Automatic effect selection for Generalized Additive Models

Collarin Claudia, Fasiolo Matteo & Agostinelli Claudio

Statistical methods and models for complex data — Poster Session: 21 September 2022

Introduction

Generalized Additive Models allow for flexible specification of dependence of the response on the covariates by defining the model in terms of smooth functions: the **effects**.

- The introduction of an algorithm which automatically selects main effects and their interactions would ensure interpretability and high accuracy.
- Gradient boosting yields an additive model whose terms are fitted in a stagewise fashion.

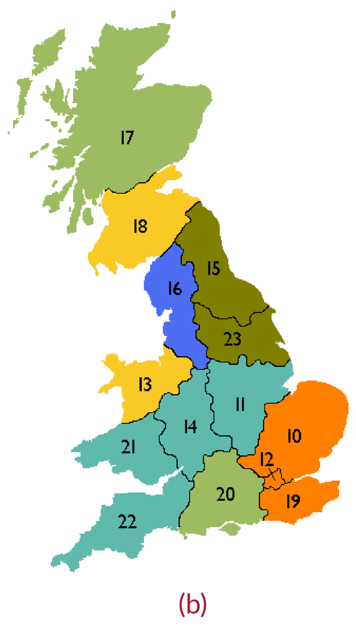
Here we propose a **four–step algorithm** which combines **Gradient boosting (**L_2**Boost)** and **Lasso**. It automatically selects up to two–dimensional effects for GAMs.
To ensure parsimony, **ANOVA decomposition of the model** into main effects and first order interactions is exploited.

Motivating application

The **electricity net-load** in Great Britain is the load on the transmission system measured at the Grid Supply Points (GSP groups).

GSP group	ID	Name
A	10	East England
C	12	London
P	17	North Scotland
L	22	South West England

(a)



(b)

Table 1a & Figure 1b: Some GSP group IDs with corresponding distribution areas in Great Britain distribution network system, and the map of GSP groups.

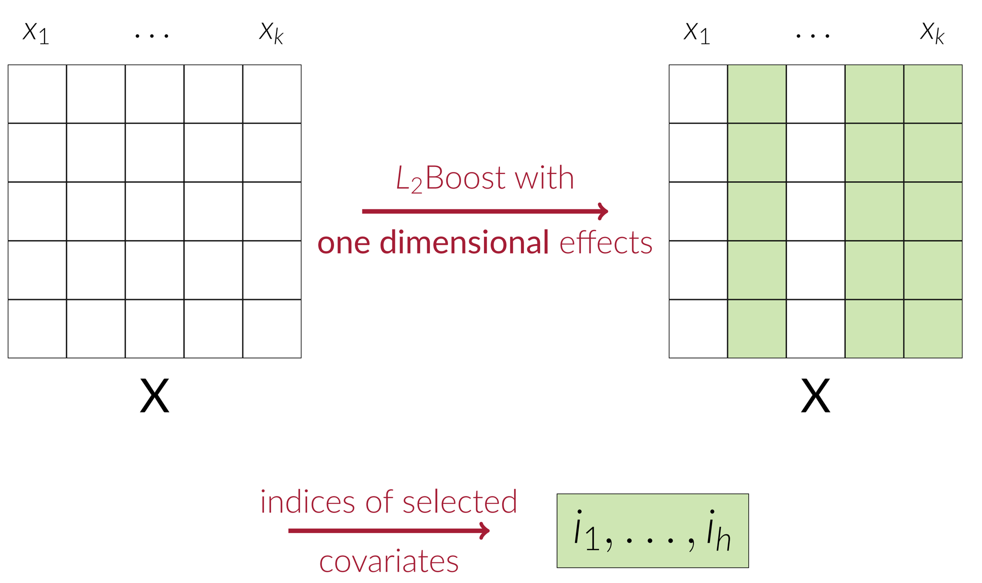
- Covariates are related to type of the day, weather and price on n2ex market;
- Each group exhibits a diversity of embedded wind and solar capacities relative to load;
- Thus, different GAM models should be used to predict **net-load** in each area;
- Data are avaiable in Browell (2021).

The proposed algorithm

- $\mathbf{X}$ the $n \times k$ matrix containing n observations of k covariates;
- We want to fit the response variable $\mathbf{y}$ with a GAM model whose effects are either **one** or **two dimensional**.

Step 1: Gradient Boosting

- Assume no interaction among covariates and fit $\mathbf{y}$ with a GAM model containing **only one–dimensional** effects, $f_1, \dots, f_k$;
- Perform L_2Boost with **pre–specified base–learners** for each covariate;
- Store the h covariates that have been selected by Gradient Boosting.

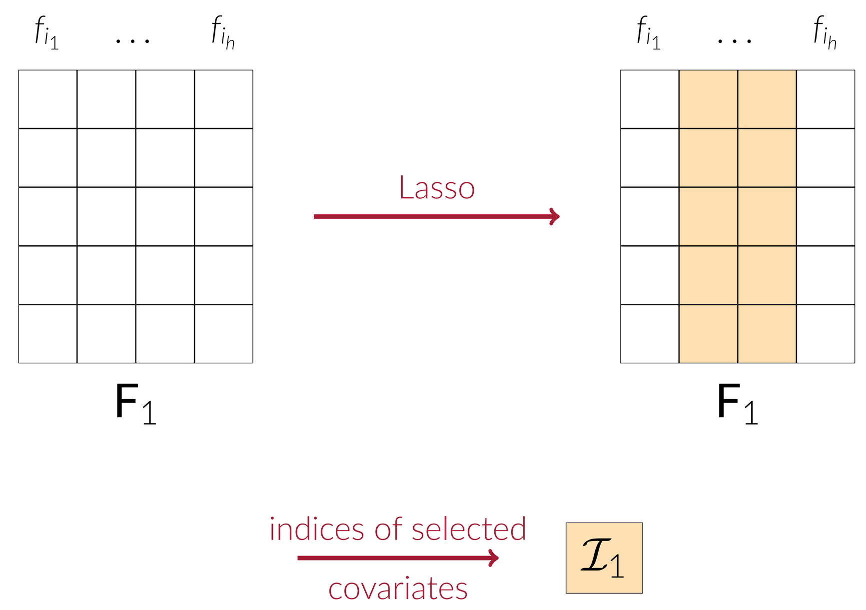


Step 2: Lasso on one dimensional effects

- $\mathbf{F}_1 = (\hat{f}_{i_1}(\mathbf{x}_{i_1}), \dots, \hat{f}_{i_h}(\mathbf{x}_{i_h}))$ the $n \times h$ matrix whose columns are the fitted effects in the previous step;
- Let $\mathbf{j}_1 = (1, \dots, 1)^\top$ be a h–dimensional vector of 1s;
- The fitted response variable can be equivalently written as

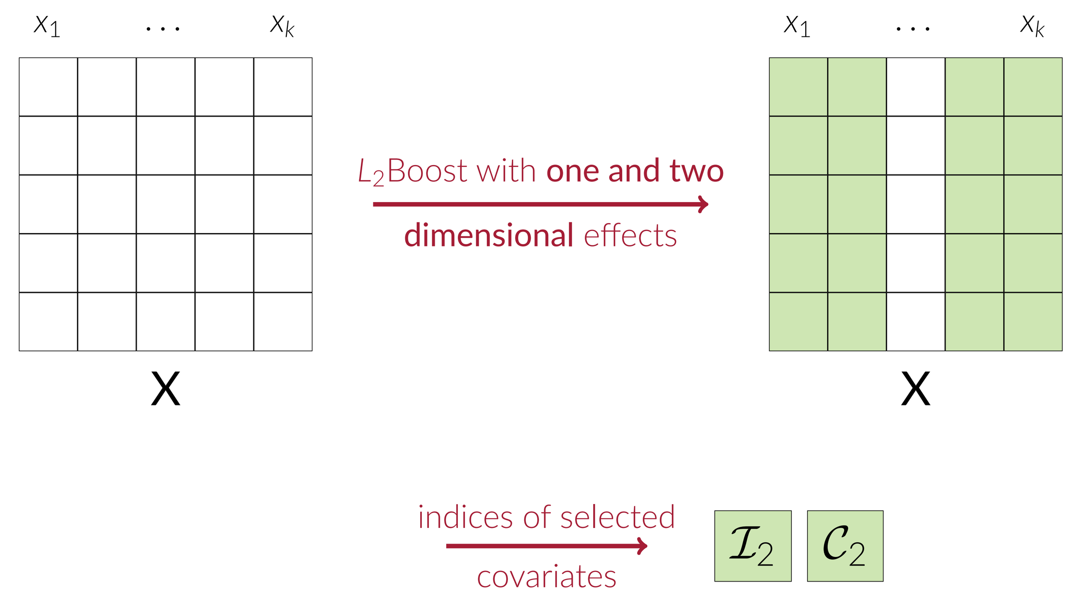
$$\hat{\mathbf{y}} = \mathbf{F}_1 \mathbf{j}_1. \tag{1}$$

- Backward eliminate covariates with **Lasso** applied to (1) using $\mathbf{F}_1$ as model matrix;
- $\mathcal{I}_1$ be the set of covariate indices selected by Lasso.



Step 3: Gradient Boosting with two dimensional effects

- $\mathcal{C}_1 = \{(i,j) \mid i,j \in \mathcal{I}_1,\ i < j\}$ the set of unique combinations of indices in $\mathcal{I}_1$;
- Introduce additional base–learners to fit interactions between variable pairs in $\mathcal{C}$;
- Perform L_2Boost with both 1–dimensional effects in step 1 and 2–dimensional effects as defined in this step;
- Store in $\mathcal{I}_2$ the indices of 1–dimensional effects, in $\mathcal{C}_2$ the pairs of indices of 2–dimensional effects.

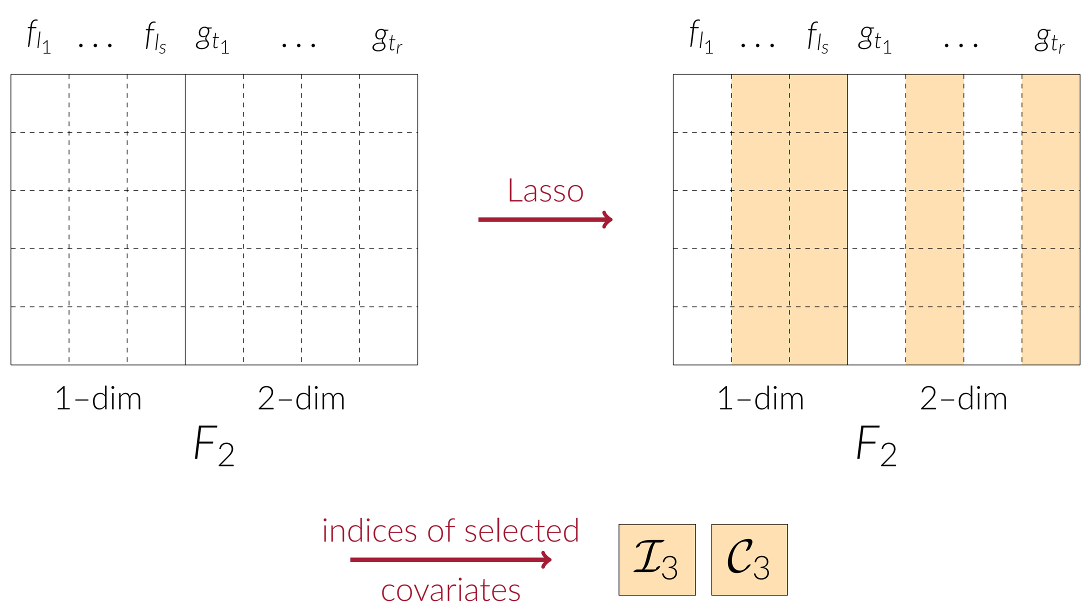


Step 4: Lasso on one and two–dimensional effects

- $\mathbf{F}_2 = (\hat{f}_{l_1}(\mathbf{x}_{l_1}), \dots, \hat{f}_{l_s}(\mathbf{x}_{l_s}), \hat{g}_{t_1}(\mathbf{x}_{t_1}), \dots \hat{g}_{t_r}(\mathbf{x}_{t_r}))$ with $l_i \in \mathcal{I}_2$ and $t_i \in \mathcal{C}_2$ denote the $n \times (s + r)$ matrix whose columns are the fitted effects in the previous step;
- $\mathbf{j}_2 = (1, \dots, 1)^\top$ a $(s + r)$–dimensional vector of 1s;
- The fitted response variable can be written as

$$\hat{\mathbf{y}} = \mathbf{F}_2 \mathbf{j}_2; \tag{2}$$

- Backward eliminate covariates with **Lasso** applied to (2) using $\mathbf{F}_2$ as model matrix;
- Let $\mathcal{I}_3$, $\mathcal{C}_3$ denote the sets of covariate indexes and pair of them selected by Lasso.



Simulation study

Figure 2 shows the results of the simulation study with 70 samples made of 2000 independent observations in train set and 2000 in test set.
Each model sample is generated by

$$y_i = \sum_{j=1}^{8} f_j(x_{i,j}) + f(x_{i,2}, x_{i,3}) + \varepsilon_i, \qquad i = 1, \dots, 2000$$

where $x_i \sim U([0,1])$, $\varepsilon \sim N(0, \sigma^2)$. Additional 20 noise covariates have been added to each observation. Their values were uniformly generated in the interval $[-0.5, 0.5]$.
Thin-plate splines are used as base–learners.

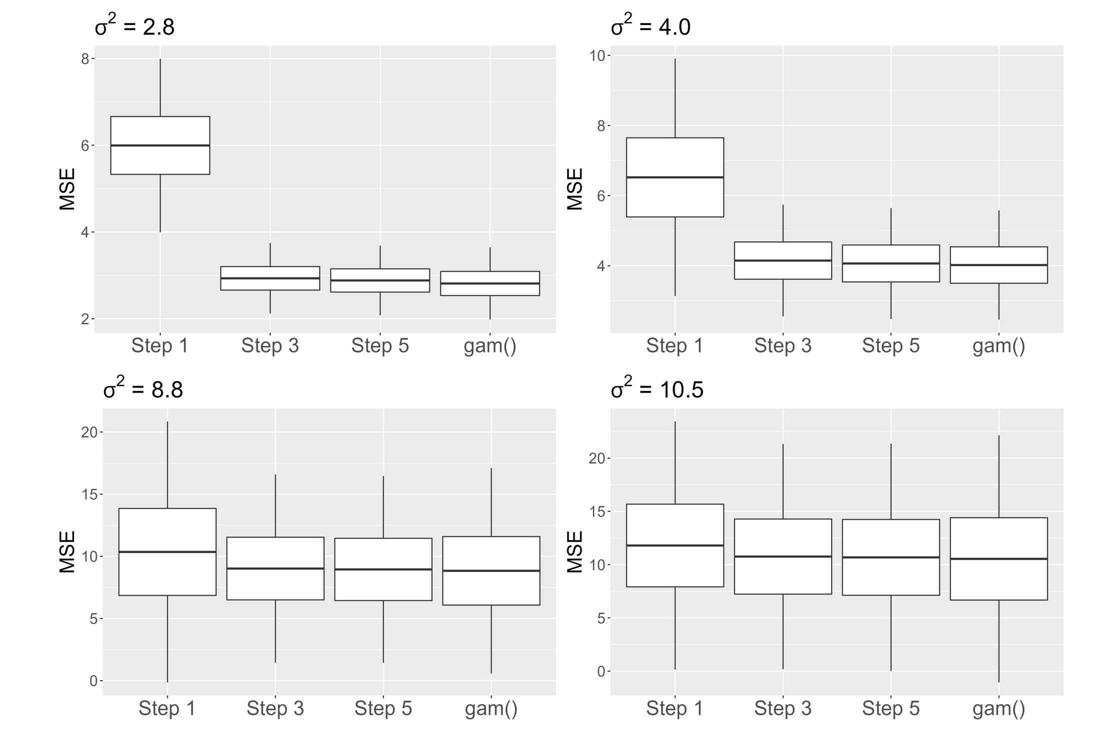


Figure 2. Mean squared errors applying the algorithm to simulated data. σ^2 is the variance of the Gaussian noise. Step 1 and 3 are the mean squared error along the algorithm, step 5 is obtained by fitting a GAM model with selected effects. The last boxplot corresponds to the error obtained from the `gam()` function of `mgcv` package.

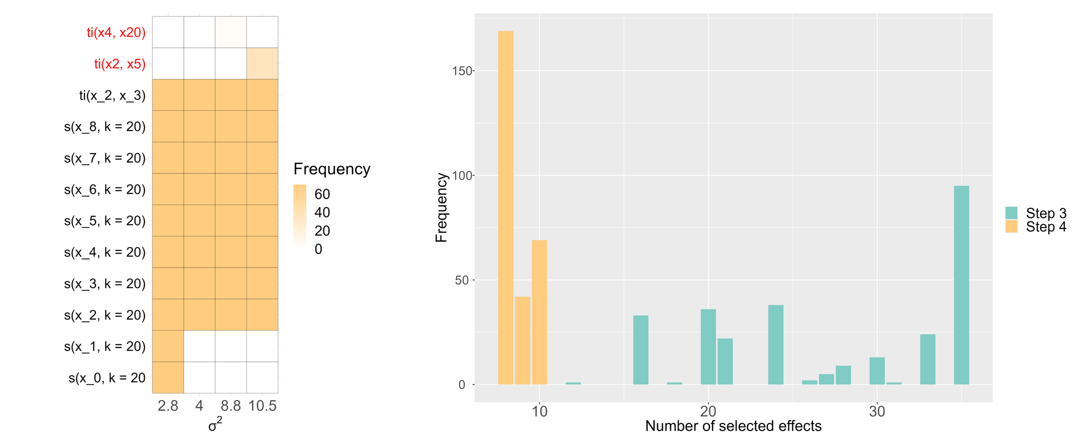


Figure 3. From left to right: the frequency selection of each effect, varying the noise variance (the first two rows are the wrong effects selected); a comparison between the number of effects selected by gradient boosting in step 3 and by Lasso in step 4.

Electricity net-load data analysis

The data set spans from 2nd January, 2014 to 31th December, 2018 with half-hourly resolution, determining **91726 observations for each GSP group**.
Measurements collected in 2018 have been used as test set.
Remaining data were split into **3 folds** to perform cross validation to fit models in the gradient boosting steps.
Thin–plate splines were used as base learners.

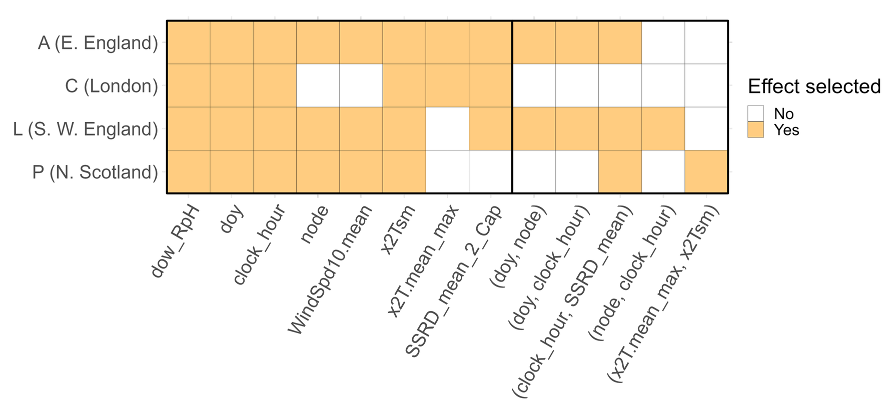


Figure 4. Effects selected applying the algorithm to Electricity net-load data set.

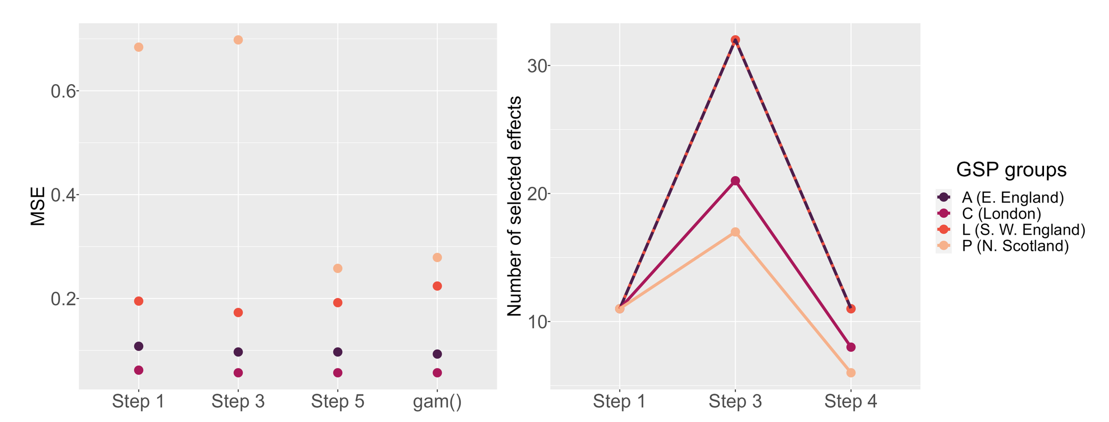


Figure 5. From left to right: the mean squared errors computed along the steps of the algorithm and with the `gam()` function of `mgcv` package; the number of selected effects in step 1, 3 and 4.

References

Browell, J. (2021). Supplementary Material for "Probabilistic Forecasting of Regional Net-load with Conditional Extremes and Gridded NWP" [Data set]. *Zenodo*. https://doi.org/10.5281/zenodo.5031704

Browell, J. and Fasiolo, M. (2021). Probabilistic forecasting of regional net-load with conditional extremes and gridded nwp. *IEEE Transactions on Smart Grid*, **12 (6)**: 5011 – 5019.

Trevor Hastie. Robert Tibshirani. (1986). Generalized Additive Models. *Statistical Science*, **1 (3)**, 297 – 310.

Contact information

Collarin Claudia, Ph.D. student
University of Padua
claudia.collarin@phd.unipd.it
linkedin.com/in/claudia-collarin-5b60291b8