IISE Transactions

Taylor & Francis
Taylor & Francis Group

ISSN: 2472-5854 (Print) 2472-5862 (Online) Journal homepage: http://www.tandfonline.com/loi/uiie21

A sparse partitioned-regression model for nonlinear system–environment interactions

Shuluo Ning, Eunshin Byon, Teresa Wu & Jing Li

To cite this article: Shuluo Ning, Eunshin Byon, Teresa Wu & Jing Li (2017): A sparse partitioned-regression model for nonlinear system–environment interactions, IISE Transactions, DOI: 10.1080/24725854.2017.1299955

To link to this article: http://dx.doi.org/10.1080/24725854.2017.1299955

IISE TRANSACTIONS
2017, VOL. 49, NO. 8, 1–13
http://dx.doi.org/10.1080/24725854.2017.1299955

Taylor & Francis
Taylor & Francis Group

A sparse partitioned-regression model for nonlinear system–environment interactions

Shuluo Ning[a], Eunshin Byon[b], Teresa Wu[a] and Jing Li[a]

[a]Industrial Engineering, School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA; [b]Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, USA

ABSTRACT

This article focuses on the modeling of nonlinear interactions between the design and operational variables of a system and the multivariate outside environment in predicting the system's performance. We propose a Sparse Partitioned-Regression (SPR) model that automatically searches for a partition of the environmental variables and fits a sparse regression within each subdivision of the partition, in order to fulfill an optimal criterion. Two optimal criteria are proposed, a penalized and a held-out criterion. We study the theoretical properties of SPR by deriving oracle inequalities to quantify the risks of the penalized and held-out criteria in both prediction and classification problems. An efficient recursive partition algorithm is developed for model estimation. Extensive simulation experiments are conducted to demonstrate the better performance of SPR compared with competing methods. Finally, we present an application of using building design and operational variables, outdoor environmental variables, and their interactions to predict energy consumption based on the Department of Energy's *EnergyPlus* data sets. SPR produces a high level of prediction accuracy. The result of the application also provides insights into the design, operation, and management of energy-efficient buildings.

ARTICLE HISTORY
Received 27 March 2016
Accepted 14 January 2017

KEYWORDS
Sparse model; regression; classification; building energy management

1. Introduction

Modeling the relationship between the design and operational variables of a system and the system's performance is a primary interest in various domains. When the system is functioning in different environments, this relationship is likely to vary. Here we give a few examples:

1. In the energy management of buildings, an important topic is to model how building design and operational variables affect energy consumption. Identification of this relationship helps the design and operation of energy-efficient buildings. However, buildings with the same design and operation may have different levels of energy consumption/efficiency, depending on where the buildings are located. A good building design/operation needs to consider outdoor environmental conditions, such as geographical location, temperature, humidity, and air flow rate (Eisenhower *et al.*, 2012).
2. In mobile communication networks, a key interest is to model how traffic volume variables affect Quality of Service (QoS) metrics, such as packet delay and loss. Understanding this relationship helps network capacity management and optimization. It is well known that mobile network operations are affected by environmental conditions, such as the type of landform (valley, mountain, or plain) and weather (Hardy, 2001).
3. In the wind energy industry, a continuing interest is to model how wind speed and direction affect the power output of a wind turbine. This relationship is used for a number of important tasks, including prediction of wind power production and evaluation of the turbine's energy production efficiency. It has been found that this relationship varies with respect to environmental conditions, such as location of the wind turbine (offshore or inland), temperature, humidity, and air pressure (Byon *et al.*, 2015).

Let $\mathbf{X} = (X_1, \ldots, X_p)^T$ denote the design and operational variables of a system, called input variables in this article, Y denote the performance or output variable of the system, and $\mathbf{Z} = (Z_1, \ldots, Z_q)^T$ denote the environmental variables. To account for environmental conditions, an intuitive approach is to concatenate the input and environmental variables into a single predictor set, which is then linked with Y by a statistical model, such as a regression. To select important predictors, classic approaches are forward selection, backward elimination, and stepwise regression (Montgomery *et al.*, 2015). With high-dimensional predictors, especially under the "small-n-large-p" setting, optimization-based methods capable of selecting a *sparse* subset of the predictors have been shown to be more effective, and indeed they constitute a popular research area in modern statistics and machine learning societies. Typical sparse regression methods include lasso (Tibshirani, 1996), Smoothly Clipped Absolute Deviation (SCAD) (Fan and Li, 2001), adaptive lasso (Zou, 2006), group lasso (Yuan and Lin, 2006), and elastic net (Zou and Hastie, 2005), to name just a few. However, these methods do not explicitly model the interaction between $\mathbf{X}$ and $\mathbf{Z}$. To consider the interaction in a regression model, a

CONTACT Jing Li jinglz@asu.edu
Color versions of one or more of the figures in this article can be found online at www.tandfonline.com/uiie.
Supplemental data for this article can be accessed on the publisher's website.

straightforward option is to apply the aforementioned methods to an expanded predictor set that includes not only the individual predictors but also their interactions up to an order of interest. However, this does not honor the well-known "hierarchical principles" in regression fitting, which state that an interaction term can only be included in a model if at least one (weak hierarchy) or all (strong hierarchy) of the individual predictors involved in the interaction term are also in the model (Hamada and Wu, 1992; Chipman, 1995).

To account for the hierarchical principles, most existing work focuses on models that involve pairwise interactions. Putting this into our context, this means a model of the following format:

$$Y = \sum_{i=1}^{p} \alpha_i X_i + \sum_{j=1}^{q} \omega_j Z_j + \sum_{j=1}^{q} \sum_{i=1}^{p} \gamma_{ij} X_i Z_j + \varepsilon, \qquad (1)$$

where α_i, ω_j, and γ_{ij} are regression coefficients. Specifically, Choi *et al.* (2010) proposed to re-parameterize the coefficient for each interaction term into a product—i.e., $\gamma_{ij} = \vartheta_{ij}\alpha_i\omega_j$—which enforces the strong hierarchy in the sense that whenever α_i and ω_j are zero, the coefficient for the interaction, γ_{ij}, is automatically zero. They further proposed to impose one l_1-regularization on ϑ_{ij} and another one on α_i and ω_j to enable a sparse estimation obeying the strong hierarchy. This model is non-convex and an iterative algorithm was developed for model estimation. Convex formulations display better mathematical tractability and computational efficiency. Toward this end, there have been a few developments. Yuan *et al.* (2009) proposed a convex optimization formulation by modifying the non-negative garrote (Breiman, 1995) and adding linear inequality constraints to enforce hierarchy. Zhao *et al.* (2009) proposed the Composite Absolute Penalties method that allows given grouping and hierarchical relationships of predictors to be expressed. Bien *et al.* (2013) proposed to honor the strong and weak hierarchy by extending the lasso formulation to include convex constraints. However, all of the aforementioned methods have the following limitations: First, they are either restricted to modeling of pairwise interactions or require the order of interactions to be pre-determined. Second, if used in our context, these methods all have to assume that the environmental variables *linearly* affect the input/output relationship, which can be violated in modeling of complex systems in practice. To see this more clearly, we can re-write Equation (1) as Equation (2):

$$Y = \sum_{j=1}^{q} \omega_j Z_j + \sum_{i=1}^{p} \left(\alpha_i + \sum_{j=1}^{q} \gamma_{ij} Z_j \right) X_i + \varepsilon, \qquad (2)$$

which shows that the relationship between X_i and Y, characterized by $\alpha_i + \sum_{j=1}^{q} \gamma_{ij} Z_j$, is linearly related to the environmental variables Z_j.

To relax the linearity assumption, we may use a nonlinear function, $f_i(\mathbf{Z})$, to replace the linear function $\alpha_i + \sum_{j=1}^{q} \gamma_{ij} Z_j$. Then, the model becomes a Varying Coefficient (VC) model. Various types of VC models have been developed in the literature. The estimation methods can be broadly classified into spline estimators (Hastie and Tibshirani, 1993; Hoover *et al.*, 1998; Chiang *et al.*, 2001), kernel-type estimators (Fan and Zhang, 1999; Xia and Li, 1999), and wavelet estimators (Zhou and You, 2004). Extended work beyond these classic methods exists. For example, Cai *et al.* (1999) generalized the response variable of VC models to the exponential family. Fan *et al.* (2003) proposed an adaptive VC model, in which $\mathbf{Z}$ is assumed to be unknown and estimated as a linear combination of input variables. Zhang *et al.* (2002) introduced a semi-VC model considering the co-existence of varying and constant coefficients in one model. This work was further extended by Hu and Xia (2012), who added an l_1-regularization to the constant coefficients to enable sparse estimation.

A common assumption of VC models is that $f_i(\mathbf{Z})$ is a smooth function of $\mathbf{Z}$. In this article, we have a different focus by aiming to identify a partition of the space of the environmental variables $\mathbf{Z}$, such that the input/output relationship in each subdivision of the partition remains constant, whereas this relationship varies across different subdivisions. From the practical point of view, each subdivision of the partition corresponds to a type of environmental condition under which the system is functioning in a specific way. When the environmental condition changes, the system may function differently. A notable difference between the proposed method and VC models is that the former relaxes the smoothness constraint; i.e., it allows unsmooth changes in the input/output relationship at *adjacent* subdivisions of the partition. This relaxation/flexibility has important practical value, because it allows for modeling of systems that are sensitive to environmental changes. For example, in mobile communication networks, the traffic volume–QoS relationship can be remarkably different even when the network is deployed at two adjacent geographical areas; e.g., when the two adjacent areas have different landforms, such as a valley next to a mountain. In building energy management, it has been observed that there exist tipping points in terms of the environmental conditions. That is, when the temperature, humidity, and air flow rate are within certain ranges, energy consumption is related to building design and operational variables in a specific way. This relationship may dramatically change if the environmental conditions are outside the ranges. Indeed, VC models can be considered as a special case of the proposed method when the partition by the proposed method is sufficiently fine and the change in the input/output relationship across adjacent subdivisions of the partition satisfies smoothness constraints. Another advantage of the proposed method is that it can take both numerical and categorical environmental variables into consideration, whereas VC models have inherent difficulty in handling categorical variables. The difficulty is due to there being no meaningful measure for the adjacency of the different categories for a categorical variable and, consequently, it is meaningless to require smoothness for categorical variables. Last, but not least, the proposed method is efficient, whereas fitting of a VC model with more than one environmental variable can be computationally very challenging.

The contributions of this article are summarized as follows:

1. We propose a new statistical method, called Sparse Partitioned-Regression (SPR), to account for the nonlinear interactions between the design/operational (input) variables of a system and multivariate environments in predicting the system's performance (output). Compared with existing sparse regression models, SPR naturally honors the hierarchical principle and

can identify the order and nonlinear pattern of the interactions between input and environmental variables in a data-driven manner. Compared with VC models, SPR relaxes the smoothness constraint in the change of input/output relationship across different environmental conditions, can model both numerical and categorical environmental variables, and is computationally efficient. Additionally, SPR can select small subsets of the environmental variables together with their optimal partitions and the input variables, respectively, that are most relevant to the output variable and therefore can handle high-dimensional problems.
2. We study the theoretical properties of SPR and derive oracle inequalities to quantify the risks of the SPR estimators for both prediction and classification problems. Also, we propose an efficient algorithm for model estimation.
3. We present an application of using building design and operational variables, outdoor environmental variables, and their interactions to predict building energy consumption based on the Department of Energy's (DOE's) *EnergyPlus* datasets. The SPR approach has a significantly higher level of prediction accuracy than competing methods. The application also helps knowledge discovery for building energy management.

The rest of this article is organized as follows: Section 2 proposes the model formulation. Section 3 presents an algorithm for model estimation. Section 4 studies the theoretical properties (i.e., the oracle inequalities) of the SPR model. Section 5 presents simulation studies. Section 6 presents an application of predicting building energy consumption using building design and operational variables together with environmental variables. Section 7 is the conclusion.

2. Formulation of SPR

Consider a training data set with n samples. Let $\mathbf{x}_k$, $\mathbf{z}_k$, y_k be the measurement on the input, environmental, and output variables of the kth sample, $k = 1, \ldots, n$. Consider a partition of the space of the environmental variables into n_S disjoint subdivisions; i.e., $\boldsymbol{\mathcal{S}} = \{\boldsymbol{\mathcal{S}}^{(1)}, \ldots, \boldsymbol{\mathcal{S}}^{(n_S)}\}$. Each sample belongs to one and only one subdivision depending on its environmental variables. That is, the kth sample belongs to the rth subdivision if $\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}$. Within each subdivision, the relationship between the input and output variables is characterized by a model $\mathrm{y}_k = f(\mathbf{x}_k; \boldsymbol{\theta}^{(r)})$ with parameter set $\boldsymbol{\theta}^{(r)}$. The exact form of the model is unknown but can be estimated from data. To assess how good the estimation is, we can define a risk function between the observed y_k and the estimated model $\hat{f}(\mathbf{x}_k; \boldsymbol{\theta}^{(r)})$, $L(\mathrm{y}_k, \hat{f}(\mathbf{x}_k; \boldsymbol{\theta}^{(r)}))$. Averaging over all of the samples in the training data set, we can obtain the empirical risk function as follows:

$$\hat{R}(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}) = \frac{1}{n} \sum_{k=1}^{n} \sum_{r=1}^{n_S} L\big(y_k, \hat{f}\big(\mathbf{x}_k; \boldsymbol{\theta}^{(r)}\big)\big) \times I\big(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\big), \quad (3)$$

where $\boldsymbol{\theta} = \{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(n_S)}\}$. $I(\cdot)$ is an indicator function.

In this article, both the partition $\boldsymbol{\mathcal{S}}$ and the model parameters $\boldsymbol{\theta}$ are treated as unknown. To estimate them, simply minimizing the empirical risk in Equation (3) will cause overfitting; i.e., finer partitions would always be preferred. To address this problem, we propose two estimators, a penalized estimator and a held-out estimator.

Definition 1: The penalized estimator is defined as

$$\hat{\boldsymbol{\mathcal{S}}}, \hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}}{\operatorname{argmin}} \{\hat{R}(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}) + \lambda_{\boldsymbol{\mathcal{S}}}\, pen(\boldsymbol{\mathcal{S}})\},$$

where $pen(\boldsymbol{\mathcal{S}})$ is a measure for the complexity of the partition. The finer the partition, the higher the complexity. $\lambda_{\boldsymbol{\mathcal{S}}}$ is a penalty parameter. Alternatively, if there are sufficient training samples, we may divide the entire training data set into a training set and a validation set consisting of n_1 and n_2 samples, respectively. Given $\boldsymbol{\mathcal{S}}$, we can obtain an estimate for $\boldsymbol{\theta}$ that minimizes the empirical risk evaluated on the training set alone; that is,

$$\hat{\boldsymbol{\theta}}_{tr} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, \hat{R}_{tr}(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{n_1} \sum_{k=1}^{n_1} \sum_{r=1}^{n_S} L\big(y_k, \hat{f}\big(\mathbf{x}_k; \boldsymbol{\theta}^{(r)}\big)\big) \times I\big(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\big).$$

Then, this estimate is plugged into the empirical risk evaluated on the held-out validation set:

$$\hat{R}_{val}(\boldsymbol{\mathcal{S}}, \hat{\boldsymbol{\theta}}_{tr}) = \frac{1}{n_2} \sum_{k=1}^{n_2} \sum_{r=1}^{n_S} L\big(y_k, \hat{f}\big(\mathbf{x}_k; \hat{\boldsymbol{\theta}}_{tr}^{(r)}\big)\big) \times I\big(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\big),$$

which measures the generalization error of the estimate and the partition $\boldsymbol{\mathcal{S}}$. Minimizing this generalization error yields the held-out estimator.

Definition 2: The held-out estimator is defined as

$$\tilde{\boldsymbol{\mathcal{S}}}, \tilde{\boldsymbol{\theta}} = \underset{\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}}{\operatorname{argmin}}\, \hat{R}_{val}(\boldsymbol{\mathcal{S}}, \hat{\boldsymbol{\theta}}_{tr}).$$

The afore-proposed framework can be used to for a numerical or a categorical output variable, resulting in a predictive model or a classification model, respectively. In the predictive model, the relationship between the input and output variables can be characterized by a linear regression; i.e., $\mathrm{y}_k = \mathbf{x}_k^T \boldsymbol{\alpha}^{(r)} + \varepsilon^{(r)}$, $\varepsilon^{(r)} \sim N(0, \sigma_\varepsilon^{2(r)})$. A typical risk function for a linear regression is the negative log-likelihood function, using which the empirical risk function in Equation (3) can be written as

$$\hat{R}(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}) = \hat{R}\big(\boldsymbol{\mathcal{S}}, \boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\big) = \frac{1}{n} \sum_{k=1}^{n} \sum_{r=1}^{n_S} \times \left\{ \frac{\big(y_k - \mathbf{x}_k^T \boldsymbol{\alpha}^{(r)}\big)^2}{\sigma_\varepsilon^{2(r)}} + \log \sigma_\varepsilon^{2(r)} \right\} I\big(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\big), \quad (4)$$

where $\boldsymbol{\alpha} = \{\boldsymbol{\alpha}^{(1)}, \ldots, \boldsymbol{\alpha}^{(n_S)}\}$ and $\boldsymbol{\sigma}_\varepsilon^2 = \{\sigma_\varepsilon^{2(1)}, \ldots, \sigma_\varepsilon^{2(n_S)}\}$ are the model parameters. When the input variables are high-dimensional, an l_1-regularized negative log-likelihood function can be used and Equation (4) can be further written as

$$\hat{R}\big(\boldsymbol{\mathcal{S}}, \boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\big) = \sum_{r=1}^{n_S} \left\{ \frac{1}{n} \sum_{k=1}^{n} \left\{ \frac{\big(y_k - \mathbf{x}_k^T \boldsymbol{\alpha}^{(r)}\big)^2}{\sigma_\varepsilon^{2(r)}} + \log \sigma_\varepsilon^{2(r)} \right\} \times I\big(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\big) + \lambda_\alpha^{(r)} \|\boldsymbol{\alpha}^{(r)}\|_1 \right\}, \quad (5)$$

where $\|\cdot\|_1$ denotes the l_1-norm, which has been used in lasso and is known to enforce sparsity in model estimation. Other well-known sparsity-induced regularizations, such as those used in fused-lasso and group-lasso, can also be adopted in Equation (5) depending on the structure of the input variables. $\lambda_\alpha^{(r)}$ is a regularization parameter. Furthermore, using Equation (5) in Definitions 1 and 2, a sparse penalized estimator and a sparse held-out estimator for a predictive model can be obtained, respectively.

In a classification model (i.e., when the output variable is categorical), the relationship between the input and output variables can be characterized by a multinomial logistic regression, which links the probability of the output variable being in the mth class with the input variables in the form of

$$P(\mathrm{y}_k = m) = \frac{\exp\left(\mathbf{x}_k^T \boldsymbol{\beta}_m^{(r)}\right)}{\sum_{l=1}^{M} \exp\left(\mathbf{x}_k^T \boldsymbol{\beta}_l^{(r)}\right)}, \quad m = 1, \ldots, M.$$

A typical risk function for a multinomial logistic regression is the negative log-likelihood function, using which the empirical risk function in Equation (3) can be written as

$$\hat{R}(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}) = \hat{R}(\boldsymbol{\mathcal{S}}, \boldsymbol{\beta}) = \frac{1}{n} \sum_{k=1}^{n} \sum_{r=1}^{n_S} \left\{ \log \sum_{l=1}^{M} \exp\left(\mathbf{x}_k^T \boldsymbol{\beta}_l^{(r)}\right) - \sum_{l=1}^{M} I\left(\mathrm{y}_k = l\right) \mathbf{x}_k^T \boldsymbol{\beta}_l^{(r)} \right\} I\left(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\right), \tag{6}$$

where $\boldsymbol{\beta} = \{\boldsymbol{\beta}_1^{(1)}, \ldots, \boldsymbol{\beta}_M^{(1)}, \boldsymbol{\beta}_1^{(2)}, \ldots, \boldsymbol{\beta}_M^{(2)}, \ldots, \boldsymbol{\beta}_1^{(n_S)}, \ldots, \boldsymbol{\beta}_M^{(n_S)}\}$ are the model parameters. When the input variables are high-dimensional, an l_1-regularized negative log-likelihood function can be used and Equation (6) can be further written as

$$\hat{R}(\boldsymbol{\mathcal{S}}, \boldsymbol{\beta}) = \sum_{r=1}^{n_S} \left\{ \frac{1}{n} \sum_{k=1}^{n} \left\{ \log \sum_{l=1}^{M} \exp\left(\mathbf{x}_k^T \boldsymbol{\beta}_l^{(r)}\right) - \sum_{l=1}^{M} I\left(\mathrm{y}_k = l\right) \mathbf{x}_k^T \boldsymbol{\beta}_l^{(r)} \right\} \times I\left(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(r)}\right) + \lambda_\beta^{(r)} \sum_{l=1}^{M} \|\boldsymbol{\beta}_l^{(r)}\|_1 \right\}. \tag{7}$$

Furthermore, using Equation (7) in Definitions 1 and 2, a sparse penalized estimator and a sparse held-out estimator for a classification model can be obtained, respectively.

In summary, the proposed SPR consists of four models: a sparse penalized estimator for prediction, a sparse held-out estimator for prediction, a sparse penalized estimator for classification, and a sparse held-out estimator for classification. Using the first model of SPR as an example, Proposition 1 shows that SPR obeys the weak hierarchy. This property also holds for the other three models of SPR. The first model of SPR takes the form of $Y = \sum_{r=1}^{n_S} \{\mathbf{X}^T \boldsymbol{\alpha}^{(r)} + \varepsilon^{(r)}\} I(\mathbf{Z} \in \boldsymbol{\mathcal{S}}^{(r)})$. Let $\hat{\boldsymbol{\mathcal{S}}}^{(r)}$ denote the rth subdivision and $\hat{\boldsymbol{\alpha}}^{(r)} = (\hat{\alpha}_0^{(r)}, \hat{\alpha}_1^{(r)}, \ldots, \hat{\alpha}_p^{(r)})^T$ denote the linear coefficients in the rth subdivision produced by the sparse penalized estimator of the first model.

Proposition 1. *SPR obeys the weak hierarchy; i.e., if there is an interaction between the i^*th input variable, X_{i^*}, and the environmental variables in the r^*th subdivision, $\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r^*)}$, then the main effect of $\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r^*)}$ must exist.*

Proof. The estimated SPR can be written as

$$\hat{Y} = \sum_{r=1}^{n_S} \mathbf{X}^T \hat{\boldsymbol{\alpha}}^{(r)} I\left(\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r)}\right) = \sum_{r=1}^{n_S} \hat{\alpha}_0^{(r)} I\left(\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r)}\right) + \sum_{r=1}^{n_S} \sum_{i=1}^{p} \hat{\alpha}_i^{(r)} X_i I\left(\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r)}\right).$$

Suppose there is an interaction between X_{i^*} and $\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r^*)}$, which means that $\hat{\alpha}_{i^*}^{(r^*)} \neq 0$ and $I(\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r^*)}) = 1$. Then, $\hat{\alpha}_0^{(r^*)} I(\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r^*)}) = \hat{\alpha}_0^{(r^*)} \neq 0$; i.e., the main effect of $\mathbf{Z} \in \hat{\boldsymbol{\mathcal{S}}}^{(r^*)}$ exists. Here, $\hat{\alpha}_0^{(r^*)} \neq 0$ because the l_1-penalty used in the sparse estimator follows the common practice of sparse regressions (e.g., lasso) that the intercept will not be penalized so it is always non-zero. □

3. Algorithm

Among the four models proposed in the previous section, we will focus on "the held-out estimator for classification" in describing the algorithm for model estimation. The algorithms for estimating the other three models share a similar procedure. The goal of the algorithm is to find an optimal partition of the space of the environmental variables and a multinomial logistic regression between the input and output variables within each subdivision of the partition based on a training set, such that the empirical risk evaluated on a held-out validation set is minimized. To achieve this goal, an exhaustive search for the optimal partition is computationally infeasible. We propose a computationally efficient algorithm based on recursive partitioning. The basic idea is to first find a variable within the set of environmental variables $\mathbf{Z}$ and a splitting point of that variable that best split the training set into two sub-groups ("best" in terms of optimizing a splitting criterion). Then, this process is repeated within each sub-group identified in the previous step until a stopping criterion is met.

SPR looks similar to the recursive partitioning used to build a Classification and Regression Tree (CART; Breiman *et al.* (1984)). The major difference is in the splitting criterion. In CART, the splitting variable and splitting point are selected to achieve the greatest reduction in an impurity measure. A sub-group is pure if it only consists of samples belonging to the same class of the output variable Y. The higher the mix of different classes, the more impure the sub-group becomes. Typical impurity measures include the Gini Index, entropy, and others. In our recursive partitioning algorithm, the splitting criterion considers not only Y but also the input variables $\mathbf{X}$; i.e., the relationship between Y and $\mathbf{X}$ characterized by a multinomial logistic regression. It selects the splitting variable and splitting point that achieve the greatest reduction in the empirical risk evaluated on the validation set. Specifically, let s denote a subdivision in the current partition $\boldsymbol{\mathcal{S}}^c$. We can compute the empirical risk of this subdivision evaluated on the validation set, $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^c, \hat{\boldsymbol{\theta}}_{tr}^{(s)})$. To find which environmental variable to use for further splitting s, we search through all of the environmental variables in $\mathbf{Z}$. For each $Z_j \in \mathbf{Z}$, the subdivision can be split into a left region defined by $Z_j \leq z_j$ and a right region defined by $Z_j > z_j$. z_j is a candidate splitting point. Then, we compute the empirical risk of

each region evaluated on the validation set, $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c+1}, \hat{\boldsymbol{\theta}}_{tr}^{(Z_j \le z_j)})$ and $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c+1}, \hat{\boldsymbol{\theta}}_{tr}^{(Z_j > z_j)})$, and a reduction in the empirical risk as

$$\Delta\hat{R}_{val}^{j} = n_s\hat{R}_{val}\big(\boldsymbol{\mathcal{S}}^{c}, \hat{\boldsymbol{\theta}}_{tr}^{(s)}\big) - n_L\hat{R}_{val}\big(\boldsymbol{\mathcal{S}}^{c+1}, \hat{\boldsymbol{\theta}}_{tr}^{(Z_j \le z_j)}\big) - n_R\hat{R}_{val}\big(\boldsymbol{\mathcal{S}}^{c+1}, \hat{\boldsymbol{\theta}}_{tr}^{(Z_j > z_j)}\big), \tag{8}$$

where n_s, n_L, and n_R are the sample sizes of the subdivision s, left region, and right region, respectively. The environmental variable and the splitting point with the largest reduction $\Delta\hat{R}_{val}^{j}$ are selected to split s. If no positive reduction is found, s will not be split and the algorithm stops. Another consideration in the stopping criterion is that the sample size of the subdivision s reaches a pre-defined minimum number.

Next, we discuss two technical details on the splitting criterion of our algorithm. One is regarding the selection of candidate splitting points for a categorical environmental variable. Assuming that the variable has B categories, there are $2^{(B-1)} - 1$ possible splits. Environmental variables with a large number of categories are common. For example, in studying the relationship between building design variables and energy consumption, an important environmental variable is the geographical location of a building. If using "states" in the United States to describe the location, $B = 50$, resulting in $2^{49} - 1$ possible splits. To reduce the computational burden, we propose an alternative transformation-based approach: First, we transform the B categories of the environmental variable into B ordinal numbers based on the average output variable for each category computed on the training set. Then, we consider every possible binary split of the ordered sequence of the B ordinal numbers as a candidate split. This results in $(B-1)$ candidate splitting points, which compose a much smaller subset of the $2^{(B-1)} - 1$ splitting points. A desirable property of this approach is that it guarantees that the optimal split within the $2^{(B-1)} - 1$ splitting points is included in the subset of $(B-1)$ candidate splitting points.

The other technical detail of the proposed splitting criterion is regarding the computation of the $\hat{R}_{val}(\cdot)$ in Equation (8). Take $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c}, \hat{\boldsymbol{\theta}}_{tr}^{(s)})$ as an example. As we focus on the classification model, $\hat{\boldsymbol{\theta}}_{tr}^{(s)}$ consists of coefficients of a multinomial logistic regression for subdivision s estimated from the training set; i.e., $\hat{\boldsymbol{\theta}}_{tr}^{(s)} = \hat{\boldsymbol{\beta}}_{tr}^{(s)} = \{\hat{\boldsymbol{\beta}}_{tr,1}^{(s)}, \ldots, \hat{\boldsymbol{\beta}}_{tr,M}^{(s)}\}$. To obtain $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$, we minimize an l_1-regularized negative log-likelihood function; that is,

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{tr}^{(s)} &= \underset{\boldsymbol{\beta}_{tr}^{(s)}}{\operatorname{argmin}}\, \hat{R}_{tr}\big(\boldsymbol{\mathcal{S}}^{c}, \boldsymbol{\beta}_{tr}^{(s)}\big) \\ &= \underset{\boldsymbol{\beta}_{tr}^{(s)}}{\operatorname{argmin}} \left\{ \frac{1}{n_1}\sum_{k=1}^{n_1} \left\{ \log\sum_{l=1}^{M}\exp\big(\mathbf{x}_k^T\boldsymbol{\beta}_l^{(s)}\big) - \sum_{l=1}^{M} \times I(\mathrm{y}_k = l)\mathbf{x}_k^T\boldsymbol{\beta}_l^{(s)} \right\} \times I\big(\mathbf{z}_k \in \boldsymbol{\mathcal{S}}^{(s)}\big) + \lambda_\beta^{(s)}\sum_{l=1}^{M}\|\boldsymbol{\beta}_l^{(s)}\|_1 \right\}. \end{aligned} \tag{9}$$

For a given $\lambda_\beta^{(s)}$, Equation (9) is a convex optimization that can be solved efficiently. To find the optimal $\lambda_\beta^{(s)}$, we can conduct a line search on a series of values for $\lambda_\beta^{(s)}$. Under each value, we solve the convex optimization in Equation (9), use the estimated $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$—i.e., the training model—to classify the samples in the validation set, and compute a misclassification error rate. The optimal $\lambda_\beta^{(s)}$ is the one under which the misclassification error rate is minimized. Furthermore, due to the l_1-regularization in Equation (9), the $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$ estimated under the optimal $\lambda_\beta^{(s)}$ will be sparse with many zero elements, and the non-zero elements will suffer from a shrinking effect by having a magnitude smaller than what they are supposed to be. To correct this estimation bias, we re-estimate the $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$ in Equation (9) without the l_1-regularization but enforcing the sparse pattern obtained from the previous l_1-regularized estimation. Finally, we plug the re-estimated $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$ into $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c}, \hat{\boldsymbol{\theta}}_{tr}^{(s)})$.

We conclude this section by presenting the major steps of the proposed algorithm in estimating SPR. The input to the algorithm includes a training set and a validation set on the input, environmental, and output variables, $\mathbf{X}$, $\mathbf{Z}$, and Y, and a minimum sample size requirement n_{min}. At the cth step of the recursive partitioning, let $\boldsymbol{\mathcal{S}}^{c}$ be the partition of the space of the environmental variables. For each subdivision in the partition, $s \in \boldsymbol{\mathcal{S}}^{c}$, perform the following steps:

Step 1: If the sample size of the subdivision s in the training set is smaller than n_{min}, stop splitting this subdivision; otherwise, proceed to Step 2.

Step 2: Fit a multinomial logistic regression model between the input and output variables and estimate the l_1-regularized regression coefficients $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$ based on Equation (9) and using the training set. The optimal $\lambda_\beta^{(s)}$ in Equation (9) is selected by minimizing the misclassification error rate of applying the training model to classify the samples in the validation set.

Step 3: Re-estimate the $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$ using Equation (9) without the l_1-regularization but enforcing the sparse pattern obtained from Step 2.

Step 4: Use the re-estimated $\hat{\boldsymbol{\beta}}_{tr}^{(s)}$ to compute the empirical risk evaluated on the validation set, $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c}, \hat{\boldsymbol{\theta}}_{tr}^{(s)})$.

Step 5: For each environmental variable $Z_j \in \mathbf{Z}$ and each candidate splitting point z_j, split the subdivision s into a left region defined by $Z_j \le z_j$ and a right region defined by $Z_j > z_j$. If Z_j is a categorical variable, use the aforementioned transformation-based approach to select the candidate splitting points. For each region, repeat Steps 2 to 4 and obtain $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c+1}, \hat{\boldsymbol{\theta}}_{tr}^{(Z_j \le z_j)})$ and $\hat{R}_{val}(\boldsymbol{\mathcal{S}}^{c+1}, \hat{\boldsymbol{\theta}}_{tr}^{(Z_j > z_j)})$. Use Equation (8) to compute the empirical risk reduction $\Delta\hat{R}_{val}^{j}$.

Step 6: If no positive $\Delta\hat{R}_{val}^{j}$ is found, stop splitting the subdivision s; otherwise, split the subdivision using the environmental variable and splitting point with the largest $\Delta\hat{R}_{val}^{j}$. ▲

The output from the algorithm is a partition of the space of the environmental variables and a multinomial logistic regression model between the input and output variables for each subdivision of the partition. To classify a new sample—e.g., the $(n+1)$th sample—we first use the environmental variables of this sample, $\mathbf{z}_{n+1}$, to find to which subdivision this sample belongs. Then, we use the input variables, $\mathbf{x}_{n+1}$, in the multinomial logistic regression of this subdivision to predict the class membership of the output variable, $\hat{y}_{n+1}$. The SPR

algorithm, as presented here, is programmed using the R software.

It is worth mentioning that when the sample size allows, it would be better to separate the data into a training and two validation sets, with one validation set used to tune the penalty parameter and the other to find the splitting point. This would reduce the potential risk of overfitting. When the sample size is limited, we could use a single validation set to serve the two purposes, as the current algorithm is in fact designed to do. This may not be much of an issue as our simulation and application studies show that the algorithm grants a good accuracy on a separate test set. However, a cautious strategy for avoiding the potential overfitting with the current algorithm may be to increase the n_{min}. This issue is worthy of more in-depth future investigation.

4. Theoretical properties

We derive the oracle inequalities for the penalized estimator and held-out estimator of the classification and predictive models in Theorems 1 to 4, respectively. An oracle inequality is a bound on the risk of a statistical estimator, which shows that the performance of the estimator is almost (up to numerical constants) as good as an ideal estimator that replies on perfect information supplied by an oracle and is not available in practice (Vapnik, 1998). An oracle inequality is an important property of a statistical estimator. For theoretical tractability, we focus on Dyadic Recursive Partitions (DRPs) of the space of the environmental variables. In DRPs, splitting a previously obtained subdivision can only happen at the midpoint of the range of an environmental variable. First, we define some notations. Let R^* be the minimum possible risk—i.e., the Bayes' risk—defined as

$$R^* = R\left(\boldsymbol{\mathcal{S}}^*, \boldsymbol{\theta}^*\right) = \inf_{\boldsymbol{\mathcal{S}} \in \boldsymbol{\mathcal{S}}_{DRP}, \boldsymbol{\theta} \in \boldsymbol{\Omega}_{\boldsymbol{\theta}}} R\left(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}\right),$$

where $R(\boldsymbol{\mathcal{S}}, \boldsymbol{\theta}) = \sum_{r=1}^{n_S} E[L(Y, \hat{f}(\mathbf{x}; \boldsymbol{\theta}^{(r)})) \times I(\mathbf{Z} \in \boldsymbol{\mathcal{S}}^{(r)})]$, $\boldsymbol{\mathcal{S}}_{DRP}$ is the set of all DRPs, and $\boldsymbol{\Omega}_{\boldsymbol{\theta}}$ is the domain of the model parameters $\boldsymbol{\theta}$. For a classification model, $\boldsymbol{\theta}$ consists of coefficients of multinomial logistic regressions; i.e., $\boldsymbol{\theta} = \boldsymbol{\beta}$. We assume that $\boldsymbol{\beta}$ is bounded; that is,

$$\boldsymbol{\Omega}_{\boldsymbol{\beta}} = \{\boldsymbol{\beta} | \max_{\substack{r=1,\ldots,n_S \\ l=1,\ldots,M}} |\mathbf{x}^T \boldsymbol{\beta}_l^{(r)}| \leq B\},$$

and B is a positive constant. For a predictive model, $\boldsymbol{\theta}$ consists of coefficients and residual variances of linear regressions; i.e., $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\}$. We assume that $\boldsymbol{\alpha}$ and $\boldsymbol{\sigma}_\varepsilon^2$ are bounded; i.e., $\boldsymbol{\Omega}_{\boldsymbol{\alpha},\boldsymbol{\sigma}_\varepsilon^2} = \{\boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2 | \max_{r=1,\ldots,n_S} |\mathbf{x}^T \boldsymbol{\alpha}^{(r)}| \leq A, \max_{r=1,\ldots,n_S} |\log \tau^{(r)}| \leq L\}$.

Also, $\tau^{(r)}$ is the reciprocal of $\sigma_\varepsilon^{2(r)}$ and A and L are positive constants. Finally, let $[\![\boldsymbol{\mathcal{S}}]\!]$ denote the complexity of $\boldsymbol{\mathcal{S}}$. Specifically, $[\![\boldsymbol{\mathcal{S}}]\!]$ can be the length of a finite-length binary string used to encode $\boldsymbol{\mathcal{S}}$ in computers. Proofs of Theorems 1 to 4 can be found in the online Supplemental Material.

Theorem 1. (oracle inequality of the penalized estimator for the classification model). *Let $\hat{\boldsymbol{\mathcal{S}}}, \hat{\boldsymbol{\beta}}$ be the penalized estimator. For a sufficiently large n and any $\delta \in (0, 1)$, the excess risk of the penalized estimator with respect to the Bayes' risk satisfies the following inequality:*

$$R(\hat{\boldsymbol{\mathcal{S}}}, \hat{\boldsymbol{\beta}}) - R^* \leq \inf_{\boldsymbol{\mathcal{S}} \in \boldsymbol{\mathcal{S}}_{DRP}} \{ \inf_{\boldsymbol{\beta} \in \boldsymbol{\Omega}_{\boldsymbol{\beta}}} \left(R\left(\boldsymbol{\mathcal{S}}, \boldsymbol{\beta}\right) - R^*\right) + 2\, pen_c\left(\boldsymbol{\mathcal{S}}\right)\},$$

with probability at least $1 - \delta$, where

$$pen_c\left(\boldsymbol{\mathcal{S}}\right) = n_S \left(BM\sqrt{2v_1} + B + \log M\right) \sqrt{\frac{[\![\boldsymbol{\mathcal{S}}]\!] \log 2 + \log\left(2/\delta\right)}{n}}.$$

Theorem 2. (oracle inequality of the held-out estimator for the classification model). *Let $\tilde{\boldsymbol{\mathcal{S}}}, \tilde{\boldsymbol{\beta}}$ be the held-out estimator. For a sufficiently large n_1, n_2, and any $\delta \in (0, 1)$, the excess risk of the held-out estimator with respect to the Bayes' risk satisfies the following inequality:*

$$R(\tilde{\boldsymbol{\mathcal{S}}}, \tilde{\boldsymbol{\beta}}) - R^* \leq \inf_{\boldsymbol{\mathcal{S}} \in \boldsymbol{\mathcal{S}}_{DRP}} \left\{ \inf_{\boldsymbol{\beta} \in \boldsymbol{\Omega}_{\boldsymbol{\beta}}} (R(\boldsymbol{\mathcal{S}}, \boldsymbol{\beta}) - R^*) + 2\phi_c^1(\boldsymbol{\mathcal{S}}) + \phi_c^2(\boldsymbol{\mathcal{S}}) \right\} + \phi_c^2(\tilde{\boldsymbol{\mathcal{S}}}),$$

with probability at least $1 - \delta$, where

$$\phi_c^1\left(\boldsymbol{\mathcal{S}}\right) = n_S \left(BM\sqrt{2v_1} + B + \log M\right) \sqrt{\frac{[\![\boldsymbol{\mathcal{S}}]\!] \log 2 + \log\left(2/\delta\right)}{n_1}},$$

and

$$\phi_c^2\left(\boldsymbol{\mathcal{S}}\right) = n_S \left(BM\sqrt{2v_1} + B + \log M\right) \sqrt{\frac{[\![\boldsymbol{\mathcal{S}}]\!] \log 2 + \log\left(2/\delta\right)}{n_2}}.$$

Theorem 3. (oracle inequality of the penalized estimator for the predictive model). *Let $\hat{\boldsymbol{\mathcal{S}}}\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}}_\varepsilon^2$ be the penalized estimator. For a sufficiently large n and any $\delta \in (0, 1)$, the excess risk of the penalized estimator with respect to the Bayes' risk satisfies the following inequality:*

$$R\left(\hat{\boldsymbol{\mathcal{S}}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}}_\varepsilon^2\right) - R^* \leq \inf_{\boldsymbol{\mathcal{S}} \in \boldsymbol{\mathcal{S}}_{DRP}} \left\{ \inf_{\{\boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\} \in \boldsymbol{\Omega}_{\boldsymbol{\alpha},\boldsymbol{\sigma}_\varepsilon^2}} \left(R\left(\boldsymbol{\mathcal{S}}, \boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\right) - R^*\right) + 2\, pen_p\left(\boldsymbol{\mathcal{S}}\right) \right\},$$

with probability at least $1 - \delta$, where

$$pen_p\left(\boldsymbol{\mathcal{S}}\right) = n_S \left(L + e^L C\right) \sqrt{\frac{[\![\boldsymbol{\mathcal{S}}]\!] \log 2 + \log\left(2/\delta\right)}{n}},$$

and $C = \sqrt{2v_2} + 2A\sqrt{2v_1} + A^2$.

Theorem 4. (oracle inequality of the held-out estimator for the predictive model). *Let $\tilde{\boldsymbol{\mathcal{S}}}, \tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\sigma}}_\varepsilon^2$ be the held-out estimator. For a sufficiently large n_1, n_2, and any $\delta \in (0, 1)$, the excess risk of the held-out estimator with respect to the Bayes' risk satisfies the following inequality:*

$$R\left(\tilde{\boldsymbol{\mathcal{S}}}, \tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\sigma}}_\varepsilon^2\right) - R^* \leq \inf_{\boldsymbol{\mathcal{S}} \in \boldsymbol{\mathcal{S}}_{DRP}} \left\{ \inf_{\{\boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\} \in \boldsymbol{\Omega}_{\boldsymbol{\alpha},\boldsymbol{\sigma}_\varepsilon^2}} \left(R\left(\boldsymbol{\mathcal{S}}, \boldsymbol{\alpha}, \boldsymbol{\sigma}_\varepsilon^2\right) - R^*\right) + 2\phi_p^1\left(\boldsymbol{\mathcal{S}}\right) + \phi_p^2\left(\boldsymbol{\mathcal{S}}\right) \right\} + \phi_p^2(\tilde{\boldsymbol{\mathcal{S}}}),$$

with probability at least $1 - \delta$, where

$$\phi_p^1\left(\boldsymbol{\mathcal{S}}\right) = n_S \left(L + e^L C\right) \sqrt{\frac{[\![\boldsymbol{\mathcal{S}}]\!] \log 2 + \log\left(2/\delta\right)}{n_1}},$$

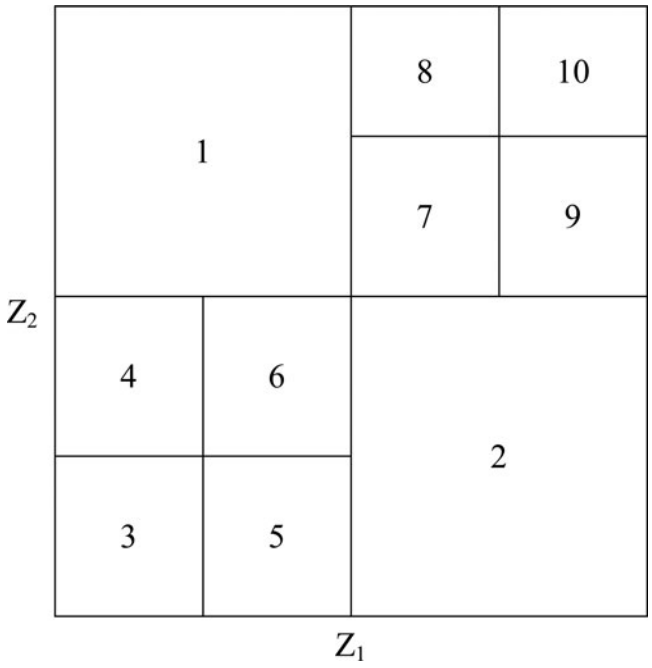

Figure 1. Subdivisions of the partition by environmental variables Z_1 and Z_2.

and

$$\phi_p^2(\boldsymbol{\mathcal{S}}) = n_{\mathrm{S}}\left(L + e^L C\right)\sqrt{\frac{[\![\boldsymbol{\mathcal{S}}]\!]\log 2 + \log(2/\delta)}{n_2}}.$$

5. Simulation studies

In this section, we present the performance of our SPR as a classification and a predictive model, respectively. We focus on the held-out estimator, due to its empirically better performance than the penalized estimator. In what follows, we first describe the data generation process of the environmental, input, and output variables.

We consider five environmental variables that are uniformly distributed on the unit hyper-cube $[0, 1]^5$. We further assume that the first two environmental variables, Z_1 and Z_2, are truly used in partitioning the space of the environmental variables into subdivisions, with the remaining three variables being noise. Specifically, Z_1 and Z_2 partition the space into 10 subdivisions by median splits, as shown in Figure 1. In each subdivision, we consider 100 input variables and generate samples for the input variables from a multivariate normal distribution $N(0, \mathbf{\Sigma}_{100\times100})$. Each element of $\mathbf{\Sigma}_{100\times100}$ is set to be $\sigma_{ij} = 0.5^{|i-j|}$, $i, j = 1, \ldots, 100$, to account for the potential correlation between input variables. To further generate samples for the output variable within each subdivision, we use a linear regression if the output variable is numerical and a multinomial logistic regression if the output variable is categorical. Without loss of generality, we focus on binary output variables in this section. In the linear/logistic regression, we assume that five out of the 100 input variables have non-zero coefficients sampled from $N(0, 1) + 3$, with the remaining input variables having zero coefficients (i.e., they are noise). For generality, we randomly select five input variables to have non-zero coefficients in each subdivision.

Following the afore-described data generation process, we generate 5000 samples to include in a training set and another 5000 samples to include in a held-out validation set. Under this setting, the smallest subdivision includes around 300 samples in the training and validation sets, respectively, which is a limited-sample scenario compared with 100 input variables. Then, we apply the algorithm presented at the end of Section 3 to the data. The result from the algorithm is a partition of the space of the environmental variables and a fitted l_1-regularized linear/logistic regression between the input and output variables within each subdivision of the partition. This entire process from data generation to model fitting is repeatedly run for 100 times. Figure 2 shows the result from one simulation run of the SPR predictive model, in which the partition is represented by a tree whose leaf nodes correspond to the subdivisions of the partition and internal nodes describe the recursive partitioning process. Coefficients of the fitted l_1-regularized regression for each leaf/internal node are represented by a bar chart. Furthermore, Figure 3 stacks up the coefficients of the fitted l_1-regularized regressions for all of the nodes to facilitate comparison across the nodes and discovery of patterns. Additionally, to test the performance of our algorithm under smaller sample sizes, we run another simulation with 2000 samples. Under this setting, the smallest subdivision includes around 120 samples in the training and validation sets, respectively, close to the number of input variables.

Furthermore, for comparison purposes, we apply two competing methods to the same simulation datasets as SPR: a generalized linear model with l_1-regularization (GLM-lasso) and CART. In the GLM-lasso, we include main effects of environmental and input variables as well as the two-way interactions between each environmental variable and each input variable. In CART, we put in all the environmental and input variables and let the CART algorithm decide which variables to use and the interaction structure that best fit the data. To tune the penalty parameter for GLM-lasso and the meta-parameters for CART (i.e., the minimize node size and cost parameter), we perform a grid search over a wide range of the tuning parameters and report the best performance for each method. We apply the three methods on another independently simulated test set consisting of 2000 samples.

Our results are presented as follows: First, we compare the three methods in the accuracy of selecting the environmental variables. We use two metrics: $precision_e$ measures the fraction of environmental variables selected by a method that are the ground-truth partitioning variables and $recall_e$ measures the fraction of the ground-truth partitioning variables that are selected by a method. In GLM-lasso, we consider an environmental variable as "selected" when it is included as either a main effect or an interaction effect. Table 1 shows the $precision_e$ and $recall_e$ under two different samples sizes for the predictive and classification models of SPR in comparison with CART and GLM-lasso.

Then, we examine the variable selection accuracy of SPR in terms of the input variables. This accuracy is computed separately for each subdivision (i.e., leaf node) in Figure 1, as the subdivisions do not have the same set of input variables with non-zero coefficients. We use two common metrics for the accuracy: $precision_I$ measures the fraction of input variables found

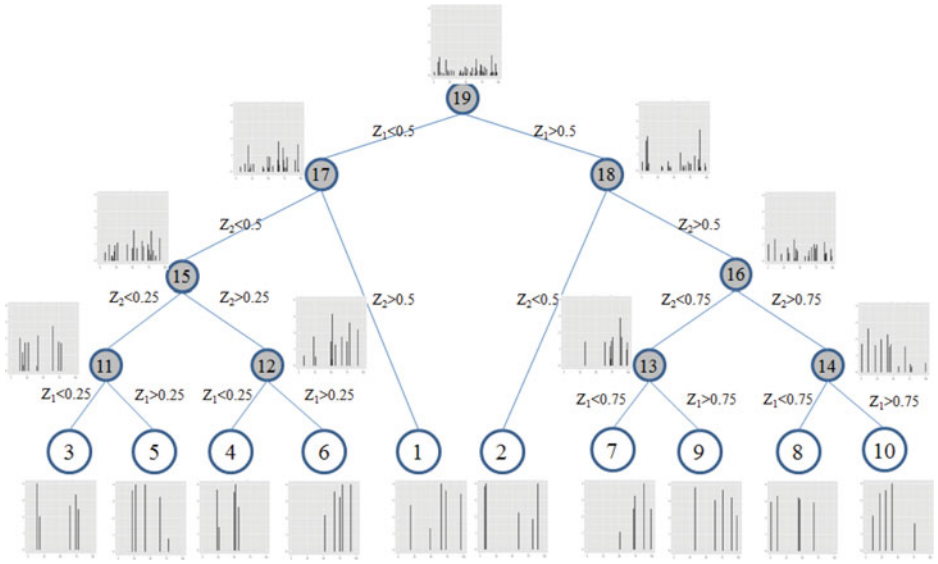

Figure 2. Result from the SPR predictive model for one simulation run (tree represents partition and bar charts represent coefficients of the l_1- regularized regressions).

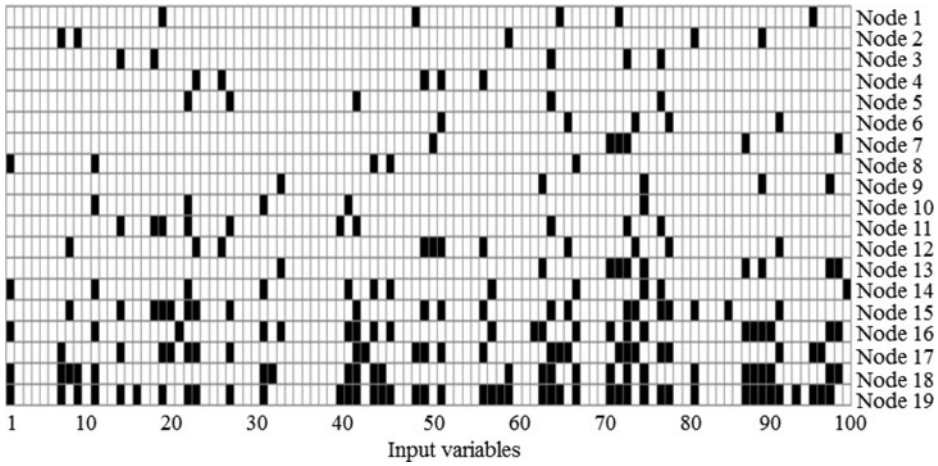

Figure 3. Coefficients of the l_1- regularized regressions fitted for all the nodes in the tree of Figure 2 (black and white represent non-zero and zero coefficients, respectively).

by our algorithm to have non-zero coefficients that truly have non-zero coefficients and $recall_I$ measures the fraction of the input variables with truly non-zero coefficients that are found by our algorithm to have non-zero coefficients. Table 2 shows the subdivision-specific $precision_I$ and $recall_I$ under two different sample sizes for the predictive and classification models.

Next, we present the prediction accuracies of SPR, GLM-lasso, and CART on an independently simulated test set in Table 3. The prediction accuracy for a numerical output variable is measured by a Mean Squared Prediction Error (MSPE) and that for a categorical output variable is measured by the classification accuracy. Additionally, we would like to compare SPR with the VC model (Hastie and Tibshirani, 1993). As VC is computationally very slow, we train it on the simulation data set with 2000 samples and only include the two true environmental variables. The MSPE of

Table 1. Accuracies of SPR, GLM-lasso, and CART in selecting the environmental variables (average (standard deviation) over 100 simulation runs).

	Sample size = 2000		*Sample size = 5000*	
Predictive model	$precision_e$	$recall_e$	$precision_e$	$recall_e$
SPR	0.85 (0.19)	1.00 (0.00)	0.95 (0.12)	1.00 (0.00)
GLM-lasso	0.01 (0.00)	1.00 (0.00)	0.01 (0.00)	1.00 (0.00)
CART	0.18 (0.06)	0.98 (0.14)	0.22 (0.04)	1.00 (0.00)
	Sample size = 2000		*Sample size = 5000*	
Classification model	$precision_e$	$recall_e$	$precision_e$	$recall_e$
SPR	0.81 (0.20)	1.00 (0.00)	0.90 (0.16)	1.00 (0.00)
GLM-lasso	0.08 (0.05)	1.00 (0.00)	0.03 (0.02)	1.00 (0.00)
CART	0.12 (0.07)	0.69 (0.33)	0.16 (0.03)	0.97 (0.12)

Table 2. Accuracy of SPR in selecting the input variables (average (standard deviation) over the simulation runs where the subdivisions in Figure 1 are recovered) for (a) predictive model, (b) classification model.

	Sample size = 2000		*Sample size = 5000*	
Predictive model	$precision_I$	$recall_I$	$precision_I$	$recall_I$
Subdivision 1	0.98 (0.06)	1.00 (0.00)	1.00 (0.02)	1.00 (0.00)
Subdivision 2	0.98 (0.02)	1.00 (0.00)	1.00 (0.02)	1.00 (0.00)
Subdivision 3	0.89 (0.18)	1.00 (0.00)	0.97 (0.07)	1.00 (0.00)
Subdivision 4	0.88 (0.17)	1.00 (0.00)	0.98 (0.06)	1.00 (0.00)
Subdivision 5	0.88 (0.16)	1.00 (0.00)	0.97 (0.08)	1.00 (0.00)
Subdivision 6	0.89 (0.17)	1.00 (0.00)	0.98 (0.08)	1.00 (0.00)
Subdivision 7	0.88 (0.14)	1.00 (0.00)	0.97 (0.08)	1.00 (0.00)
Subdivision 8	0.87 (0.15)	1.00 (0.00)	0.98 (0.07)	1.00 (0.00)
Subdivision 9	0.89 (0.13)	1.00 (0.00)	0.97 (0.07)	1.00 (0.00)
Subdivision 10	0.89 (0.14)	1.00 (0.00)	0.97 (0.07)	1.00 (0.00)
	Sample size = 2000		*Sample size = 5000*	
Classification model	$precision_I$	$recall_I$	$precision_I$	$recall_I$
Subdivision 1	0.94 (0.09)	1.00 (0.00)	0.99 (0.08)	1.00 (0.00)
Subdivision 2	0.94 (0.07)	1.00 (0.00)	0.96 (0.12)	1.00 (0.00)
Subdivision 3	0.84 (0.13)	1.00 (0.00)	0.92 (0.13)	1.00 (0.00)
Subdivision 4	0.86 (0.12)	1.00 (0.00)	0.90 (0.17)	1.00 (0.00)
Subdivision 5	0.87 (0.19)	1.00 (0.00)	0.94 (0.10)	1.00 (0.00)
Subdivision 6	0.90 (0.15)	1.00 (0.00)	0.91 (0.15)	1.00 (0.00)
Subdivision 7	0.84 (0.18)	1.00 (0.00)	0.92 (0.16)	1.00 (0.00)
Subdivision 8	0.90 (0.12)	1.00 (0.00)	0.97 (0.09)	1.00 (0.00)
Subdivision 9	0.87 (0.17)	1.00 (0.00)	0.91 (0.14)	1.00 (0.00)
Subdivision 10	0.84 (0.14)	1.00 (0.00)	0.92 (0.15)	1.00 (0.00)

VC is 77.40, which is substantially higher than the MSPE of SRP (0.09).

Furthermore, we demonstrate the oracle properties of SPR that were discussed in Section 4 in comparison with GLM-lasso and CART. Specifically, the excess risk of each method is computed by taking the difference between the empirical risk of the method and the Bayes' risk. The Bayes' risk is computed from the ground-truth model. Therefore, the smaller the excess risk, the better the oracle property possessed by a method. To compute the empirical risk for GLM-lasso, we follow the paper by Van der Geer (2008). Since both SPR and GLM-lasso use the Negative Log-Likelihood Function (NLLF) as the empirical risk, we would like to use NLLF for CART for consistency. However, the NLLF for CART does not exist. To tackle this problem, we follow a similar idea to that by Friedman and Popescu (2008) and convert the tree trained by CART into an empirical regression that includes the nodes of the tree as categorical predictors. The NLLF for the empirical regression is then computed to represent the empirical risk of CART. The results are summarized in Table 4.

Also, to compare the computational efficiency of the different methods, we record the runtimes of model training by SPR, GLM-lasso, and CART, respectively, for each experiment performed in this section. On average, the runtimes for SPR, GLM-lasso, and CART are 8.82, 11.63, and 15.50 seconds, respectively.

Finally, we summarize our observations from the results:

1. SPR achieves high precision and recall in selecting the environmental variables (Table 1) and the input variables (Table 2). A smaller sample size slightly affects the precision but not the recall.
2. The precision in selecting the input variables varies across the subdivisions (Table 2). Specifically, subdivisions 1 and 2 have the highest precision, whereas the other subdivisions have slightly lower precisions. This is because subdivisions 1 and 2 have the largest sample size.
3. In comparison with the competing methods, SPR has significantly higher precision in selecting the environmental variables and prediction accuracy than GLM-lasso and CART (Tables 1 and 3). SPR also significantly outperforms VC in prediction accuracy. GLM-lasso performs worse because it uses a single model to fit all of the data, which are known to be a mixture of 10 different distributions. CART performs worse, which is somewhat surprising as CART is known to be a flexible approach for modeling complex variable relationships with good performance. Its underperformance may be a result of it not respecting the (generalized) linear relationship between the input and output variables within each subdivision. VC performs worse because its smoothness assumption for the input/output relationship across adjacent subdivisions of the partition does not hold in our simulation settings.

Table 3. Prediction accuracy of SPR in comparison with GLM-lasso and CART.

	Sample size = 2000			*Sample size = 5000*		
	SPR	*GLM-lasso*	*CART*	*SPR*	*GLM-lasso*	*CART*
Predictive model (MSPE)	2.94	86.48	119.05	0.09	81.89	115.21
Classification model (classification accuracy)	0.90	0.68	0.62	0.94	0.70	0.62

Table 4. Oracle properties of SPR in comparison with GLM-lasso and CART.

	Sample size = 2000			*Sample size = 5000*		
Predictive model	*SPR*	*GLM-lasso*	*CART*	*SPR*	*GLM-lasso*	*CART*
Bayes' risk		0.72			0.73	
Empirical risk	0.78	3.78	4.38	0.74	3.76	4.25
Excess risk	0.06	3.06	3.66	0.01	3.03	3.52
	sample size = 2000			*sample size = 5000*		
Classification model	*SPR*	*GLM-lasso*	*CART*	*SPR*	*GLM-lasso*	*CART*
Bayes' risk		0.12			0.11	
Empirical risk	0.30	0.64	0.68	0.14	0.62	0.68
Excess risk	0.18	0.52	0.56	0.03	0.51	0.57

4. Table 4 shows that SPR has a substantially smaller empirical/excess risk than GLM-lasso and CART under a fixed sample size. When the sample size increases from 2000 to 5000, the excess risk for SPR shrinks dramatically while those for GLM-lasso and CART show little change. These provide evidence that SPR has a better oracle property.
5. In terms of computational efficiency, SPR on average only needs 76% and 57% of the runtimes by GLM-lasso and CART to complete model training, respectively.

Lastly, in this section, we would like to discuss the pattern of the recursive partitioning process of SPR, as revealed by Figures 2 and 3. The pattern holds across all simulation runs. Specifically, we observe that the regression fitted at the root node is the densest (i.e., having the most non-zero coefficients). As the recursive partitioning proceeds, the fitted regressions become sparser and sparser. Eventually, at the leaf nodes, the sparsest regressions are obtained, which are consistent with the ground truth that each subdivision has only five out of 100 input variables with non-zero coefficients. The reason for this trend is that the data used to fit a regression at an earlier stage of the recursive partitioning (i.e., closer to the root node) are more mixed with different distributions. Therefore, more input variables with non-zero coefficients are needed to fit the data. Even so, the fitting is still not good, which keeps the recursive partitioning going until reaching the leaf nodes. Another piece of evidence of the less adequate fitting of the regression at earlier stages of the partitioning is that the magnitude of non-zero coefficients is generally smaller than that of the leaf nodes. All of these support the necessity and adequacy of the recursive partitioning in SPR.

6. Application

In this section, we present an application of using building design variables, outdoor environmental variables, and their interactions to predict building energy consumption. To obtain relevant data, we use *EnergyPlus*, a building energy simulation platform developed by the DOE (https://energyplus.net/). DOE developed *EnergyPlus* with a goal of making substantial progress toward improving energy efficiency for commercial and residential buildings in the United States. Since its establishment, *EnergyPlus* has been used by numerous researchers, engineers, and architects to model energy consumption in various types of buildings. In fact, 16 building types can be simulated by *EnergyPlus*. In this study, we focus on "Big Offices," which is the most prevalent building type for which *EnergyPlus* has been used.

Specifically, based on domain knowledge and existing literature (Eisenhower *et al.*, 2012), we choose to include 13 input variables on the operational features of Big Offices that potentially affect building energy consumption. We further include six outdoor environmental variables, among which there is one categorical variable of building locations. Six locations across different climate zones of the United States are included: San Francisco (CA), Boulder (CO), Phoenix (AZ), Houston (TX), Miami (FL), and Baltimore (MD). The output variable is building energy consumption. Abbreviations and physical meanings of all of the variables are given in Table 5. We run *EnergyPlus* and generate a dataset of one month (July) with a sampling frequency of every 30 minutes, which results in a total of 8922 samples.

Next, we apply SPR to the dataset. Since building energy consumption is a numerical variable, we apply the predictive model in our method. We choose to use the held-out estimator for the predictive model due to its empirically better performance than the penalized estimator. The entire data set is split into a training set (first 11 days of data), a held-out validation set (next 10 days of data), and a test set (last 10 days of data). Figures 4 and 5 show the partition found by SPR and coefficients of the fitted l_1-regularized regression in each subdivision of the partition. Four out of the six environmental variables are used in the recursive partitioning, including loc, out_temp, out_airflow_frac, and solar_direct. Loc is used as the first variable to start the partitioning, which indicates that it helps the most on lowering the prediction error among all the environmental variables. The grouping of loc to the left and right branches makes sense, as the right includes cities with high temperature or/and humidity that are known factors to significantly affect building energy consumption, while the left branch includes cities with different characteristics. The right branch further splits into $loc = \{MIA, IAH\}$, two high-temperature, high-humidity cities, and $loc = \{PHX\}$, a high-temperature, low-humidity city. Moreover, within the same location $loc = \{PHX\}$, this is the solar_direct that affects the input/output relationship of buildings. Specifically, a close examination of the last two rows of Figure 4 shows that, compared with the regression model fitted at the high level solar_direct (>492.5 W/m^2), two additional building operational variables (air_re_hum and window_rad) are selected to predict energy consumption at the

Table 5. Abbreviations and physical meanings of input, environmental, and output variables in building energy consumption modeling.

	Variable abbreviation (unit)	*Physical meaning*
Input variables	people_occ	Total number of people within the building zone
	air_temp (°C)	Indoor air temperature
	air_re_hum (%)	Indoor air relative humidity
	therm_set_temp (°C)	Thermostat cooling setpoint temperature
	equip_sch	Building equipment schedule; 0 and 1 for equipment off and on, respectively
	window_rad (W)	Window total transmitted solar radiation rate
	window_sunlit_frac	Fraction of window surface illuminated by unreflected beam solar radiation
	window_heat_gain (W)	Surface window heat gain rate
	window_heat_loss (W)	Surface window heat loss rate
	water_tank_temp (°C)	Water heater tank temperature
	heat_coil (W)	Average total heating capacity provide by heat pump
	cool_coil (W)	Average total cooling load provided by heat pump
	people_air_temp (°C)	Thermal comfort temperature that determines the balance between people heat gain and loss
Environmental variables	out_temp (°C)	Outdoor air drybulb temperature
	out_re_hum (%)	Outdoor air relative humidity
	out_airflow_frac	Outdoor air flow fraction
	solar_diffuse (W/m^2)	Diffuse solar radiation rate
	solar_direct (W/m^2)	Direct solar radiation rate
	loc: {SFO, BLD, BWI, MIA, IAH, PHX}	Airport codes of the six cities
Output variable	electricity (kw)	Electricity consumption

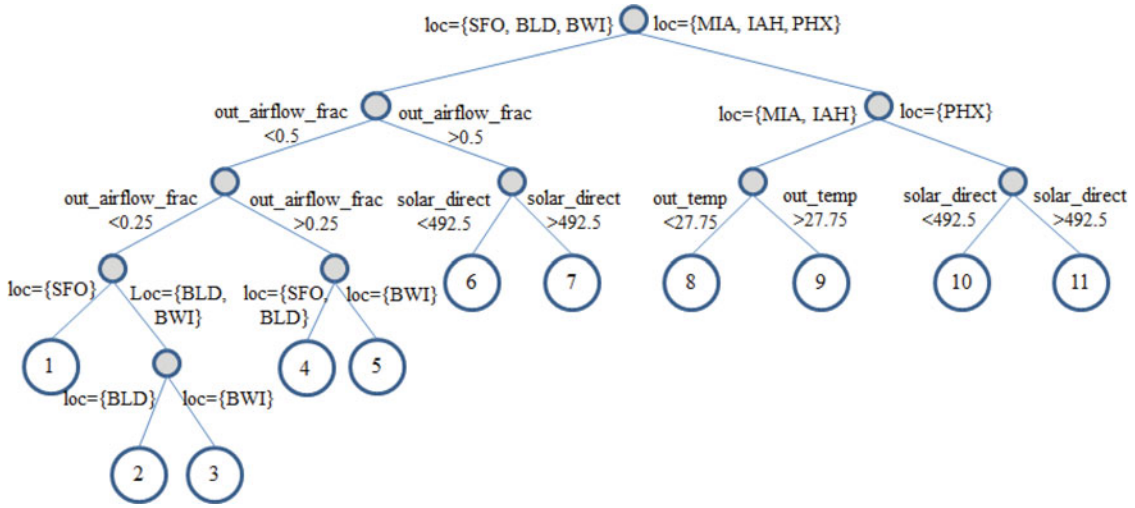

Figure 4. Partition of the space of environmental variables found by SPR.

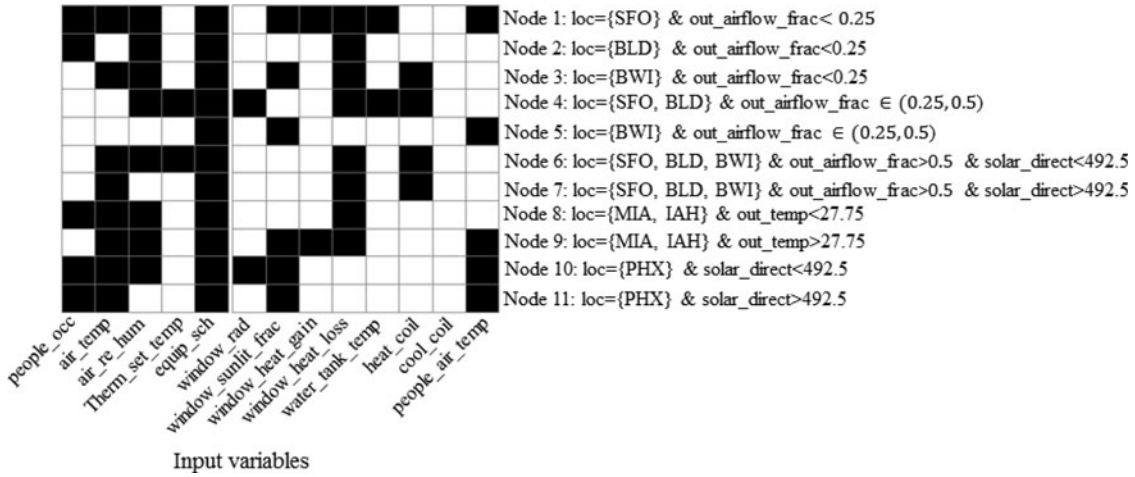

Figure 5. Zero (white) and non-zero (black) coefficients of the fitted l_1-l_1-regularized regression in each subdivision of the partition in Figure 4.

lower-level of solar_direct (<492.5 W/m^2). This makes sense, as when the outdoor direct solar radiation rate is low, building energy consumption is sensitive to how much of the radiation can be transmitted to indoors by windows (window_rad) and the indoor air humidity (air_re_hum). When the outdoor direct solar radiation rate is high, its effect on building energy consumption tends to be more dominant and thus makes window_rad and air_re_hum less important.

Furthermore, we examine the left branch of the root node, which is further split by out_airflow_frac. Outdoor airflow rate affects indoor air circulation. The interaction effect of indoor air circulation and other building variables, such as temperature and humidity, on energy consumption is well known. For example, for two buildings to achieve the same indoor temperature and humidity, the one with a lower level of air circulation typically needs to consume more electricity. After splitting by out_airflow_frac, the left branch is further split by solar_direct and loc, which are also variables used in the right branch.

In addition, we compare the regressions fitted for the 11 leaf nodes (subdivisions). The input variables selected by a majority of the regressions include equip_sch (selected by 11/11), air_temp (8/11), air_re_hum (8/11), window_heat_loss (8/11), and window_sunlit_frac (6/11), which are well-known factors that affect building energy consumption. In particular, equip_sch is found to be the only globally significant input variable that affects building energy consumption. From the practical point of view, the existence of a globally significant input variable, such as equip_sch, is an advantage, as it means that equip_sch is a robust input variable against the environment. That is, by adjusting equip_sch, we have a chance to change the electricity consumption regardless of where the building is located. On the other hand, an input variable such as heat_coil is not a globally significant input variable. By adjusting it, we can change the electricity consumption of some subdivisions such as 3, 4, 6, 7 but not others.

Moreover, there are no two regressions using the same set of input variables to predict the output. This finding is important for building energy management. Specifically, it suggests that how to adjust building design and operational variables (including what variables to adjust and how much to adjust) in order to reduce energy consumption should consider the environmental condition the building is operated under, especially that characterized by loc, out_airflow_frac, out_temp, and solar_direct. Just like "no treatment fits all" in personalized medicine, there is no energy management strategy that is universally applicable to all buildings even of the same type (Large Offices in this application). On the other hand, unlike personalized medicine, in which the precision of treatment needs to be down to the level of individual patients, building energy management can be performed at a much coarser granularity. Not every building needs a different "treatment"; buildings within a certain range of the combinatorial environmental variables can be managed in the same way. By using SPR, ranges of this kind can be automatically identified. The existence of these ranges is further supported by the superior prediction accuracy of SPR, which will be presented next.

Finally, for the purpose of comparison, we employ two competing methods, GLM-lasso and CART, on the same data set. Since the output variable is numerical, the GLM-lasso is a lasso model and the CART is a regression tree, both of which use all input and environmental variables as predictors. We compute the MSPEs of the three methods on the test set, which are 5123.99, 11 875.65, and 8802.35 for SPR, GLM-lasso, and CART, respectively. SPR has a significantly better prediction accuracy.

7. Conclusions

In this article, we developed an SPR for modeling the nonlinear interaction between a system and the multivariate environment under which it operates. We proposed a penalized estimator and a held-out estimator for the SPR, analyzed theoretical properties of the estimators, and developed a recursive partitioning algorithm for model estimation. We conducted extensive simulation experiments to demonstrate the better performance of SPR compared with GLM-lasso and CART. Finally, an application of building energy prediction and management was presented. Extending from this research, there are abundant future directions. Immediate extensions include the use of other sparsity-induced regularizations than the l_1-regularization to account for various structures of the input variables specified by domain knowledge, modeling of multiple correlated output variables, and fitting of nonlinear models between the input and output variables. Extensions that may need more substantial amounts of effort include design of ensemble methods similar to bagging, boosting, and random forest to reduce the variability of the recursive partitioning and development of optimization algorithms to search for the partition with a better optimality property. Both the SPR and its extensions have broad applicability to domains beyond building energy management, including but not limited to mobile communication networks and wind energy as presented in the Introduction, as well as bioinformatics in studying gene–environment interactions related to diseases or disease traits.

Funding

This work is partly supported by the National Science Foundation under grants CMMI-1069246 and CMMI-1149602.

Notes on contributors

Shuluo Ning is a Ph.D. candidate in Industrial Engineering at Arizona State University. He received his B.S. in Industrial Engineering from Nankai University and an M.S. in Industrial Engineering from Ohio University in 2010 and 2012, respectively. His research interests are statistical modeling and machine learning with applications in health care and building energy.

Eunshin Byon is an Assistant Professor with the Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor. She received her Ph.D. degree in Industrial and Systems Engineering from Texas A&M University, College Station, in 2010. Her research interests include data analytics, quality and reliability engineering, system infomatics and uncertainty quantification. She is a member of IIE, INFORMS, IEEE, and ASQ.

Teresa Wu is a Professor of Industrial Engineering School of Computing, Informatics, Decision Systems Engineering at Arizona State University. She received her Ph.D. in Industrial Engineering from the University of Iowa in 2001. Her current research interests include swarm intelligence, distributed decision support, distributed information system, and health informatics. She has published more than 80 journal articles in journals such as *IEEE Transactions on Evolutionary Computation*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and *Information Science*. She is currently serving as the editor-in-chief for *IIE Transactions on Healthcare Systems Engineering*.

Jing Li is an Associate Professor in Industrial Engineering at Arizona State University. She received her B.S. from Tsinghua University in China and an M.A. in Statistics and a Ph.D. in Industrial and Operations Engineering from the University of Michigan in 2005 and 2007, respectively. Her research interests are applied statistics, data mining, and quality and systems engineering. She is a recipient of an NSF CAREER award. She is a member of IIE, INFORMS, and ASQ.

References

Bien, J., Taylor, J. and Tibshirani, R. (2013) A lasso for hierarchical interactions. *The Annals of Statistics*, **41**(3), 1111–1141.

Breiman, L. (1995) Better subset regression using the nonnegative garrote. *Technometrics*, **37**(4), 373–384.

Breiman, L., Friedman, J., Stone, C.J. and Olshen, R.A. (1984) *Classification and Regression Trees*, CRC Press, Boca Raton, FL.

Byon, E., Choe, Y. and Yampikulsakul, N. (2015) Adaptive learning in time-variant processes with application to wind power systems. *IEEE Transactions on Automation Science and Engineering*, **99**, 1–11.

Cai, Z., Fan, J. and Li, R. (1999) Generalized varying-coefficient models. Report, Department of Statistics, UCLA, Los Angeles, CA.

Chiang, C.T., Rice, J.A. and Wu, C.O. (2001) Smoothing spline estimation for varying coefficient models with repeatedly measured dependent variables. *Journal of the American Statistical Association*, **96**(454), 605–619.

Chipman, H. (1996) Bayesian variable selection with related predictors. *Canadian Journal of Statistics*, **24**(1), 17–36.

Choi, N.H., Li, W. and Zhu, J. (2010) Variable selection with the strong heredity constraint and its oracle property. *Journal of the American Statistical Association*, **105**(489), 354–364.

Eisenhower, B., O'Neill, Z., Narayanan, S., Fonoberov, V.A. and Mezić, I. (2012) A methodology for meta-model based optimization in building energy models. *Energy and Buildings*, **47**, 292–301.

Fan, J. and Li, R. (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, **96**(456), 1348–1360.

Fan, J., Yao, Q. and Cai, Z. (2003) Adaptive varying coefficient linear models. *Journal of the Royal Statistical Society: Series B* (*Statistical Methodology*), **65**(1), 57–80.

Fan, J. and Zhang, W. (1999) Statistical estimation in varying coefficient models. *Annals of Statistics*, **27**(5), 1491–1518.

Friedman, J.H. and Popescu, B.E. (2008) Predictive learning via rule ensembles. *The Annals of Applied Statistics*, **2**(3), 916–954.

Hamada, M. and Wu, C.J. (1992) Analysis of designed experiments with complex aliasing. *Journal of Quality Technology*, **24**(3), 130–137.

Hardy, W.C. (2001) *QoS: Measurement and Evaluation of Telecommunications Quality of Service*, John Wiley & Sons, Hoboken, NJ.

Hastie, T. and Tibshirani, R. (1993) Varying-coefficient models. *Journal of the Royal Statistical Society. Series B* (*Methodological*), **55**(4), 757–796.

Hoover, D.R., Rice, J.A., Wu, C.O. and Yang, L.P. (1998) Nonparametric smoothing estimates of time-varying coefficient models with longitudinal data. *Biometrika*, **85**(4), 809–822.

Hu, T. and Xia, Y. (2012) Adaptive semi-varying coefficient model selection. *Statistica Sinica*, **22**, 575–599.

Montgomery, D.C., Peck, E.A. and Vining, G.G. (2015) *Introduction to Linear Regression Analysis*, John Wiley & Sons, Hoboken, NJ.

Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B* (*Methodological*), **58**(1), 267–288.

Van de Geer, S.A. (2008) High-dimensional generalized linear models and the lasso. *The Annals of Statistics*, **36**(2), 614–645.

Vapnik, V. (1998) *Statistical learning theory*, John Wiley & Sons, Hoboken, NJ.

Xia, Y. and Li, W.K. (1999) On the estimation and testing of functional-coefficient linear models. *Statistica Sinica*, **9**, 735–757.

Yuan, M., Joseph, V.R. and Zou, H. (2009) Structured variable selection and estimation. *The Annals of Applied Statistics*, **3**(4), 1738–1757.

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

Zhang, W., Lee, S.Y. and Song, X. (2002) Local polynomial fitting in semi-varying coefficient model. *Journal of Multivariate Analysis*, **82**(1), 166–188.

Zhao, P., Rocha, G. and Yu, B. (2009) The composite absolute penalties family for grouped and hierarchical variable selection. *The Annals of Statistics*, **37**(6), 3468–3497.

Zhou, X. and You, J. (2004) Wavelet estimation in varying-coefficient partially linear regression models. *Statistics & Probability Letters*, **68**(1), 91–104.

Zou, H. (2006) The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, **101**(476), 1418–1429.

Zou, H. and Hastie, T. (2005) Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B* (*Statistical Methodology*), **67**(2), 301–320.