MOX-Report No. 43/2018

Performing Learning Analytics via Generalized Mixed-Effects Trees

Fontana, L.; Masci, C.; Ieva, F.; Paganoni, A.M.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it http://mox.polimi.it

Performing Learning Analytics via Generalized Mixed-Effects Trees

Luca Fontana, Chiara Masci, Francesca Ieva, Anna Maria Paganoni

July 25, 2018

MOX - Modelling and Scientific Computing, Department of Mathematics,
Politecnico di Milano, via Bonardi 9, Milano, Italy
luca11.fontana@mail.polimi.it
chiara.masci@polimi.it
francesca.ieva@polimi.it
anna.paganoni@polimi.it

Abstract

Nowadays, the importance of Educational Data Mining and Learning Analytics in higher education institutions is increasingly recognized. The analysis of university careers and of student dropout prediction is one of the most studied topics in the area of Learning Analytics. In the perspective of modeling the student dropout, we propose an innovative statistical method, that is a generalization of mixed-effects trees for a response variable in the exponential family: Generalized Mixed-Effects Trees (GMET). We perform a simulation study in order to validate the performance of our proposed method and to compare GMET to classical models. In the case study, we apply GMET to model Bachelor student dropout in different degree programmes of Politecnico di Milano. The model is able to identify discriminating student characteristics and estimate the degree programme effect on the probability of student dropout.

Keywords: Mixed-effects models; Regression and Classification trees; Student dropout; Academic data.

1 Introduction

The present work is part of the international SPEET project (Student Profile for Enhancing Engineering Tutoring), an ERASMUS$^+$ project aiming to open a new perspective to university tutoring systems. It intends to extract useful information from academic data provided by its partners[1] and to identify different Engineering students profiles across Europe [21]. Here, our goal is to find out which indicators may discriminate between two different student profiles: *dropout* students, who permanently finish their career for any reason other than the achievement of the Bachelor of Science (BSc) degree, and *graduate* students, who complete their career with the achievement of academic qualification. This choice is motivated by the fact that, across all SPEET partners, almost a student out of two leaves his/her Engineering studies before obtaining the BSc degree. If it was possible to know as soon as possible to which profile a student belongs, it would be of valuable help for tutors to improve counseling actions.

Data provided by universities usually includes indicators about the socio-economic background and both current and previous performance of the students. However, academic success depends on different factors, both internal and external [2]. The dataset we use in our analysis includes more than 18,000 BSc careers from Politecnico di Milano: it essentially consists of student record data, so it just partially covers these factors. Similar dataset structures have already been used in recent developments oriented to the prediction of performance and detection of dropouts or students at risk [20]. The hypothesis is that both background and career indicators are enough to identify the students at risk and to draw the attention of tutors, who should complete the student profile with further information.

In our situation, students are naturally nested within the degree programme they are attending. In addition, further levels of hierarchy are possibile, such as programmes within faculties, faculties within universities and finally universities within countries. While investigating the learning process, it is necessary to disentangle the effects given by each level of hierarchy [4]. Indeed, if the clustered aspect of the data is not inspected, it may result in a loss of likely valuable information. Multilevel models take into account the hierarchical nature of data and are able to quantify the portion of variability in the response variable that is attributable to each level of grouping [9]. Generalized Linear Mixed Models (GLMM) fit a multilevel model on a binary response variable, but they impose a linear effect of covariates on a transformation of the response variable [1]. On the contrary, tree-based methods such as the CART model learn the relationship between the response and the predictors by identifying dominant patterns in the training data [5]. In addition, these methods allow a clear graphical representation of the results that is easy to communicate. The goal of our work is to propose a novel method able to preserve the flexibility of the CART model and to extend it to a clustered data structure, where multiple observations can be viewed as being sampled within groups.

In the literature this is not the first time in which tree-based methods are adopted to deal with longitudinal and clustered data. In [19] a regression tree method for longitudinal or clustered data is proposed. This method is called Random Effects Expectation-

[1] Universitat Autonoma de Barcelona (UAB) - Spain; Instituto Politecnico de Braganca (IPB) - Portugal; Opole University of Technology - Poland; Politecnico di Milano (PoliMi) - Italy; Universidad de Leon - Spain; University of Galati *Dunarea de Jos* - Romania.

Maximization (RE-EM) tree. Independently, in [12] a Mixed-Effect Regression Tree (MERT) model is proposed. If clustered observations are considered, these are extensions of a standard regression tree to the case of individuals nested within groups. These methods use observation-level covariates in the splitting process and can deal with the possible random effects associated to those covariates. However, they both deal with a Gaussian response variable and they are not suitable to a classification problem.

In [11] the MERT approach is extended to non-gaussian data and a generalized mixed effects regression tree (GMERT) is proposed. This algorithm is basically the PQL algorithm used to fit GLMMs where the weighted linear mixed-effect pseudo-model is replaced by a weighted MERT pseudo-model.

Following a different strategy, our proposed method intends to generalize the RE-EM tree approach. In particular, in this work we expand its use to different classes of response variables from the exponential family: this would allow to extend it to a classification setting. At the same time this method can deal with the grouped data structure, similarly to traditional multilevel models. As in RE-EM tree estimation, we develop an algorithm that disentangles the estimation of fixed and random effects. That is, an initial tree is built ignoring the grouped data structure, a mixed-effects model is fitted based on the resultant tree structure, and a final mixed-effects tree is reported.

In this paper we apply this model to the Politecnico di Milano dataset. In this specific case, we can identify which fixed-effects covariates discriminate between dropout and graduate students. Through a GMET model, we can relax the assumption of linear effects of student-level covariates on their performance and we can identify which interactions relevantly influence the career status. In addition, the choice of a multilevel model allows to estimate the degree programme effect on the predicted probability of obtaining the degree.

The paper is organized as follows. In Section 2 we describe model and methods - generalized mixed tree algorithm (GMET) - and in Section 3 we show a simulation study. In Section 4 we describe the PoliMi dataset, we report the application of the proposed algorithm to the case study and outline the results. Finally, in Section 5 we draw our conclusions.

All the analysis are made using R software [17]. The code for the algorithm is available upon request to the authors.

2 Model and methods

In this section, we present the proposed generalized mixed-effects tree model (Subsection 2.1) and the algorithm for the estimation of its parameters (Subsection 2.2).

2.1 Generalized mixed-effects tree model

We start considering a generic GLMM. This model is an extension of a generalized linear model that includes both fixed and random effects in the linear predictor [1]. Therefore, GLMMs handle a wide range of response distributions and a wide range of scenarios where observations are grouped in groups rather than completely independently. For a GLMM with a two-level hierarchy, each observation j, for $j = 1, \ldots, n_i$, is nested within a group i, for $i = 1, \ldots, I$. Let $\boldsymbol{y}_i = (y_{1i}, \ldots, y_{n_i i})$ be the n_i-dimensional response vector

for observations in the i-th group. Conditionally on random effects denoted by $\boldsymbol{b}_i$, a GLMM assumes that the elements of $\boldsymbol{y}_i$ are independent, with density function from the exponential family, of the form

$$f_i(y_{ij}|\boldsymbol{b}_i) = \exp\left[\frac{y_{ij}\eta_{ij} - a(\eta_{ij})}{\phi} + c(y_{ij}, \phi)\right]$$

where $a(\cdot)$ and $c(\cdot)$ are specified functions, η_{ij} is the natural parameter and ϕ is the dispersion parameter. In addition, we have

$$\begin{aligned} E[y_{ij}|\boldsymbol{b}_i] &= a'(\eta_{ij}) = \mu_{ij} \\ Var[y_{ij}|\boldsymbol{b}_i] &= \phi\, a''(\eta_{ij}) \end{aligned}$$

A monotonic, differentiable link function $g(\cdot)$ specifies the function of the mean that the model equates to the systematic component. Usually, the canonical link function is used, i.e., $g = a'^{-1}$. From now on, without loss of generality the canonical link function is used. In this case, the model is the following [14]:

$$\begin{aligned} \boldsymbol{\mu}_i &= E[\boldsymbol{Y}_i|\boldsymbol{b}_i] \qquad i = 1, \ldots, I \\ g(\boldsymbol{\mu}_i) &= \boldsymbol{\eta}_i \\ \boldsymbol{\eta}_i &= X_i\boldsymbol{\beta} + Z_i\boldsymbol{b}_i \\ \boldsymbol{b}_i &\sim N_q(\boldsymbol{0}, \Psi) \qquad ind. \end{aligned} \tag{1}$$

where i is the group index, I is the total number of groups, n_i is the number of observations within the i-th group and $\sum_{i=1}^{I} n_i = J$, $\boldsymbol{\eta}_i$ is the n_i-dimensional linear predictor vector. In addition, X_i is the $n_i \times (p+1)$ matrix of fixed-effects regressors of observations in group i, $\boldsymbol{\beta}$ is the $(p+1)$-dimensional vector of their coefficients, Z_i is the $n_i \times q$ matrix of regressors for the random effects, $\boldsymbol{b}_i$ is the $(q+1)$-dimensional vector of their coefficients and Ψ is the $q \times q$ within-group covariance matrix of the random effects. Fixed effects are identified by parameters associated to the entire population, while random ones are identified by group-specific parameters.

Our proposed Generalized Mixed-Effects Tree (GMET) method expands the use of tree-based mixed models to different classes of response variables from the exponential family. At the same time the method can deal with the grouped data structure as GLMMs do. We now specify the GMET model. The random component of this model consists of a response variable Y from a distribution in the exponential family. The fixed part in the GMET is not linear as in (1) but it is replaced by the function $\boldsymbol{f}(X_i)$ that is estimated through a tree-based algorithm. Thus, the matrix formulation of the model is the following:

$$\begin{aligned} \boldsymbol{\mu}_i &= E[\boldsymbol{Y}_i|\boldsymbol{b}_i] \qquad i = 1, \ldots, I \\ g(\boldsymbol{\mu}_i) &= \boldsymbol{\eta}_i \\ \boldsymbol{\eta}_i &= \boldsymbol{f}(X_i) + Z_i\boldsymbol{b}_i \\ \boldsymbol{b}_i &\sim N_q(\boldsymbol{0}, \Psi) \qquad ind. \end{aligned} \tag{2}$$

where i is the group index, I is the total number of groups, n_i is the number of observations within the i-th group and $\sum_{i=1}^{I} n_i = J$. In addition, $\boldsymbol{\eta}_i$ is the n_i-dimensional linear predictor vector and $g(\cdot)$ is the link function. Finally, X_i is the $n_i \times (p+1)$ matrix of

fixed-effects regressors of observations in group i, Z_i is the $n_i \times q$ matrix of regressors for the random effects, $\boldsymbol{b}_i$ is the $(q+1)$-dimensional vector of their coefficients and Ψ is the $q \times q$ within-group covariance matrix of the random effects. As in a GLMM, $\boldsymbol{b}_i$ and $\boldsymbol{b}_{i'}$ are independent for $i \neq i'$. Fixed effects are identified by a non-parametric CART tree model associated to the entire population, while random ones are identified by group-specific parameters.

Without loss of generality, let us now specify model (2) for the case of a binary random variable and univariate random effect. The logit function is the canonical link function:

$$g(\mu_{ij}) = g(p_{ij}) = \log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \text{logit}(p_{ij}).$$

Here, the random-effects structure simplifies to a random intercept. The model formulation for observation y_{ij} may therefore be written as:

$$\begin{aligned} Y_{ij} &\sim \text{Bernoulli}(p_{ij}) \qquad\qquad i = 1, ..., I \qquad j = 1, ..., n_i \\ p_{ij} &= E[Y_{ij}|b_i] \\ \text{logit}(p_{ij}) &= f(\boldsymbol{x}_{ij}) + b_i \\ b_i &\sim N(0, \sigma^2) \qquad ind. \end{aligned} \tag{3}$$

where we observe $\boldsymbol{x}_{ij} = (x_{1ij}, .., x_{ijp})^T$, a $(p+1)$-dimensional vector of fixed-effects covariates for each observation j in group i.

2.2 Generalized mixed-effects tree estimation

In this subsection we show the algorithm for the estimation of the parameters of the GMET model (2). The basic idea behind the algorithm is to disentangle the estimation of fixed and random effects. The structure of the algorithm is the following:

1. Initialize the estimated random effects $\boldsymbol{b}_i$ to zero.
2. Estimate the target variable μ_{ij} through a generalized linear model (GLM), given fixed-effects covariates $\boldsymbol{x}_{ij} = (x_{ij1}, ..., x_{ijp})^T$ for $i = 1, ..., I$ and $j = 1, ..., n_i$. Get estimate $\hat{\mu}_{ij}$ of target variable μ_{ij}.
3. Build a regression tree approximating f using $\hat{\mu}_{ij}$ as dependent variable and $\boldsymbol{x}_{ij} = (x_{ij1}, ..., x_{ijp})^T$ as vector of covariates. Through this regression tree, define a set of indicator variables $I(\boldsymbol{x}_{ij} \in R_\ell)$ where the index ℓ ranges over all of the terminal nodes in the tree.
4. Fit the mixed effects model (2), using y_{ij} as response variable and the set of indicator variables $I(\boldsymbol{x}_{ij} \in R_\ell)$ as fixed-effects covariates. Specifically, for $i = 1, ..., I$ and $j = 1, ..., n_i$, we have $g(\mu_{ij}) = I(\boldsymbol{x}_{ij} \in R_\ell)\gamma_\ell + \boldsymbol{z}_{ij}^T \boldsymbol{b}_i$. Extract $\hat{\boldsymbol{b}}_i$ from the estimated model.
5. Replace the predicted response at each terminal node R_ℓ of the tree with the estimated predicted response $g(\hat{\gamma}_\ell)$ from the mixed-effects model fitted in Step 4.

The GLM in Step 2 is fitted through maximum likelihood. The maximum likelihood estimates can be found using an iteratively reweighted least squares algorithm or a Newton-Raphson method [15].

The fitting of the tree in Step 3 can be achieved using any tree algorithm, based on any tree-growing rules that are desired. Here, tree building is based on the CART tree algorithm [5]. After building a large tree T_0, pruning is advised to avoid overfitting on training data. In principle, any tree-pruning rule could be used; here, we propose cost-complexity pruning [13]. It considers a sequence of nested trees indexed by a nonnegative tuning parameter α which controls the trade-off between the subtree's complexity and its fit to the training data. For each value of α exists a subtree $T \subset T_0$ to minimize

$$\sum_{\ell=1}^{|T|} \sum_{x_i \in R_\ell} (y_i - \hat{y}_{R_\ell})^2 + \alpha |T|. \tag{4}$$

Here, $|T|$ indicates the number of terminal nodes of tree T. When $\alpha = 0$, then the subtree T will simply be equal to T_0. However, as α increases, the quantity (4) will tend to be minimized for a smaller subtree. We can select a value of α using a validation set or using K-fold cross-validation: for example, we can pick $\tilde{\alpha}$ to minimize the average CV error. Tree building and pruning is implemented in `R` library `rpart` [22], according to the CART tree-building algorithm and cost-complexity pruning. In order to ensure that initial trees are sufficiently large, we set the complexity parameter to zero. Thus, the largest tree is grown then pruned based on ten-fold cross-validation error. Instead of choosing the tree that achieves the lowest CV error, we use the so-called *1-SE rule*: any CV error within one standard error of the achieved minimum is marked as being equivalent to the minimum. Among all these equivalent models in terms of CV error, the simplest one is chosen as final tree model.

The generalized linear mixed model in Step 4 can be estimated using fitting techniques that were previosly described. Different statistical packages can estimate those type of models: the `glmer` function of the `R` library `lme4` [3] is used here. It fits a generalized linear mixed model via maximum likelihood. For a GLMM the integral must be approximated: the most reliable approximation is adaptive Gauss-Hermite quadrature, at present implemented only for models with a single scalar random effect, otherwise Gaussian quadrature is used.

Prediction for new observations

After estimating a GMET it is possible to make out-of-sample predictions for new observations. Suppose the tree is estimated on data from groups $i = 1, .., I$ for observations y_{ij}, $j = 1, ..., n_i$. Given a new observation $\boldsymbol{x}_{ij'}$ we are able to output its corresponding response since we know the estimation of the fixed-effects function $f(\cdot)$, of the random effects $\boldsymbol{b}_i$ and of the associated covariance matrix Ψ. We may look for two types of prediction:

- predict response $y_{ij'}$ given a new observation $\boldsymbol{x}_{ij'}$ for a group in the sample $i \in \{1, ..., I\}$. We define it a *group-level prediction*.
- predict response $y_{i'j'}$ given an observation $\boldsymbol{x}_{i'j'}$ for a group i' for which there are

no observations in our current sample, or for which we do not know the group it belongs to. We define it a *population-level prediction*.

For the first type of prediction, we estimate $f(\boldsymbol{x}_{ij'})$ using the estimated tree and attributes $\boldsymbol{x}_{ij'}$ and then add $\boldsymbol{z}_{ij'}^T \boldsymbol{b}_i$ on the linear predictor scale, and get back to the response scale through the inverse link function $g^{-1}(\cdot)$. As we underlined before, random-effects coefficients $\boldsymbol{b}_i$ are known from the estimation process.

For the second type of prediction, we have no information to evaluate $\boldsymbol{b}_i$. A possible solution is to set it to its expected value of 0, yielding the value $\hat{f}(\boldsymbol{x}_{i'j'})$, and transform it back to the response scale through the inverse link function. As noted in [19], in this case we might expect that methods that do not incorporate random effects would have comparable performance to those that do, as long as the sample is large enough so that the fixed-effects function $f(\boldsymbol{x}_{ij'})$ is well-estimated by both types of methods.

3 Simulation study

In this section we compare the performance of the proposed GMET method to standard classification trees on different simulated binary outcomes datasets.

We first use a variation of a simulation design proposed in [11]. It has a two-level data structure of $I = 50$ groups with $n_i = 60$ observations each: 10 observations in each group are included in the training sample, and the other 50 observations constitute the test sample. Therefore, $N_{\text{train}} = 500$, while $N_{\text{test}} = 2500$. Setting $i = 1, ..., I$ and $j = 1, ..., n_i$, the response values y_{ij} are simulated according to a Bernoulli distribution with conditional probability of success μ_{ij}. Both fixed and random effects are used to generate μ_{ij}. Overall, we consider 10 different Data Generating Processes (DGP) outlined in Table 1 by combining different fixed- and random-effect specifications.

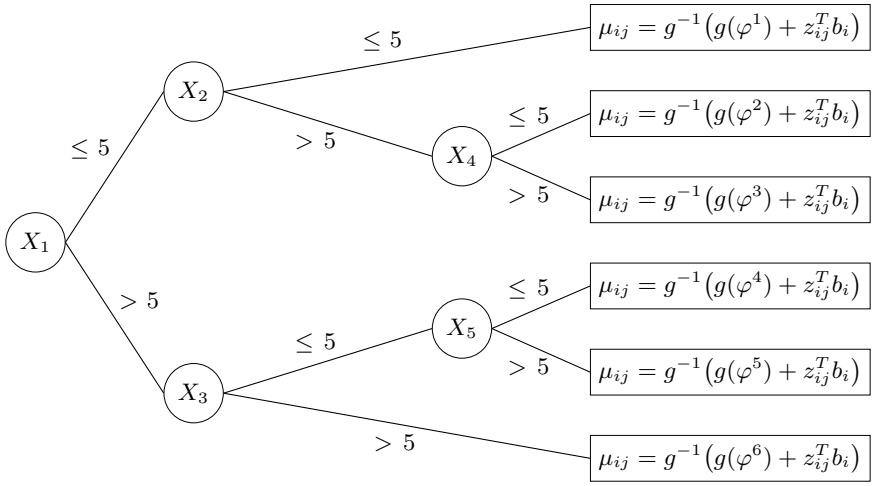

Figure 1: *Mixed-effects tree structure used to generate the conditional probability of success μ_{ij} in the simulation study*

Let us define the fixed-effect structure. Eight random variables $X_1, ..., X_8$, independent and uniformly distributed in the interval $[0, 10]$, are generated. While all of them are being used as predictors, only five of them are actually used to generate μ_{ij}, based on the tree rule summarized in Figure 1. Each observation is classified into one of the six terminal nodes according to the values $x_{ij1}, ..., x_{ij5}$. Within each leaf, values $\varphi^1, ..., \varphi^6$ denote the probabilities of success when the random effects b_i are equal to zero:

Leaf 1: if $x_{1ij} \leq 5 \wedge x_{2ij} \leq 5$ then $\mu_{ij} = g^{-1}\big(g(\varphi^1) + z_{ij}^T b_i\big)$;

Leaf 2: if $x_{1ij} \leq 5 \wedge x_{2ij} > 5 \wedge x_{4ij} \leq 5$ then $\mu_{ij} = g^{-1}\big(g(\varphi^2) + z_{ij}^T b_i\big)$;

Leaf 3: if $x_{1ij} \leq 5 \wedge x_{2ij} > 5 \wedge x_{4ij} > 5$ then $\mu_{ij} = g^{-1}\big(g(\varphi^3) + z_{ij}^T b_i\big)$;

Leaf 4: if $x_{1ij} > 5 \wedge x_{3ij} \leq 5 \wedge x_{5ij} \leq 5$ then $\mu_{ij} = g^{-1}\big(g(\varphi^4) + z_{ij}^T b_i\big)$;

Leaf 5: if $x_{1ij} > 5 \wedge x_{3ij} > 5 \wedge x_{5ij} > 5$ then $\mu_{ij} = g^{-1}\big(g(\varphi^5) + z_{ij}^T b_i\big)$;

Leaf 6: if $x_{1ij} > 5 \wedge x_{3ij} > 5$ then $\mu_{ij} = g^{-1}\big(g(\varphi^6) + z_{ij}^T b_i\big)$;

where $g(\cdot)$ is the logit link function. Two different possibilities are specified for the fixed effects: in the *large* fixed-effects specification, the standard deviation of the typical probabilities across the leaves is higher than in the *small* one (0.37 versus 0.24).

The random component $b_i \sim N(0, \Psi)$ is generated according to three different possibilities:

- No random effects: $\Psi = 0$;
- Random intercept: $z_{ij} = 1 \quad \forall i, \forall j$ and $\Psi = \psi_{11}$;
- Random intercept and slope, which add a linear random effect for the fixed-effect covariate X_1, uncorrelated from the random effect on the intercept. That is, $z_{ij} = [1 \quad x_{1ij}]^T \quad \forall i, \forall j$ and $\Psi = \begin{bmatrix} \psi_{11} & 0 \\ 0 & \psi_{22} \end{bmatrix}$.

Within each fixed effects scenario with random effects, we consider two specifications (*low* and *high*) for the covariance matrix Ψ to account for different levels of magnitude of the between-group variability.

Simulation results

We fit four different models for each one of the 10 DGPs: a standard binary classification tree model (*Std*), a random intercept GMET model (*RI*), a random intercept and slope GMET model (*RIS*), a parametric mixed-effects logistic regression model (*MElog*) that uses the true model leaves' indicators as fixed covariates. As noted in [12] the MElog model could not be a real competitor of any other model. Indeed, it is not possible in practice to specify this parametric structure without knowing the underlying data generating process. This model only serves as a reference to compare the performance of the other models. In tree-based models, we fix to 10 the maximum depth parameter and to 20 the minimum number of observations necessary to attempt a split. After fitting each model on the training set, we can compute the corresponding predicted probability $\hat{\mu}_{ij}$ and the predicted class $\hat{y}_{ij}$ of observation j in group i in the test dataset. While the

DGP	Random component				Fixed component						
	Structure	**Effect**	ψ_{11}	ψ_{22}	**Effect**	φ^1	φ^2	φ^3	φ^4	φ^5	φ^6
1	No random	–	–	–	Large	0.10	0.20	0.80	0.20	0.80	0.90
2	effect	–			Small	0.20	0.40	0.70	0.30	0.60	0.80
3	Random Intercept	Low	4.00	–	Large	0.10	0.20	0.80	0.20	0.80	0.90
4		High	10.00	–							
5		Low	0.50	–	Small	0.20	0.40	0.70	0.30	0.60	0.80
6		High	4.00	–							
7	Random Intercept and Slope	Low	2.00	0.05	Large	0.10	0.20	0.80	0.20	0.80	0.90
8		High	5.00	0.25							
9		Low	0.25	0.01	Small	0.20	0.40	0.70	0.30	0.60	0.80
10		High	2.00	0.05							

Table 1: *Data Generating Processes (DGP) for the simulation study*

former is directly estimated by the algorithm, the latter depends on the threshold value μ_k^* used to classify subjects in the test set: $\hat{\mu}_{ij} \geq \mu_k^* \Rightarrow \hat{y}_{ij} = 1$ where $(i, j) \in$ test. There are at most K distinct fitted values μ_k, with $K \leq I|T|$. We use each of them to classify observations in the training set and we fix the threshold μ_k^* as the one that yields the closest proportion of class 1 to the actual proportion of class 1 in the training set.

We measure the predictive performance by:

- the *predictive mean absolute deviation* (PMAD) of the estimated probability

$$\text{PMAD} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{I} \sum_{j=1}^{n_i^{\text{test}}} |\mu_{ij} - \hat{\mu}_{ij}|$$

- the *predictive misclassification rate* (PMCR)

$$\text{PMCR} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{I} \sum_{j=1}^{n_i^{\text{test}}} |y_{ij} - \hat{y}_{ij}|.$$

The mean, standard deviation, minimum and maximum of the PMAD and the PMCR over 50 runs were calculated and are reported in Table 2.

We observe that when there is no random effect (DGPs 1 and 2), the standard classification tree algorithm performs better, specifically when the fixed effect is large. However, when random effects are present (DGPs 3 to 10), the mixed effects classification tree performs better than the standard classification tree in terms of average PMAD. The highest improvement in PMAD using a mixed tree model is observed when both the fixed and the random effects are large (16.50% in DGP4 - *Std vs RI* and 16.78% in DGP8 - *Std vs RIS*). The lowest improvement is observed when both the fixed and the random effects are small (2.35% in DGP5 - *Std vs RI* and 2.34% in DGP9 - *Std vs RIS*). Analogous considerations can be made about PMCR. In addition, GMETs perform better than standard trees even

DGP	Random effect	Fixed effect	Fitted model	PMAD (%)				PMCR (%)			
				mean	sd	min	max	mean	sd	min	max
1	NO RANDOM EFFECT	Large	Std	5.35	1.53	2.71	8.79	17.20	1.40	14.64	20.52
			RI	20.22	2.31	15.15	24.72	31.09	2.63	26.04	37.68
			RIS	20.36	2.36	13.14	24.88	31.03	2.40	24.24	35.48
			MElog	3.11	0.88	1.42	4.95	17.79	3.18	14.52	24.24
2		Small	Std	12.93	2.78	7.01	19.28	33.16	2.18	28.92	38.60
			RI	13.99	1.78	9.84	17.19	37.57	1.88	32.72	41.64
			RIS	14.08	1.82	9.93	17.81	37.33	1.79	33.16	41.56
			MElog	4.16	1.30	1.02	6.45	29.32	1.63	26.96	33.16
3	Low	Large	Std	23.83	2.94	17.53	29.88	30.53	3.13	23.32	38.20
			RI	**18.28**	1.47	15.07	22.67	**26.80**	1.86	22.84	31.92
			RIS	18.43	1.31	15.28	21.89	26.84	1.73	22.72	30.76
			MElog	**8.59**	0.87	6.02	10.56	**19.34**	1.29	16.08	22.48
4	High		Std	32.05	2.37	26.90	37.59	37.80	2.65	32.08	44.96
			RI	15.55	1.28	12.49	18.71	21.62	1.88	16.32	26.56
			RIS	15.66	1.27	12.52	18.91	21.71	1.87	16.56	26.40
			MElog	8.09	0.76	6.04	10.06	16.32	1.53	13.32	19.80
5	INTERCEPT Low	Small	Std	17.89	2.32	13.28	22.48	35.30	2.23	31.40	41.40
			RI	15.54	1.58	12.52	19.12	35.89	2.18	30.76	41.20
			RIS	15.76	1.56	12.76	19.63	36.12	2.14	31.20	41.32
			MElog	8.63	0.92	6.49	10.53	28.90	0.95	27.20	31.84
6	High		Std	29.47	2.22	24.56	35.08	41.42	2.36	36.36	45.48
			RI	**14.11**	1.46	10.17	17.38	**26.23**	2.35	21.40	30.96
			RIS	14.25	1.49	10.39	17.81	26.27	2.40	21.28	31.20
			MElog	**9.36**	0.98	7.07	11.25	**22.85**	1.70	19.12	26.08
7	Low	Large	Std	23.24	2.49	18.54	29.68	29.61	2.91	23.44	38.44
			RI	19.59	1.37	15.42	22.51	27.89	1.98	22.16	31.20
			RIS	**19.29**	1.40	15.15	22.22	**27.84**	1.82	22.08	31.08
			MElog	**10.01**	1.02	8.07	11.91	**19.92**	1.37	17.20	24.04
8	High		Std	32.89	2.61	27.47	38.04	38.69	3.67	31.64	46.32
			RI	17.52	1.57	14.29	20.85	22.03	2.04	17.48	26.08
			RIS	16.11	1.41	12.90	18.93	21.26	1.92	17.04	25.48
			MElog	9.86	1.02	7.82	13.16	16.59	1.48	13.20	20.36
9	INTERCEPT & SLOPE Low	Small	Std	18.15	2.25	13.36	24.73	35.34	2.56	31.36	42.64
			RI	15.84	1.17	12.37	18.61	35.83	1.92	30.84	40.48
			RIS	15.81	1.24	12.41	19.05	35.76	1.92	31.28	39.80
			MElog	9.31	0.86	7.95	11.06	29.11	0.94	26.76	30.96
10	High		Std	29.09	2.06	24.21	33.51	41.64	2.45	37.16	49.76
			RI	15.88	1.26	13.60	19.77	27.66	1.97	23.00	32.76
			RIS	**15.21**	1.15	13.20	18.32	**27.20**	1.93	21.96	31.64
			MElog	**10.80**	1.02	9.20	13.06	**24.25**	1.69	20.32	28.04

Table 2: *Results of the 50 simulation runs in terms of predictive probability mean absolute deviation (PMAD) and predictive misclassification rate (PMCR). In bold, DGPs in which the performance gap between MElog and GMET is the largest or the smallest are marked.*

when we fit a mixed tree whose random component is over-specified (like in DGPs 3-6, *Std vs RIS*) or under-specified (like in DGPs 7-10, *Std vs RI*) in relation to the true data generating process.

Next, we compare the performance of the GMET approach to the results of the MElog reference model. If the DGP does not include random effects, the difference in PMAD and PMCR is higher when the fixed effects are large (DGP1). When random effects are large and fixed effects are small (DGPs 6 and 10), the GMET model performs closer to the MElog model. In terms of PMAD, this difference equals to 4.75% and 4.41% in DGPs 6 and 10 respectively; in terms of PMCR it equals to 3.38% and 2.95% respectively. The difference in predictive accuracy between the two models reaches the maximum when random effects are small and fixed effects are large (DGPs 3 and 7). In terms of PMAD, this difference equals to 9.69% and 9.28% in DGPs 3 and 7 respectively; in terms of PMCR it equals to 7.46% and 7.92% respectively.

4 Case study: application of mixed-effects tree algorithm to education PoliMi data

In this section, we describe the PoliMi dataset and we apply the generalized mixed-effects tree algorithm to these data. Using a GMET model, we can identify discriminating fixed-effects covariates and estimate the degree programme effect on the predicted success probability. In addition, we also analyse the accuracy of this model in predicting dropout careers.

The PoliMi dataset consists of 18,612 careers in Bachelor of Science (BSc) that began between A.Y. 2010/2011 and 2013/2014. Students are nested within $I = 19$ degree programmes.[2] A descriptive analysis shows that a high percentage of students leaves the Politecnico before obtaining the degree. Therefore, our goal is to find out which student-level indicators could discriminate between two different profiles: *dropout* and *graduate* students.

We assume the binary GMET model (3) where student j is nested within degree programme i. The response variable Y is the career `status`, a two-level factor we code as a binary variable:

- `status` $= 1$ for careers definitely completed with graduation;
- `status` $= 0$ for careers definitely concluded with a dropout.

We would like to make predictions at the very early stage of the academic career. So, we choose as predictors five variables available at the time of enrollment and three more variables collected just after the first semester of studies. The list and explanation of student-level variables to be included as covariates is reported in Table 3. In addition we choose as grouping variable the degree programme at the time of the enrollment (factor `DegreeProgramme`) which has 19 levels. The influence of the grouping factor on the predictor is modeled through a group-level intercept b_i. We randomly split the dataset

[2]We are considering the following Engineering programmes: Aerospace, Automation, Biomedical, Building, Chemical, Civil, Civil and Environmental, Electrical, Electronic, Energy, Computing Systems, Environmental and Land Planning, Industrial Production, Management, Materials and Nanotechnology, Mathematical Mechanical, Physics, Telecommunications.

into training and test subsets, with a ratio of 80% for training and 20% for evaluation. Thus, the training subset equals to 14,890 careers while the test subset amounts to 3,722 careers.

Variable	Description	Type of variable
`Sex`	gender	factor (2 levels: M, F)
`Nationality`	nationality	factor (Italian, foreigner)
`PreviousStudies`	high school studies	factor (*Liceo Scientifico*, *Istituto Tecnico*, Other)
`AdmissionScore`	PoliMi admission test result	real number
`AccessToStudiesAge`	age at the beginning of the BSc studies at PoliMi	natural number
`WeightedAvgEval1.1`	weighted average of the evaluations during the first semester of the first year	real number
`AvgAttempts1.1`	average number of attempts to be evaluated on subjects during the first semester of the first year (passed and failed exams)	real number
`TotalCredits1.1`	number of ECTS credits obtained by the student during the first semester of the first year	natural number

Table 3: *List and explanation of variables at student level to be included as covariates in the GMET model*

While growing the tree, we fix to 10 the maximum depth parameter and to 20 the minimum number of observations necessary to attempt a split. Figure 2 shows the estimated mixed-effects tree for the graduating probability. Every internal node has its corresponding condition that splits it into two sons: if the condition is true, observations are sent down the tree through the left son, while through the right son if the condition is false. In addition, all nodes report two values: the estimated graduating probability and the percentage of observations in the node over the total training set. We remind that variable `PreviousStudies` has been coded as a three-level factor with levels S (*Liceo Scientifico*), T (*Istituto Tecnico*) and O (other high school studies). The number of ECTS obtained in the first semester of the first year is used as first split: students who obtained less than 13 ECTS are associated to lower success probability (0.16 versus 0.86). Then, students are further classified using other explainatory variables: we can notice that Italian students who obtained more than 24 ECTS have the highest predicted success probability (0.95). Other variables actually used to split smaller internal nodes are `Nationality` and `PreviousStudies`: in these nodes, students who attended *Istituto Tecnico* and foreign students have lower predicted success probability than the others. Through this model, it is possible to point out significant interactions among the covariates: for example, variable `Nationality` is used to split the group of students that obtained at least 13 ECTS, while this same variable does not appear in the complementary branch of the tree. Finally, covariates `Sex`, `AdmissionScore` and `AvgAttempts1.1` do not compare in the trees, so they do not appear to have strong influence on how a career ends.

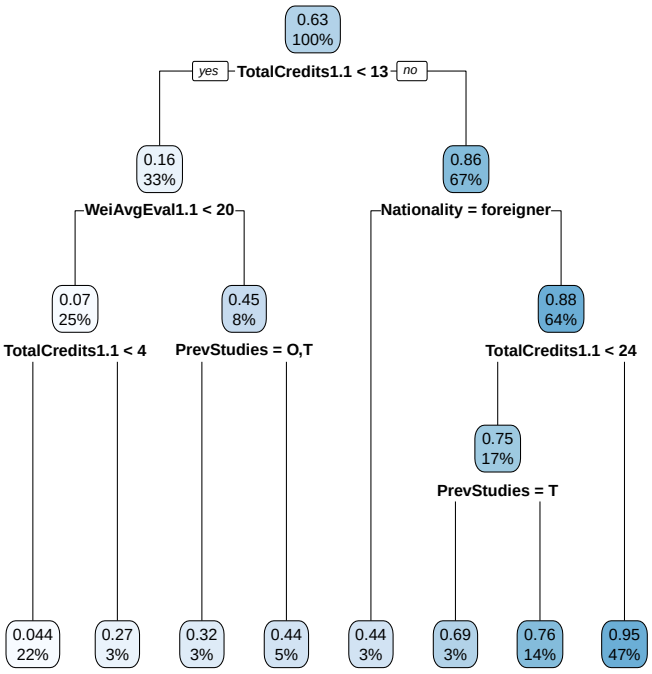

Figure 2: *Estimated mixed-effects tree of model* (3) *for the graduating probability*

Using the tree structure in Figure 2, we can get a population-level prediction for new observations that do not include the effect of the programme. However, if we also specify the level of the random effect covariate, our model is able to adjust this prediction to account for this effect and make a group-specific prediction. Indeed, we can extract coefficients $\hat{b}_i$ from the full estimated mixed model (3) and provide different predictions for different programmes within each leaf of the tree structure. Figure 3 shows the estimated random effects for all 19 groups in the dataset. The coefficients b_i are rearranged by their point estimate. In many groups, the 95% confidence interval does not overlap the vertical line at zero, underlining substantial differences between the groups. If we use this model to estimate the graduating probability, in many of the groups it is significantly different from the average one. After fixing all other covariates, levels *Environmental and Land Planning Engineering* and *Civil and Environmental Engineering* have higher positive effect on the intercept: being a student from one of these programmes improves the log odds by 1.051 and 0.705 respectively. On the contrary, studying either *Civil Engineering* or *Electrical Engineering* penalizes the log odds by 0.680 and 0.546 respectively.

Since we are using a multilevel model we can account for the interdependence of observations by partitioning the total variance into different components due to the clustered data structure in model (3). The *Variance Partition Coefficient* (VPC) is a possible measure of intraclass correlation: it is equal to the percentage of variation that is found at the higher level of hierarchy over the total variance [10]. The idea of VPC was extended using the latent variable approach, to define a method to partition the total variance in

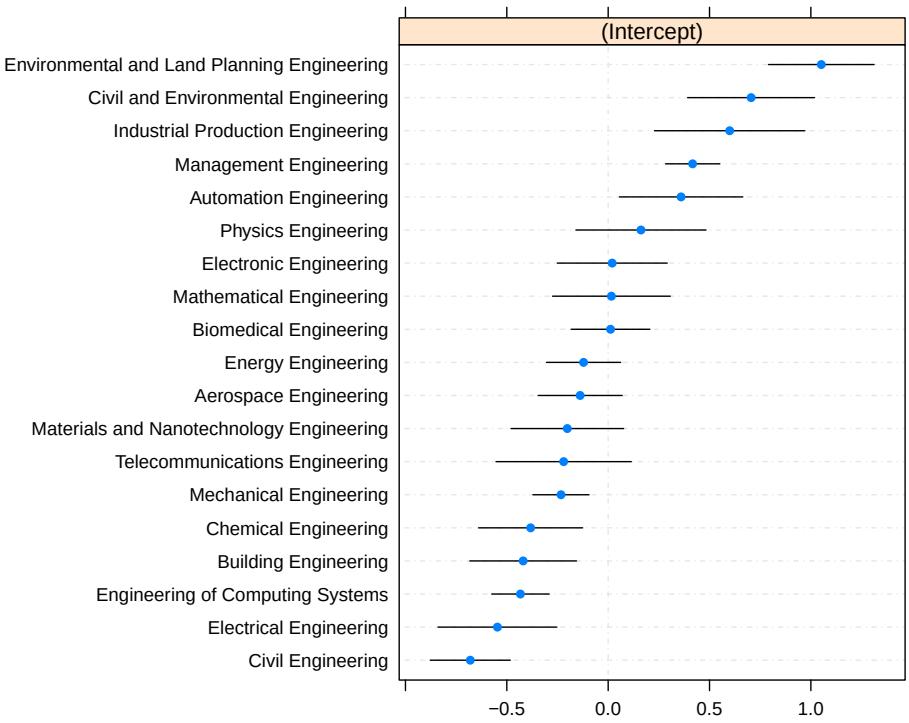

Figure 3: *Estimated random intercept for each degree programme in model* (3)*. For each Engineering programme, the blue dot and the horizontal line marks the estimate and the 95% confidence interval of the corresponding random intercept*

the case of a binary response and group-specific intercept as random effects structure [7]. In this case, the Variance Partition Coefficient is constant across all individuals and it can be estimated as:

$$\text{VPC} = \frac{\hat{\sigma}_b^2}{\hat{\sigma}_b^2 + \sigma_{lat}^2} = \frac{0.2988}{0.2988 + \pi^2/3} = 0.0612$$

where $\hat{\sigma}_b^2$ is the estimated variance of the random intercept and σ_{lat}^2 is the residual variability that can neither be explained by fixed effects, nor through the group features that are represented by the random intercept. In this case, it is equal to the variance of the standard logistic distribution. This VPC value means that 6.12% of variation in the response is attributed to the classification by degree type. This value underlines the need to use a mixed model.

We can now evaluate the performance of the model and its predictive quality using the test data. For each test observation, we are given a full set of covariates: therefore, we are able to compute an estimate $\hat{p}$ of the probability of successfully concluding the BSc and getting the degree. We use this estimate to define a binary classifier based on model (3): we choose $p_0 = 0.6$ as optimal cutoff value through ROC curve analysis. For 20 iterations, we randomly split the observations in training and test set, we fit a

GMET model on the training set and we classify test observations using the optimal threshold value. At the end, we compute the average accuracy, sensitivity and specificity and their standard deviation, reported in Table 4. High values of accuracy, sensitivity and specificity point to a good effectiveness of the model. In addition, the model performance is robust, as highlighted by the low standard deviation of mean performance indexes.

It is interesting to compare these average performance indexes against those obtained using different methods. This approach has similar accuracy to a standard classification tree (0.878 versus 0.879), but its accuracy shows less variability across the iterations. For example, its standard deviation of accuracy is 0.5% against 2.8% for a classification tree. Since we are interested in the detection of dropout careers, we should compare mean sensitivity using different models. Using mixed-effects trees, we get higher sensitivity than using standard classification trees (0.835 versus 0.800). Thus, the choice of a mixed-effects model seems appropriate: the degree programme is a meaningful covariate for the prediction of career `status`. A mixed-effects tree is slightly less sensitive than a classifier build through a GLMM (0.835 versus 0.850), suggesting that a tree-like structure for fixed effects might not be as suitable as the GLMM one. However, it has other advantages like offering an easily interpretable model that could be graphically displayed and understood.

Index	**Mean**	**Std deviation**
Accuracy	0.860	0.006
Sensitivity	0.816	0.012
Specificity	0.886	0.008

Table 4: *Performance indexes of a classifier based on the mixed-effects tree of model* (3)

5 Conclusions

This paper proposes a multilevel tree-based model for a non-gaussian response (GMET algorithm), shows a simulation study and applies the GMET algorithm to the PoliMi careers dataset as a tool to find discriminating student-level variables between two different student profiles (graduate and dropout) and to estimate the degree programme effect on the predicted success probability.

The GMET model can deal with a grouped data structure, while providing easily interpretable models that can outline complex interactions among the input variables. In the simulation study, the performance of the proposed mixed-effects tree method is a marked improvement over the CART model when the data generating process (DGP) includes random effects, even if of small magnitude. In addition, the performance of the GMET model is closer to the one of the benchmark logistic model that is fitted assuming the whole specification of the DGP. Although our study focuses on the binary response case, the mixed-effects tree approach could be extended to other types of response variables. Using a suitable link function, we could study if the method is appropriate to model different outcomes such as counts data or a multinomial factor response. Moreover, ensemble methods which use a mixed-effects tree as base learner may be developped.

In our case study, the effectiveness of the GMET model in dropout prediction is comparable to the ones of more established classification methods. A GMET model with high accuracy and sensitivity has been obtained by considering information available at the time of the admission and the career of the first semester of studies. In addition, our work identifies a significant effect of the Engineering programme on dropout probability.

In the context of the SPEET project, a future development could be the extension of our analysis to the other project partners in order to compare the programme effect at country level. This would allow us to relate this effect to programme-level variables and we could establish if the same profiles of students with dropout risk arise at country level. Moreover, in accordance to the validity and the potential of GMET method when applied to model student dropout prediction, our future perspective goes in the direction of major applications in the Learning Analytics area. This method, when applied to educational data, can be a useful tool to support the definition of best practices and new tutoring programmes aimed at enhancing student performances and reducing student dropout. A worthwhile aspect regards also the approach that teachers and students have with respect to its results. Indeed, this method is also valuable in the perspective of recommendation systems, since, if its results are interpreted and communicated in the right way, they can be used to drive students in their career choices.

Acknowledgements

This work is within the Student Profile for Enhancing Engineering Tutoring (SPEET) project, funded by Erasmus$^+$. The authors are grateful to Umberto Spagnolini and Aldo Torrebruno for their comments and support during this work.

References

[1] A. Agresti. *Categorical Data Analysis.* 2nd ed. Wiley Series in Probability and Statistics. Wiley, 2007.

[2] M. Barbu, R. Vilanova, J. Lopez Vicario, M.J. Varanda, P. Alves, M. Podpora, M.A. Prada, A. Moran, A. Torrebruno, S. Marin, R. Tocu. Data Mining Tool for Academic Data Exploitation, Literature review and first architecture proposal. *SPEET Intellectual Output #1*, ERASMUS+ KA2 / KA203, 2017.

[3] D. Bates, M. Malcher, B. Bolker, S. Walker. Fitting Linear Mixed-Effects Models using lme4. *Journal of Statistical Software* 67, 1, 1-48, 2015.

[4] R.D. Bock. *Multilevel Analysis of Educational data.* Elsevier, 2014.

[5] L. Breiman, J. Friedman, C.J. Stone, R.A. Olshen. *Classification and Regression Trees.* The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984.

[6] N.E. Breslow, D.G. Clayton. Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* 88, 9-25, 1993.

[7] W. Browne, S.V. Subramanian, K. Jones, H. Goldstein. Variance partitioning in multilevel logistic models that exhibit overdispersion. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 168, 3, 599-613, 2005

[8] W.R. Gilks, D. Spiegelhalter, S. Richardson. Introducing Markov chain Monte Carlo. In *Markov Chain Monte Carlo in Practice*, pp. 1-19, Chapman and Hall

[9] H. Goldstein. *Multilevel Statistical Models.* 4th ed. Wiley Series in Probability and Statistics. Wiley, 2010.

[10] H. Goldstein, W. Browne, J. Rasbash. Partitioning Variation in Multilevel Models. *Understanding Statistics* 1,4, pp. 223-231, 2002.

[11] A. Hajjem, D. Larocque, F. Bellavance. Generalized mixed effects regression trees. *Statistics & Probability Letters* 126, 2017.

[12] A. Hajjem, F. Bellavance, D. Larocque, . Mixed Effects Regression Trees for Clustered Data. *Statistics & Probability Letters* 81, 451-459, 2011.

[13] T. Hastie, R. Tibshirani, J. Friedman. *The Elements of Statistical Learning.* 2nd ed. Springer Series in Statistics. Springer, 2008.

[14] C.E. McCulloch, S.R. Searle. *Generalized, Linear, and Mixed Models.* Wiley Series in Probability and Statistics. Wiley, 2005.

[15] P. McCullagh, J.A. Nelder. *Generalized Linear Models.* Chapman and Hall, 1989.

[16] J.C. Pinheiro, E.C. Chao. Efficient Laplacian and adaptive Gaussian quadrature algorithms for multilevel generalized linear mixed models. *Journal of Computational and Graphical Statistics* 15, pp. 58-81, 2006.

[17] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2014.

[18] S.W. Raudenbush, M. Yang, M. Yosef. Efficient Laplacian and adaptive Gaussian quadrature algorithms for multilevel generalized linear mixed models. *Journal of Computational and Graphical Statistics* 9, 141-157, 2000.

[19] R.J. Sela, J.S. Simonoff. RE-EM trees: a data mining approach for longitudinal and clustered data. *Machine Learning* 86, 2, 2012.

[20] C. Romero, S. Ventura. Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics*, Part C (Applications and Reviews), 40, 6, 601-618, 2010.

[21] *SPEET, proposal for strategic partnerships (proposal narrative).* 2017. URL: `https://www.speet-project.com/the-project`

[22] T. Therneau, B. Atkinson. rpart: Recursive Partitioning and Regression Trees. *R package version 4.1-12*, 2018.

[23] R. Wolfinger, M. O'Connell. Generalized linear mixed models: a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation* 48, 233-243, 1993.

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

42/2018 Antonietti, P.F.; Melas, L.
Algebraic multigrid schemes for high-order discontinuous Galerkin methods

41/2018 Mazzieri, I.; Melas, L.; Smerzini, C.; Stupazzini, M.
The role of near-field ground motion on seismic risk assessment in large urban areas

39/2018 Ferro, N.; Micheletti, S.; Perotto, S.
Density-based inverse homogenization with anisotropically adapted elements

40/2018 Chiappa, A.S.; Micheletti, S.; Peli, R.; Perotto, S.
Mesh adaptation-aided image segmentation

38/2018 Domanin, M.; Gallo, D.; Vergara, C.; Biondetti, P.; Forzenigo, L.V.; Morbiducci, U.
Prediction of long term restenosis risk after surgery in the carotid bifurcation by hemodynamic and geometric analysis

37/2018 Bonaventura, L.; Della Rocca A.;
Convergence analysis of a cell centered finite volume diffusion operator on non-orthogonal polyhedral meshes

32/2018 Dal Santo, N.; Deparis, S.; Manzoni, A.; Quarteroni, A.
An algebraic least squares reduced basis method for the solution of nonaffinely parametrized Stokes equations

34/2018 Laurino, F.; Coclite, A.; Tiozzo, A.; Decuzzi, P.; Zunino, P.;
A muliscale computational approach for the interaction of functionalized nanoparticles with the miscrovasculature

35/2018 Possenti, L.; Casagrande, G.; Di Gregorio, S.; Zunino, P.; Costantino, M.L.
Numerical simulations of the microvascular fluid balance with a non-linear model of the lymphatic system

36/2018 Agosti, A.; Ambrosi, D.; Turzi, S.
Strain energy storage and dissipation rate in active cell mechanics