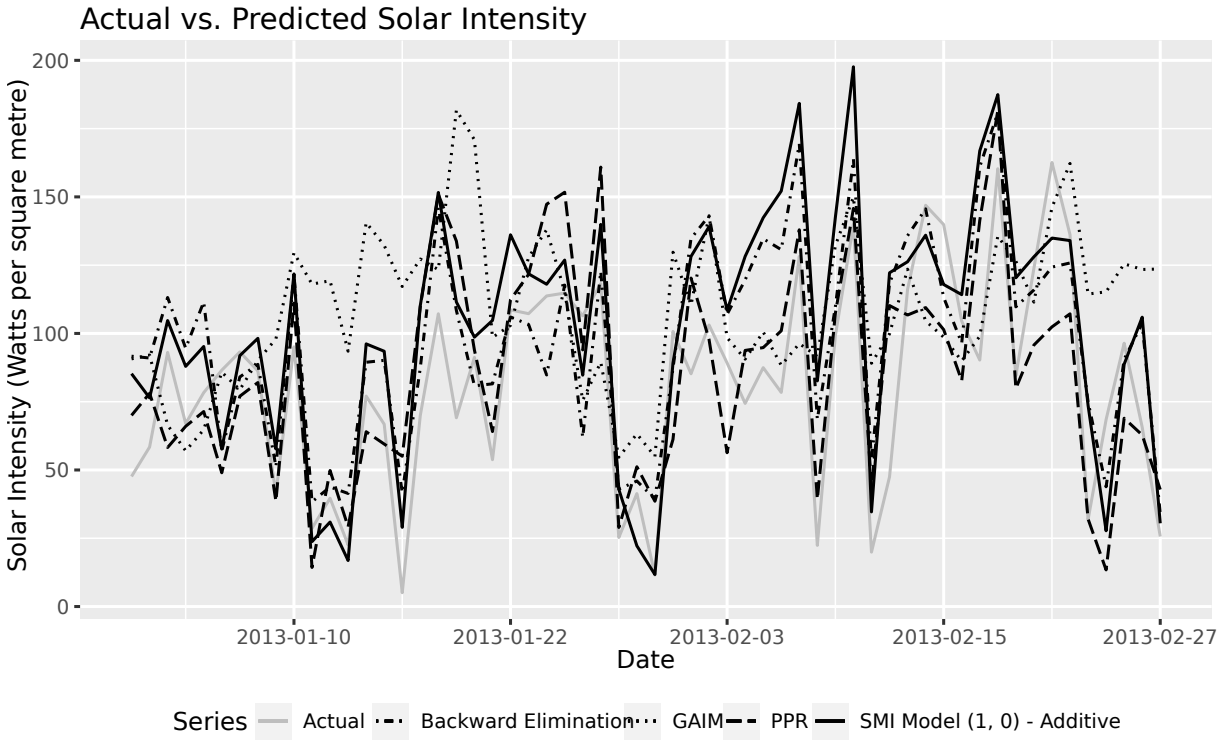

Figure 8: *Actual solar intensity vs. predicted solar intensity from "SMI Model (1, 0) - Additive" and benchmark models.*

In summary, the two empirical applications presented above highlight the challenge of finding a universally applicable initialisation option for the SMI Model across various applications. As mentioned in Section 4, we encourage users to follow a trial-and-error procedure to identify the most effective initialisation option for their specific application.

The two empirical applications were performed using R statistical software (R Core Team 2023), and the Rstudio integrated development environment (IDE, Posit team 2024). We used the commercial MIP solver ***Gurobi*** (Gurobi Optimization, LLC 2023) to solve the MIQPs related to the proposed SGAIM algorithm, through the ***Gurobi plug-in*** (ROI.plugin.gurobi, Schwendinger 2023) available from the ***R Optimization Infrastructure*** (ROI, Hornik et al. 2023; Theußl, Schwendinger & Hornik 2020) package. Furthermore, the GAMs were fitted using the R package ***mgcv*** (v1.9.1, Wood 2011).

6 Conclusions and Further Research

In this paper, we presented a novel algorithm for estimating a nonparametric additive index model with optimal predictor selection, which we refer to as Sparse Multiple Index (SMI) Model. The SMI Modelling algorithm is an iterative procedure that is developed based on mixed integer programming to solve an ℓ_0-regularised nonlinear least squares optimisation problem with linear constraints.

The proposed SMI Modelling algorithm has a number of key features: 1) It performs automatic selection of both the number of indices and the predictor grouping when estimating the nonparametric additive index model. Users need to input the set of predictors entering indices and a starting model (index structure and a set of index coefficients) to initiate the algorithm. 2) It performs automatic variable selection, which is particularly beneficial in high-dimensional settings. This feature contributes to an objective and principled estimation, reducing subjectivity across different users. 3) It is capable of estimating a wide spectrum of models, from single index models (one index) to additive models (number of indices equals the number of predictors entering indices). Hence, the SMI Modelling algorithm is a more general estimation tool for nonparametric additive models. 4) It provides the flexibility to include separate non-liner and linear predictors in the model that are not entering any indices, allowing the estimation of semi-parametric additive models.

Due to the limited input information provided to the algorithm, the estimation of a SMI Model is a challenging problem. We demonstrated the performance of the proposed algorithm through a simple simulation and two empirical applications. Since we observed that the final estimated model changes with the choosen initialisation, one limitation of the proposed algorithm is the difficulty of specifying an initialisation that works in general. Hence, an interesting future research problem would be to explore the potential for determining a generalised initialisation for the SMI Modelling algorithm that will work across various applications.

Moreover, we admit that the empirical examples presented in the paper may not be diverse enough to draw definitive conclusions about the unique strengths or weaknesses of the proposed algorithm. This study should be viewed as an attempt to develop a more objective methodology for variable selection and model estimation in the broader class of nonparametric additive models for forecasting. An important future research problem is therefore, to assess the performance of the proposed SMI Modelling algorithm across various data sets with diverse properties, identifying scenarios where it outperforms other benchmark methods.

Furthermore, the MIQP in the algorithm is somewhat analogous to the *best subset selection* method frequently used in least squares problems. Thus, another limitation of the proposed algorithm is the increase in computational time as the number of predictors and number of indices increase. Therefore, it would be an interesting research to obtain further insights regarding the algorithm to see what improvements can be made to the algorithm design to reduce the computational cost in a high-dimensional context.

Acknowledgements

We thank Professor Louise Ryan for joining the discussions during the initial stage of the project, and for her valuable comments and feedback on this research work.

Furthermore, this research is partially supported by the Monash eResearch Centre through the use of the MonARCH (Monash Advanced Research Computing Hybrid) HPC Cluster.

References

Bakker, M & F Schaars (2019). Solving groundwater flow problems with time series analysis: you may not even need another model. *Groundwater* **57**(6), 826–833.

Bellman, R (1957). *Dynamic Programming*. Princeton, New Jersey: Princeton University Press.

Bertsimas, D, A King & R Mazumder (2016). Best subset selection via a modern optimization lens. *Annals of Statistics* **44**(2), 813–852.

Boggs, PT & JW Tolle (1995). Sequential Quadratic Programming. *Acta Numerica* **4**, 1–51.

Boyd, SP & L Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Breiman, L (1995). Better Subset Regression Using the Nonnegative Garrote. *Technometrics* **37**(4), 373–384.

Efron, B, T Hastie, I Johnstone & R Tibshirani (2004). Least Angle Regression. *Annals of Statistics* **32**(2), 407–499.

Energy Institute (2023). *Statistical Review of World Energy*. https://www.energyinst.org/__data/assets/pdf_file/0004/1055542/EI_Stat_Review_PDF_single_3.pdf.

Fan, J & R Li (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association* **96**(456), 1348–1360.

Fan, S & RJ Hyndman (2012). Short-term load forecasting based on a semi-parametric additive model. *IEEE Transactions on Power Systems* **27**(1), 134–141.

Frank, IE & JH Friedman (1993). A Statistical View of Some Chemometrics Regression Tools. *Technometrics* **35**(2), 109–135.

Friedman, JH & JW Tukey (1974). A Projection Pursuit Algorithm for Exploratory Data Analysis. *IEEE Transactions on Computers* **C-23**(9), 881–890.

Friedman, JH & W Stuetzle (1981). Projection Pursuit Regression. *Journal of American Statistical Association* **76**(376), 817–823.

Guo, J, M Tang, M Tian & K Zhu (2013). Variable selection in high-dimensional partially linear additive models for composite quantile regression. *Computational Statistics and Data Analysis* **65**, 56–67.

Gurobi Optimization, LLC (2023). *Gurobi Optimizer Reference Manual*. https://www.gurobi.com.

Härdle, W, P Hall & H Ichimura (1993). Optimal Smoothing in Single-Index Models. *Annals of Statistics* **21**(1), 157–178.

Hastie, T (2023). *gam: Generalized Additive Models*. R package version 1.22-2. https://CRAN.R-project.org/package=gam.

Hazimeh, H & R Mazumder (2020). Fast Best Subset Selection: Coordinate Descent and Local Combinatorial Optimization Algorithms. *Operations Research* **68**(5), 1517–1537.

Hazimeh, H, R Mazumder & P Radchenko (2023). Grouped variable selection with discrete optimization: Computational and statistical perspectives. *Annals of Statistics* **51**(1), 1–32.

Ho, CC, LJ Chen & JS Hwang (2020). Estimating ground-level PM2.5 levels in Taiwan using data from air quality monitoring stations and high coverage of microsensors. *Environmental Pollution* **264**, 114810.

Hornik, K, D Meyer, F Schwendinger & S Theussl (2023). *ROI: R Optimization Infrastructure*. R package version 1.0-1. https://CRAN.R-project.org/package=ROI.

Huang, J, JL Horowitz & F Wei (2010). Variable Selection in Nonparametric Additive Models. *Annals of Statistics* **38**(4), 2282–2313.

Huber, PJ (1985). Projection Pursuit. *Annals of Statistics* **13**(2), 435–475.

Hyndman, RJ & S Fan (2010). Density forecasting for long-term peak electricity demand. *IEEE Transactions on Power Systems* **25**(2), 1142–1153.

Ibrahim, S, R Mazumder, P Radchenko & E Ben-David (2022). "Predicting Census Survey Response Rates via Interpretable Nonparametric Additive Models with Structured Interactions". https://arxiv.org/abs/2108.11328.

Konzen, E & FA Ziegelmann (2016). LASSO-type penalties for covariate selection and forecasting in time series. *Journal of Forecasting* **35**(7), 592–612.

Kruskal, JB (1969). "Toward a practical method which helps uncover the structure of a set of multivariate observations by finding the linear transformation which optimizes a new

"index of condensation"". In: *Statistical Computation*. Ed. by Roy C. Milton and John A. Nelder. Academic Press, pp.427–440.

Land, AH & AG Doig (1960). An Automatic Method of Solving Discrete Programming Problems. *Econometrica* **28**(3), 497–520.

Lian, H (2012). Variable selection in high-dimensional partly linear additive models. *Journal of Nonparametric Statistics* **24**(4), 825–839.

Little, JDC, KG Murty, DW Sweeney & C Karel (1963). An algorithm for the traveling salesman problem. *Operations Research* **11**(6), 972–989.

Liu, X, L Wang & H Liang (2011). Estimation and Variable Selection for Semiparametric Additive Partial Linear Models (SS-09-140). *Statistica Sinica* **21**(3), 1225–1248.

Masselot, P, F Chebana, C Campagna, É Lavigne, TBMJ Ouarda & P Gosselin (2022). Constrained groupwise additive index models. *Biostatistics* **00**(00), 1–19.

Mazumder, R, P Radchenko & A Dedieuc (2022). Subset Selection with Shrinkage: Sparse Linear Modeling When the SNR Is Low. *Operations Research*, 1–19.

Nocedal, J & SJ Wright (2006). *Numerical Optimization*. 2nd. Springer Series in Operations Research and Financial Engineering. Springer New York, NY.

Park, H & F Sakaori (2013). Lag weighted lasso for time series model. *Computational Statistics* **28**(2), 493–504.

Peng, RD (2022). *Advanced Statistical Computing*. https://bookdown.org/rdpeng/advstatcomp/. Accessed: 2023-5-19.

Peterson, TJ & AW Western (2014). Nonlinear time-series modeling of unconfined groundwater head. *Water Resources Research* **50**(10), 8330–8355.

Posit team (2024). *RStudio: Integrated Development Environment for R*. Posit Software, PBC. Boston, MA. http://www.posit.co/.

R Core Team (2023). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/.

Radchenko, P (2015). High dimensional single index models. *Journal of Multivariate Analysis* **139**, 266–282.

Rajaee, T, H Ebrahimi & V Nourani (2019). A review of the artificial intelligence methods in groundwater level modeling. *Journal of Hydrology* **572**, 336–351.

Ravindra, K, P Rattan, S Mor & AN Aggarwal (2019). Generalized additive models: Building evidence of air pollution, climate change and human health. *Environment International* **132**, 104987.

Schwendinger, F (2023). *ROI.plugin.gurobi: 'Gurobi' Plug-in for the 'R' Optimization Infrastructure*. R package version 0.4-0. http://r-forge.r-project.org/projects/roi.

Simon, N, J Friedman, T Hastie & R Tibshirani (2013). A Sparse-Group Lasso. *Journal of Computational and Graphical Statistics* **22**(2), 231–245.

Stoker, TM (1986). Consistent Estimation of Scaled Coefficients. *Econometrica* **54**(6), 1461–1481.

Stone, CJ (1982). Optimal global rates of convergence for nonparametric regression. *Annals of Statistics* **10**(4), 1040–1053.

Theußl, S, F Schwendinger & K Hornik (2020). ROI: An Extensible R Optimization Infrastructure. *Journal of Statistical Software* **94**, 1–64.

Thornton, PE, R Shrestha, M Thornton, SC Kao, Y Wei & BE Wilson (2021). Gridded daily weather data for North America with comprehensive uncertainty quantification. *Scientific Data* **8**(1), 190.

Tibshirani, R (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society (Series B)* **58**(1), 267–288.

Tibshirani, R & X Suo (2016). An Ordered Lasso and Sparse Time-Lagged Regression. *Technometrics* **58**(4), 415–423.

University of Massachusetts (2023). *The UMass trace repository*. https://traces.cs.umass.edu/index.php/Sensors/Sensors.

Wang, H, L Guodong & CL Tsai (2007). Regression Coefficient and Autoregressive Order Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society (Series B)* **69**(1), 63–78.

Wang, L, L Xue, A Qu & H Liang (2014). Estimation and model selection in generalized additive partial linear models for correlated data with diverging number of covariates. *Annals of Statistics* **42**(2), 592–624.

Wang, T, P Xu & L Zhu (2015). Variable selection and estimation for semi-parametric multiple-index models. *Bernoulli* **21**(1), 242–275.

Wang, T, J Zhang, H Liang & L Zhu (2015). Estimation of a Groupwise Additive Multiple-Index Model and its Applications. *Statistica Sinica* **25**, 551–566.

Wood, SN (2017). *Generalized Additive Models: An Introduction with R*. 2nd. Chapman & Hall/CRC.

Wood, SN (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society (Series B)* **73**(1), 3–36.

Yuan, M & Y Lin (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society (Series B)* **68**(1), 49–67.

Zhang, CH (2010). Nearly Unbiased Variable Selection under Minimax Concave Penalty. *Annals of Statistics* **38**(2), 894–942.

Zhang, X, L Liang, X Tang & HY Shum (2008). L1 regularized projection pursuit for additive model learning. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp.1–8.

Zou, H (2006). The Adaptive Lasso and its Oracle Properties. *Journal of the American Statistical Association* **101**(476), 1418–1429.