UNIVERSITÀ
DEGLI STUDI DI BARI
ALDO MORO

DIPARTIMENTO JONICO IN SISTEMI
GIURIDICI ED ECONOMICI DEL MEDITERRANEO
SOCIETÀ, AMBIENTE, CULTURE
IONIAN DEPARTMENT OF LAW, ECONOMICS
AND ENVIRONMENT

ANNO VII
ANNALI 2019
DEL DIPARTIMENTO JONICO
ESTRATTO

MASSIMO BILANCIA, PAOLA PRCHINUNNO, DOMENICO VITALE

A Monte Carlo study on learning algorithms for predicting student dropouts in higher education

EDIZIONI DJSGE

ISBN 9788894281088

Contatti:
Prof. Nicola Triggiani
Dipartimento Jonico in Sistemi Giuridici ed Economici
del Mediterraneo: Società, Ambiente, Culture

Convento San Francesco
Via Duomo, 259 - 74123 Taranto, Italy
e–mail: annali.dipartimentojonico@uniba.it
telefono: + 39 099 372382 • fax: + 39 099 7340595
https://www.uniba.it/ricerca/dipartimenti/sistemi-giuridici-ed-economici/edizioni-digitali

SAGGI

Massimo Bilancia[1], Paola Perchinunno[2], Domenico Vitale[3]

A MONTE CARLO STUDY ON LEARNING ALGORITHMS FOR PREDICTING STUDENT DROPOUTS IN HIGHER EDUCATION[*]

ABSTRACT

Il problema dell'abbandono degli studi universitari è una rilevante difficoltà con la quale il sistema universitario italiano deve confrontarsi. In questo lavoro, utilizzando opportune tecniche di simulazione, abbiamo esplorato i limiti e le possibilità di alcuni algoritmi di Machine Learning per prevedere gli abbandoni sulla base di una serie di variabili che sono immediatamente disponibili per ciascuno studente.

The phenomenon of dropping out is one the most significant problems faced by the Italian university system. In this paper, using suitable simulation techniques, we have explored the limits and the possibilities of some Machine Learning Algorithms to predict the probability of abandonment in a timely and efficient way, using an information set that is available at the time of matriculation.

PAROLE CHIAVE

Abbandono degli studi universitari – machine learning – previsioni

Students' dropout at University – machine learning – prediction

[1] Ionian Department of Law, Economics and Environment, University of Bari Aldo Moro, Taranto (IT). Correspondence to: massimo.bilancia@uniba.it.

[2] Department of Business and Law Studies (DEMDI), University of Bari Aldo Moro, Bari (IT).

[3] Department for Innovation in Biological, Agro-food and Forest Systems, University of Tuscia, Viterbo (IT)

[*] This paper was double-blind peer reviewed. We would like to thank the Rector of the University of Bari for authorizing the consultation, in an anonymous form and for scientific research purposes, of the data relating to the UniBA student population of the MIUR-Cineca Student-Didactic Observatory.

1. The phenomenon of dropping out is one of the most significant problems facing the Italian university system, also in terms of comparison with the European university system. Although the data on the most recent student cohorts show a slight improvement the phenomenon remains significant, with a dropout rate within 5 years reaching almost 30% [9].

The analysis of this issue can be conducted using two fundamentally distinct approaches. The first is centered on the theoretical identification of the socio-economic and organizational determinants that may be at the basis of the risk of failure in the university course. The most recent research has focused on the weight that must be attributed, in order to determine the probability of dropout, both to individual-endogenous factors, such as personal attitudes, motivation and educational background coming from high schools, as well as organizational-exogenous factors, such as the bad functioning of the universities or the educational orientation deficit. An important study that falls in line with this approach is [39], where a bivariate Probit model was used to show that the probability of early leaving (in the first semester) is directly correlated to the number of students in the classroom who follow the compulsory courses, while in the subsequent semesters the probability of dropout decreases as the academic performance increases, and therefore the perceived self-regulatory efficacy increases [17, 5].

The studies carried out in relation to the Italian experience, characterized by a high rate of early dropouts, which stood at 12.2% in the transition from the first and second year of the first-level degree courses [9], confirm the presence of a mix of endogenous/exogenous factors that, directly or inversely, are strongly correlated to the risk of dropout. Among those of particular relevance are: the chosen Study Program has a limited number of students, the quality of the freshman orientation programs, the number of students attending the courses and the perceived self-efficacy in the organization of individual study [13, 4, 7, 35].

These methods have the undoubted advantage of making possible the identification of the most appropriate policy guidelines to reduce dropout rates in future cohorts. However, in recent years an alternative approach has become widespread, aiming at predicting the probability of dropout for each student. These methodologies are largely inspired by the churn analysis used in many marketing studies. The churn rate, or attrition rate, is any estimate of the number of individuals who leave a certain group at a defined time interval. The churn analysis techniques aim to identify these individuals early, in order to implement actions at an individual level that increase the retention rate, thus countering dropouts [23, 27].

This specific way of approaching the problem of dropout is an integral part of a broader research field that has emerged over recent years, called *Educational Data Mining* (EDM) [47]. The Data Mining process, also known as *Knowledge Discovery in Databases* (KDD), consists of the automatic discovery through appropriate algorithms of new and potentially useful information hidden within large amounts of data. The EDM is precisely focused on the development of ad hoc methods that can be used to discover regularities and new information within databases from contexts related to education, aimed at better understanding the individual students and the environments within which this instruction is provided, as well as their relation to the expected performance and objectives [3, 36]. The analysis and use of algorithms that extract new knowledge from databases is part of that discipline generally known as *Machine Learning* (ML) [38, 18].

Supervised classification techniques already used in literature and the prediction of dropout are manifold [29]. For example: logistic regression [52], CART algorithms (Classification and Regression Trees) [11, 28], Naïve Bayes [46], Support Vector Machines (SVM) [49], Artificial Neural Networks (ANN) [44, 50]. As it is well known, supervised classification algorithms are ML techniques based on the availability of a training set with complete information, in which for each instance of the problem under study both the classification label (usually a 0/1 binary label) and a set of values relating to s qualitative/quantitative input variables (predictors) are available. Based on this set, the algorithm learns an empirical relationship between the space of the input variables and the label, thus making it possible to predict the label also for new future instances, for which only the input variables are available, while the label must still be observed [20]. In this sense, the term ‘supervised’ indicates precisely the learning based on the information that has already been observed. In our case, the binary label will obviously represent the occurrence or non-occurrence of the dropout. With appropriate coding, we can insert the occurrence/non-occurrence of this event at an individual level within a classification algorithm.

A related problem, immediately after gathering information, lies in choosing the most appropriate algorithm. Comparison between classification techniques, in terms of accuracy, has a solid theoretical basis. We must indeed remember that there is no learning algorithm that systematically obtains the highest performances in any application context. For certain problems, we tend to find that certain algorithms perform better than others, which is a consequence of the algorithm’s fitness to the particular problem, and that the use of domain knowledge can improve performance at the cost of generality [53, 54]. This means that the evaluation of accuracy must be essentially empirical in nature, and that the algorithms compared can hardly ever be selected on the basis of a well-defined criterion, but are the result of a ‘reasonable’ choice that must be in each case validated a posteriori, based on the data.

A fundamental point on which, however, it is necessary to better focus derives from the very nature of the problem we want to address. If the goal is to accurately predict as early as possible the likelihood of leaving the undertaken study course, there are obvious constraints on the variables that can be used as input variables by any classification algorithm. This kind of difficulty is well exemplified in [33]; in this study a multi-stage experiment is described in which the set of input variables has been enlarged sequentially starting from an initial step I. In this first level only six independent variables were used, related to the social status of the students and to the characteristics of the class in which the attendance of the courses of the first semester took place, up to the step V in which among the input variables were also included the marks in the most important exams (plus many other variables, concerning the performance and the individual characteristics of each student). The authors point out that the most accurate predictions in terms of sensitivity and specificity are obtained, as was to be expected, with the input variables of step V, with any classification algorithm among those used.

However, most of the variables of step V can be observed only in a more advanced moment of the university career of each student, and therefore end up being irrelevant for the purposes of an early prediction of the probability of dropout. In other words, the prediction error drops to zero as we approach the adverse event (dropout). All this is of little use for the possibility of implementing an automatic classification system for students at risk that is really effective, and that allows corrective action to be taken at an individual level in a timely manner.

Most of the literature on dropout prediction is affected by problems of this nature. Therefore, results obtained are difficult to reproduce and are often overly optimistic. In light of these difficulties, the research question we want to start exploring in the following study is: how accurately can we predict dropout by using a minimal set of input variables? By *minimal* we mean a small set of individual variables that are immediately available at the time of matriculation and that remain constant throughout the university career (therefore not requiring prospective collection of new data). As in other literature, in our paper we have considered a variable which is available only at dropout or graduation, that is, the age of leaving the university system (age at exit). However, the insertion of this variable should not be considered as a limitation but rather as a tool that makes it possible to evaluate the risk of dropout at any point in time in the course of university career.

The learning of the classifier takes place using the age at exit as an input variable while, at the time of prediction on future students (for whom the degree has not yet been achieved or the dropout has not yet occurred), we use the age actually reached at that time. This covariate shift can be justified on the basis of the characteristics of the distribution of age at exit from the university system (see the exploratory analysis shown in Figure 1), and has a surprising result. While the predictions made using the age at exit have a low to moderate sensitivity (and cannot be used in reality for obvious reasons), those made on the basis of a simulated age prior to the moment in which exit from the system occurs have a significantly greater sensitivity, obtained at the expense of a modest reduction in specificity. In this way, the risk of dropout can be assessed at any time without collecting any new data after matriculation. In order to remove any ambiguity, we agree that students who drop out constitute the *positive class* (i.e. they are labelled '1'), and thus sensitivity is the ability to correctly classify a student who drops out, while specificity is the ability to correctly classify a student who completes the course undertaken.

Finally, as noted earlier, we do not have general theoretical results allowing us to identify the best algorithm for a given problem in advance. The only solution is to empirically evaluate the existing trade-off between generality and predictive accuracy. For this reason, we have considered a set of three supervised classification algorithms characterized by a growing computational complexity, coupled with a cross-validation resampling procedure in order to mitigate the tendency to be overly optimistic about the sensitivity and the specificity.

The paper is organized as follows. In Section 2 we briefly describe the study population and the input variables of the classification algorithms used. The supervised classification algorithms used for predicting dropout are described in Section 3. In Section 4 we have reported some details on data preparation, the experimental design used to evaluate the accuracy in an unbiased way, as well an extensive simulation to demonstrate the effect discussed above. Section 5 contains a brief discussion of the results and suggests the way forward for future research.

2. The data used in this paper have been extracted from the Miur Cineca Student-Didactic Observatory database [37]. This Observatory is specifically reserved for universities, for communication of cases to the National Registry of Students, which is, in turn, made up of a vast administrative archive in which the members of the Italian university system are registered.

The outcome variable is the exit from the system for students enrolled at the University of Bari Aldo Moro, from 2013 to 2016. The possible conditions at the exit covered by the system are: $\mathtt{L} \equiv$ Graduated $(\rightarrow 0)$, $\mathtt{R} \equiv$ Abandonment of Studies $(\rightarrow 1)$, $\mathtt{M} \equiv$ Death. The instances containing the exit code $\mathtt{M}$, in very small numbers, have been preliminary removed from the database. In this way, the classification label becomes a binary variable. The system actually contemplates other possibilities of exiting the system that never occurred among the instances falling within the observation window taken into consideration. Furthermore, transfers to another location are not taken into account in this work, as they are not considered to be real abandonments of the National University System.

In total we have available $N = 41{,}614$ observations and for each available instance we have selected the following five independent variables:

1. `BIRTH REGION`: categorical variable with obvious meaning. One of the levels of

this variable is Abroad, to indicate that the student was born abroad (but is not necessarily of foreign nationality).

2. RESIDENCE REGION: also in this case we have a categorical variable with obvious meaning, as well as also in this case we have an Abroad level to indicate the foreign students who attend the University of Bari.
3. TYPE OF STUDY COURSE: categorical variable that describes the course of study undertaken. The levels of this categorical variable are: LV ≡ old degree, LT ≡ undergraduate degree m.d. 509/99, LS ≡ post-graduate m.d. 509/99, TU ≡ single-cycle degree m.d. 509/99, MT ≡ first level degree m.d. 270/04, MS ≡ master's degree m.d. 270/04, LM ≡ single-cycle master degree m.d. 270/04 (m.d. ≡ Ministerial Decree).
4. SEX: categorical variable indicating the sex of the student.
5. AGE AT EXIT: age at the time of graduation, or dropout. Although this variable is measured at the time when one of the two possible events occurs (degree or dropout), it does not contain any information that allows to determine exactly which of the two events occurred, and therefore can be used to learn, through the classification model, the probability of dropout. As we said before, when we assess the probability of dropout in future students (for which we must make a prediction of the label, L or R) this variable must be populated with the actual age at the time of the prediction, or with age at matriculation if we wish to determine the risk of dropout starting from the moment of matriculation. However, from the point of view of assessing the accuracy of the classification models, this introduces a temporal misalignment between the labels of the training set, which are related to the age of exit from the university system, and the labels guessed by the algorithm, which instead refer to the age for which the class label prediction is made. The consequences of this temporal misalignment are surprising, and will be better examined later in this Section as well as in the Section 4.

As highlighted in the introduction, the choice of these variables is not the result of a data-based selection process assisted by an appropriate algorithm, but rather represents a reasoned choice that allowed us to identify a minimal series of demographic and academic career variables that are immediately available from the administrative databases as of matriculation, and do not change during the student's career. For example, the combined use of the BIRTH REGION and RESIDENCE REGION seeks to capture the impact on the outcome variable of aspects such as: being born abroad, family mobility during adolescence and studying away from home. These aspects could be better described by variables having a finer spatial resolution, such as, the province and residence of birth. However, in this case, the computational explosion would make the problem intractable (except when using simple structure classification models, having lower accuracy).

It is worth nothing that age at matriculation was not available at the time of writing this paper. Moreover, other variables could help improve the quality of predictions. For example an individual performance measure such as the high school final grade could prove to be of great importance. These aspects are further discussed in Section 5, in light of the results obtained. However, the use of such a limited set of independent variables has also the function of stressing under a well designed simulation the use of classification models study when a large information set is lacking. If we can achieve acceptable results in this situation, we will certainly be able to provide much more accurate (and usable) predictions when more data are available.

To examine the main characteristics of the study population, in this Section we will indicate the outcome variable as `CLASS` (with the two labels `L` and `R`). First of all, from Figure 1, which contains a brief exploratory analysis, it is clear that there is not too extreme imbalance in favor of the female sex in terms of numerical consistency (top left panel). In fact, among graduate students, women are 67% compared to 33% of men, while among those who give up, women are 54.7% against 45.3% of men (in total, female students represent roughly 64% of the total).

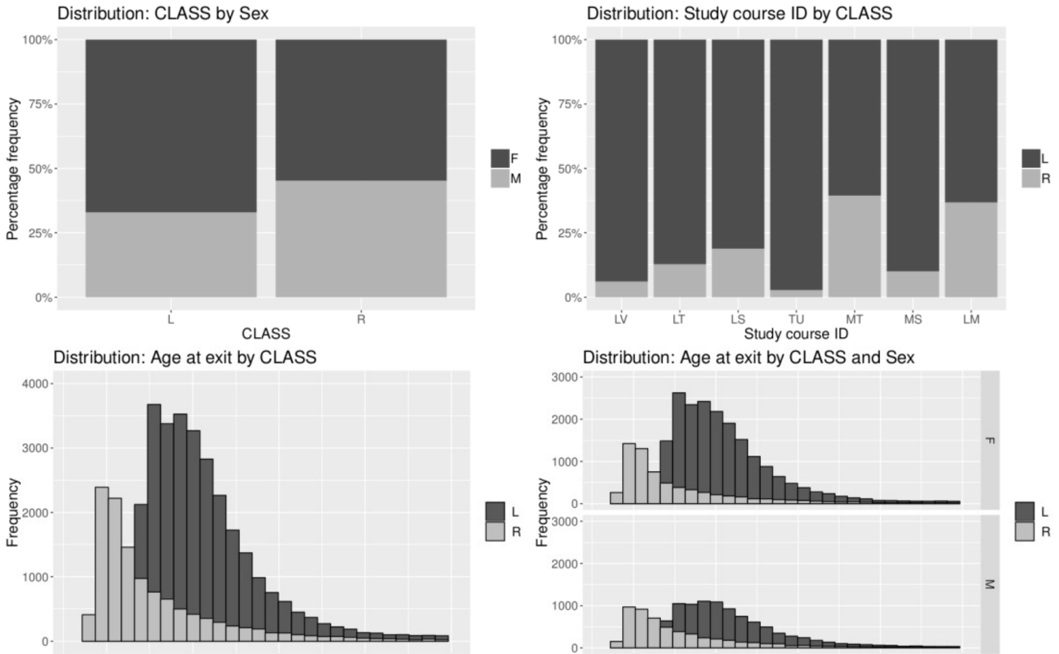

Figure 1. Graphical exploratory data analysis of outcome variable and some of the chosen predictors. The levels of `TYPE OF STUDY COURSE` categorical variable are: `LV` ≡ old degree, `LT` ≡ undergraduate degree m.d. 509/99, `LS` ≡ post-graduate m.d. 509/99, `TU` ≡ single-cycle degree m.d. 509/99, `MT` ≡ first level degree m.d. 270/04, `MS` ≡ master's degree m.d. 270/04, `LM` ≡ single-cycle master degree m.d. 270/04 (m.d. ≡ Ministerial Decree).

At the top right in Figure 1 we show the distribution of the relative frequencies of the condition at the exit, stratified by the type of Degree Program. It is interesting to observe the good perfromance occurred with the study courses belonging to the old four-year systems (LV, graduation 93.9% against 6.1% of dropouts). It is worth noting that students still enrolled in the old four-year course (which hs been inexistent for several years) are almost certainly strongly motivated to complete their course of study. On the other hand, the best figures can be seen in the single-cycle post graduate study courses of the old system pursuant to Ministerial Decree 509/99 (TU, 97.3% of graduates, against 2.9% of dropouts during the period considered). In this case too, motivation plays a fundamental role: the majority of students enrolled in study courses in this system are students of the one-cycle degree in Medicine. It is therefore evident that also in this case the will to complete the studies plays a role: the observed dropouts are not of the same nature as the youthful ones, which occur between the first and second year of the course, but are due to the occurrence of circumstances that make the continuation of studies impracticable, despite the motivation. Although these considerations have an independent importance for the analysis of the phenomenon, it is however not clear what impact they can have on the accuracy of dropout prediction.

The analysis of age at the exit from the university system is much more interesting. In general, the distribution of graduate students presents an apparent positive asymmetry (Figure 1, bottom-left panel; data are expressed in years and fractions of a year: min = 20.6, Q_1 = 23.5, median = 25.1, average = 26.3, Q_3= 27.4, max = 77.1). The average age at graduation is 26.3 years, and 25% of students achieve a degree above 27.4 years old, up to a maximum of 77.1 years. The distribution of students who give up their studies has substantially the same form even if, obviously, the measures of synthesis are quite different in the left part of the distribution (min = 18.1, Q_1 = 20.2, median = 21.4, average = 23.3, Q_3= 24.4 , max = 72.6). The average age of waivers is therefore 23.3 years, but 25% of students give up at an age equal to or less than 20.2 years, thus confirming the dramatic importance the dropout phenomenon plays in the transition from the first to the second year of the course of studies. It is equally important to note that the global average age is 25.9 years. As demonstrated in Section 4, AGE AT EXIT is the variable of greatest importance in terms of impact on predictive accuracy, while all the other input variables entering the training set have a decidedly more modest importance. Therefore, the classifier will tend to replicate the distribution of CLASS around the global average age. Since around the global average age we have a clear prevalence of graduate students we expect, as we actually find, that the predictions have a high specificity (i.e. that they correctly identify the majority of graduates) and a moderate to low sensitivity (i.e. a substantial percentage of those who leave is not correctly identified by the algorithm).

However, we know that the predictions must necessarily be made using an age before the age of exit (the latter can only be used in the learning phase). This is equivalent to moving to the left along the distribution of ages at exit, in a range in which the weight of the students who dropout becomes increasingly important. With this simple device we expect (as indeed happens: see Section 4 for further details) that the predictions on the test set gain significantly in sensitivity, to the detriment of a moderate reduction in specificity.

Finally, in the bottom-right panel, we report the same data disaggregated by gender: it is clear that, for both sexes, the age distribution at exit shows the same qualitative behavior that we have just discussed. A substantial difference between the genders is not apparent, regarding age at graduation (average = 26.7 years for men, average = 26.1 years for women). On the other hand, even taking into account the split between the genders, the average age of waivers is drastically lowered for both (average = 23.5 years for men, average = 23.1 years for women, with 25% of men who give up at an age of less than or equal to 20.3 years, and 25% of women at an age of less than or equal to 20.1 years). This is a further signal that AGE AT EXIT has an important weight in determining the dynamics of the abandonments.

We will not insist further on the exploratory analysis of the 'gepgraphical' variables BIRTH REGION and RESIDENCE REGION, as they assume too many distinct values (the geographical dimension is not considered relevant in itself). Rather, we want to emphasize again that although the dropout of university studies is a complex phenomenon that is often based on an individual decision, it is clear that there are regularities that can, at least in principle, be used for early identification of student at risk, before the decision to end university studies is actually implemented. What we want to quantify is how accurately this process can be carried out, using the minimal information set we have described above.

3. In what follows, let c denote the binary label of the outcome variable CLASS associated with each student, with $c \in \{0,1\}$, the positive class $c = 1$ indicating a dropout (or, as before, L $\rightarrow 0$, R $\rightarrow 1$). For each student, a feature vector $\mathbf{x}^{\top} = (x_1, \dots, x_s) \in X \subseteq \mathbb{R}^s$ of s input variables is also available. The effective dimension s of the feature vector $\mathbf{x}$ depends on the way categorical predictors are encoded with numerical values. We will come back to this issue later in next Section. The optimal theoretical classifier under a 0/1 loss function has the following form [20]:

$$\gamma(\mathbf{x}) = \underset{c \in \{0,1\}}{\operatorname{argmax}} \quad p(c|\mathbf{x}). \tag{1}$$

Each classification algorithm differs from the others in the specific process by which the training data are used to estimate the posterior probabilities in the classes. Once this process has been completed, we have the empirical classifier $\hat{\gamma}(\mathbf{x})$, which represents the empirical counterpart estimated on the theoretical optimal classifier (1). We will now briefly describe the algorithms used to extract the results from the available data.

3.1. Logistic regression is a discriminative classifier, which learns conditional probabilities $p(c|\mathbf{x})$ directly, instead of providing a model of how the data are actually generated. In another way, we can say that discriminative classifiers learn a direct map from inputs $\mathbf{x}$ to class labels [40]. In logistic classification, theoretical posterior probabilities have the following form (for $i = 1, \dots, N_{tr}$, where N_{tr} denotes the dimension of the training set):

$$p(1|\mathbf{x}^{(i)}, \mathbf{w}) = \frac{\exp\left(w_0 + \sum_{j=1}^{s} w_j x_j^{(i)}\right)}{1+\exp\left(w_0 + \sum_{j=1}^{s} w_j x_j^{(i)}\right)}, \tag{2}$$

$$p(0|\mathbf{x}^{(i)}, \mathbf{w}) = \frac{1}{1+\exp(w_0 + \sum_{j=1}^{s} w_j x_j^{(i)})} = 1 - p(1|\mathbf{x}^{(i)}, \mathbf{w}). \tag{3}$$

With a few algebraic manipulations, the optimal theoretical classifier for a new instance, whose input vector is $\mathbf{x}^{new}$, assumes the following form:

$$\gamma(\mathbf{x}^{new}, \mathbf{w}) = 1 \qquad \text{iff} \qquad w_0 + \sum_{j=1}^{s} w_j x_j^{new} > 0. \tag{4}$$

Parameter estimation usually relies on the maximization of the following Bernoulli conditional log-likelihood over the training set:

$$\widehat{\mathbf{w}}_{opt} \longleftarrow \underset{\mathbf{w}}{\text{argmax}} \;\; \sum_{i=1}^{N_{tr}} \log p(c^{(i)}|\mathbf{x}^{(i)}, \mathbf{w}), \tag{5}$$

where $\mathbf{w}^\top = (w_0, w_1, \dots, w_s)$ is the parameter vector. A suitable numerical method for solving optimization problem (5) is known as Iteratively Reweighted Least Squares (ILRS, for further details see [45]). Obviously, once the training data have been used to obtain parameter estimates $\widehat{\mathbf{w}}_{opt}$, we plug-in them into the form of the optimal classifier (1) to obtain its empirical version:

$$\hat{\gamma}(\mathbf{x}) \equiv \gamma(\mathbf{x}, \widehat{\mathbf{w}}_{opt}). \tag{6}$$

3.2. Support vector machines (SVM) is the second learning algorithm considered. Formally, the SVM problem consists in finding a decision hyperplane:

$$\langle \mathbf{w}, \mathbf{x} \rangle + b = \mathbf{w}^\top \mathbf{x} + b = 0, \tag{7}$$

where $\mathbf{w}^{\top} = (w_1, \dots, w_s)$ in this case. The optimal decision boundary is described by the following constrained quadratic optimization problem (the classification label is coded as $y^{(i)} = +1 \rightarrow \mathtt{R}$, $y^{(i)} = -1 \rightarrow \mathtt{L}$), leading to an optimal classifier with a large geometric margin around the decision boundary:

$$\widehat{\mathbf{w}}_{opt} \longleftarrow \underset{\mathbf{w}}{\operatorname{argmin}} \quad \frac{1}{2}||\mathbf{w}||^2 + C\big(\textstyle\sum_{i=1}^{N_{tr}} \xi_i\big), \qquad \text{subject} \quad \text{to:} \tag{8}$$

$$y^{(i)}\big[\langle \mathbf{w}, \phi(\mathbf{x})^{(i)} \rangle + b\big] \geq 1 - \xi^{(i)}, \qquad \text{for} \quad i = 1,2 \dots, N_{tr}. \tag{9}$$

In the above expressions, $\xi^{(i)} \geq 0$ are slack variables allowing for non-empty feasible solution even in cases where the two classes are not separable by a hyperplane [25, 32]. The sum of the ξ_i gives an upper bound on the number of training errors. The cost parameter C is a regularization term, which provides a way to control overfitting by trading off training errors against the width of the geometric margin.

Finally, $\phi: X \longrightarrow F$ is a map from the input space to a feature space, introduced to improve the linear separability of the training input vectors in the transformed feature space. Omitting the details, both the quadratic programming problem solution and the final classification function depend only on dot products between input vectors in the transformed feature space. Thus, if we have a way to compute the inner product $\langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$ in the feature space F using the input vectors directly, then we would not need to know the transformed feature vector $\phi(x)$ or even the mapping function ϕ itself [22]. In SVM, this is done through the use of kernel functions, denoted by K:

$$K(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle, \qquad \forall \quad \mathbf{x}, \mathbf{x}' \in X. \tag{10}$$

Not every function $K(\mathbf{x}, \mathbf{x}')$ is a valid kernel function satisfying expression (10) for at least one map ϕ. Mercer's conditions that a function K has to fulfill to be a valid kernel function are discussed, among the others, in [32], pag. 332. In this paper we experiment with the two following kernels:

- the linear kernel, that is the simplest of all kernel functions (input vectors are not transformed): $K(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle$.
- the Gaussian radial basis kernel: $K(\mathbf{x}, \mathbf{x}') = \exp(-\sigma||\mathbf{x} - \mathbf{x}'||^2)$, which is equivalent to mapping the input vectors into an infinite dimensional Hilbert space, and depends on a scaling parameter σ.

SVMs often lead to good generalization performance because of the large-margin separation principle, which was been introduced by V. Vapnik in its seminal work on statistical learning theory [51]. The final form of the empirical classifier is:

$$\hat{\gamma}(\mathbf{x}^{new}) \equiv \gamma(\mathbf{x}^{new}, \hat{\mathbf{w}}_{opt}, \hat{b}) = \text{sign}(\langle \hat{\mathbf{w}}_{opt}, \mathbf{x}^{new} \rangle + \hat{b}), \tag{13}$$

with $\text{sign}(z) = +1$ if $z > 0$, and $\text{sign}(z) = -1$ if $z \leq 0$.

3.3. A multilayer perceptron (MLP) is a feedforward artificial neural network consisting of at least three layers of nodes: an input layer, a hidden layer and an output layer. Formally, a MLP is a nonlinear input-output mapping $y = F(\mathbf{x})$, where $\mathbf{x} \in X$ and the output y can be either discrete or continuous (that is $y \equiv c \in \{0,1\}$ in our case).

To construct a MLP we start with composing blocks of hidden layers [42, 15]. Let $f_1, \dots, f_L$ be univariate activation functions. An activation rule is given by:

$$f_\ell^{W_\ell, b_\ell} = \left(f_\ell\left(\sum_{j=1}^{N_{\ell-1}} w_{1,j}^{(\ell)} z_j^{(\ell-1)} + b_1^{(\ell)}\right), \dots, f_\ell\left(\sum_{j=1}^{N_{\ell-1}} w_{N_\ell,j}^{(\ell)} z_j^{(\ell-1)} + b_{N_\ell}^{(\ell)}\right)\right)^\top, \tag{14}$$

where $W_\ell \in \mathbb{R}^{N_\ell, N_{\ell-1}}$ and $\mathbf{z}_{\ell-1}^\top = (z_1^{(\ell-1)}, \dots, z_{N_{\ell-1}}^{(\ell-1)}) \in \mathbb{R}^{N_{\ell-1}}$ are the weight matrix and inputs of the ℓth hidden layer. In the same way, the bias/intercept vector is expressed as $\mathbf{b}_\ell^\top = (b_1^{(\ell)}, \dots, b_{N_\ell}^{(\ell)})$. Of course, for $\ell = 1$ we have $\mathbf{z}_0 \equiv \mathbf{x}$ and $N_0 \equiv dim(\mathbf{x}) = s$. A popular choice for the activation function is the sigmoid function $f_\ell(z) = (1 + \exp(-z))^{-1}$. Thus, MLPs use composition of a series of simple nonlinear functions to model nonlinearity:

$$\left(f_L^{W_L, b_L} \circ \dots \circ f_1^{W_1, b_1}\right)(\mathbf{x}). \tag{15}$$

In a two class problem, given an output from the final hidden layer L we have two output neurons, one for each binary label $y \equiv c \in \{0,1\}$. In order to ensure that the outputs can be interpreted as posterior probabilities, they must be comprised between zero and one and sum to one. In practice this is achieved by using a softmax activation function having the same functional form of (2) and (3):

$$p_1^{(i)} \equiv p(1 | W_{L+1}, \mathbf{z}_L, \mathbf{b}_{L+1}, \mathbf{z}_0^{(i)}) = \frac{\exp\left(\sum_{j=1}^{N_L} w_{1,j}^{(L+1)} z_j^{(L)} + b_1^{(L+1)}\right)}{1 + \exp\left(\sum_{j=1}^{N_L} w_{1,j}^{(L+1)} z_j^{(L)} + b_1^{(L+1)}\right)}, \tag{16}$$

$$p_0^{(i)} \equiv p(0 | W_{L+1}, \mathbf{z}_L, \mathbf{b}_{L+1}, \mathbf{z}_0^{(i)}) = \frac{1}{1 + \exp\left(\sum_{j=1}^{N_L} w_{1,j}^{(L+1)} z_j^{(L)} + b_1^{(L+1)}\right)}, \tag{17}$$

for $i = 1, \dots, N_{tr}$. The loss function to minimize over the training set is the negative cross-entropy between the labels y_{ik} in the form of two-dimensional one-hot indicator vector

($y_{ik} = 1$ if example i is associated with class $k \in \{0,1\}$, or $y_{ik} = 0$ otherwise) and the score vector $\mathbf{p}^\top = (\mathbf{p_1}, \dots, \mathbf{p}_{N_{tr}})$, with $\mathbf{p}_i^\top = (p_0^{(i)}, p_1^{(i)})$:

$$C(\mathbf{p}) = -\sum_i \sum_{k\in\{0,1\}} y_{ik} \log p_k^{(i)}. \tag{18}$$

If the neuron's actual output is close to the desired output for all training inputs then the cross-entropy will be close to zero. The Multinomial negative log-likelihood (18) always strongly penalizes the most incorrect predictions, and the cost-based training becomes a form of winner-take-all (one of the two outputs is nearly 1, and the other is nearly 0; [19]). Finally we can use the trained network for prediction using new inputs, much in the same way as logistic regression.

Training can be carried out via batch or stochastic gradient descent with backpropagation for efficient gradient computation [48, 1]. Although ANNs are very powerful to solve a wide variety of complex problems, such systems require manual configuration and tuning. The learning algorithm itself has two free parameters: the learning rate η, which specifies the gradient descent step width, and the maximum output difference d_{max}, that defines how much difference between output and target value is treated as zero error and not back-propagated. They both control the degree of regularization and can be suitably varied to prevent overfitting [6]. Furthermore, the network architecture must be specified by choosing the number of hidden layers, as well as the number of neurons for each of these hidden strata. We will come back to these issues later in the next paragraph.

4. Since the data available includes categorical variables, these must be appropriately processed to enter into the classification models. For the logistic model the training matrix $X \in \mathbb{R}^{N_{tr},(s+1)}$ has to be of full column rank, where $N_{tr} > (s+1)$ and the first column is a vector of ones allowing intercept estimation. Otherwise, if a model matrix has linearly dependent columns the crossproduct matrix $X^\top X$ is singular and the parameter estimates are not unique. For this reason, we used a standard dummy coding with logistic classification, that is for all but one (the reference level) of the levels of the categorical variable a new variable will be created that has a value of one for each observation at that level and zero for all others, whereas the reference level will be coded with a vector of zeros [34].

On the other hand, it also makes perfect sense that each column of the resulting model matrix X corresponds to one unique value of each original categorical value. The simplest way to do this consist of encoding each level of a categorical variable by using a binary indicator column (one-hot encoding; [10]). It is a singular parameterization because if

$X_1, X_2, \dots, X_k$ are the binary columns that encode the k levels of a categorical variable then $\sum_k X_i = 1$ for each instance. Therefore, if in the matrix X there is a vector of ones in the first column (such as a bias neuron), then $X^\top X$ will inevitably be singular. However, the algorithms determining an optimal solution for an SVM or an MLP are perfectly usable even in the presence of a singular $X^\top X$. Therefore, for these two algorithms we will always use a one-hot encoding, so that the size of the support vectors (in the case of SVM) or the number of input neurons (in the case of MLP) is exactly equal to the total number of distinct levels of categorical input variables plus the number of continuous input variables. In our case, the only continuous variable is `AGE AT EXIT`, and using a one-hot encoding the dimension of the input space becomes $s = 53$. Subsequently, we have eliminated some instances for which the data were missing or affected by unrecoverable errors on at least one variable. After this step, the size of the dataset was reduced to $N = 41{,}950$ instances. At this point, we have randomly divided the entire dataset into a training and a test subset, using a 90%-10% splitting, obtaining the following numbers for the two subsets: $N_{tr} = 37{,}755$, $N_{te} = N - N_{tr} = 4{,}195$. The `AGE AT EXIT` variable was standardized on the training set, and subsequently standardized also on the test set using the mean and deviation standards calculated on the training set.

On the training set we estimated the accuracy by resampling, using a 10-fold cross-validation (CV). In particular, on the training set we calculated (taking the average of the values obtained in each of the 10 folds of the training set used as a test set during the CV procedure) the Area Under the Curve (Auc) associated with the ROC curve [16], as well as sensitivity and specificity [41]:

$$Sens = \frac{TP}{TP+FN}, \tag{19}$$

$$Spec = \frac{TN}{TN+FP}, \tag{20}$$

where TP = True Positives (i.e. the number of actual students abandoning the course of study undertaken that are correctly identified as such) and, obviously, FN = False Negatives, TN = True Negatives and FP = False Postivies.

Sensitivity measures the fraction of students, among dropouts, correctly identified by the algorithm. On the other hand, the specificity measures the fraction of students, among all those who have achieved the qualification, which are correctly classified by the algorithm. Optimizing for sensitivity or specificity obviously means pursuing different objectives, and there is a trade-off between the two measures, in the sense that optimizing for one of the two generally means reducing the value of the other. However, greater sensitivity is obviously the most important goal to achieve, since greater sensitivity corresponds to a greater ability to correctly identify the students who leave.

The test set was used as a validation set to measure the predictive capabilities of the model in realistic situations. As already widely observed, the `AGE AT EXIT` variable can only be used in the learning phase and certainly cannot be used to make predictions about future students. Therefore, we calculated the sensitivity, specificity and accuracy, $Acc = (TP + TN)/N_{te}$, both on the original test set (indicated with T, where T refers precisely to age on leaving the university system), and on a collection of artificial test sets obtained in the following way:

- $T - 0.5$: the test age was brought back one semester with respect to the age at exit, with the only condition that $T - 0.5$ could not be lower than the youngest age at exit actually observed. In this way, we evaluate the accuracy of the predictions made six months before the event (degree or dropout).
- $T - 1.0$. We evaluate the accuracy of predictions made two semesters before the event
- $T - 2.0$. As above, we evaluate the accuracy of predictions made four semesters before the event.
- $T + U(-2,0)$. The three artificial test sets we have created so far correspond to situations that are still not sufficiently realistic, because they presuppose testing the accuracy at a certain time point prior to the event, which however is exactly the same for all students. Obviously, this is once again equivalent to assuming the age at exit is known for all future students. For this reason, we have created a new test set in which the test age was brought back up to four semesters with respect to the age at exit, adding a uniform random impulse over the interval $[-2,0]$. In this way, we calculate the accuracy of the forecasts made up to four semesters before the event. Furthermore, the age at which the prediction is carried out is random, and is no longer the same for all students.
- $T + U(-3,0)$. As above, we calculate the accuracy of the forecasts made up to six semesters before the event. The age at which the prediction is carried out is random, and is no longer the same for all students.
- $T + U(-2, +2)$ In this case, we calculate the accuracy of forecasts made up to four semesters before and after the event (i.e. we are moving, randomly, both to the left and to the right along the `AGE AT EXIT` distribution). In this way, the relationship between the age at the exit and the label, learned during the training phase, is partially broken in the test phase by this random bidirectional shift.
- T_{boot}. The test ages are obtained by a complete reshuffling of the ages at the exit, taking a bootstrap sample. In this way, the relationship between the age at the exit and the label, learned during the training, is completely broken in the test phase.

Finally, in the presence of control parameters of classification models, using the supervised learning infrastructure provided by the R `caret` library (v.6.0-82; [30, 43]), we made these parameters vary in an appropriate range, and we chose the final model as the one which had the highest sensitivity (calculated on the training set by CV in the way described above). In particular:

- The logistic classifier has no control parameters.
- For SVMs with linear kernels we have the cost parameter C, which governs the width of the geometric classification margin. It has been set equal to $C = 2^k$, with $k = -2, -1, \dots, 7$, from a minimum of $C = 0.25$ to a maximum of $C = 128$.
- For SVMs with radial kernels we have the cost parameter C and the scaling parameter σ. The latter was preliminarily estimated on the training set using the empirical rule described in [8], while C was made to vary exactly as in the previous case. In both cases (linear and radial kernel) we used the standard `kernlab` library (ver 0.9-27; [24]).
- For the MLP we have chosen to consider at most three hidden layers of neurons, based on the trade-off between the generalization capabilities and the relative computational cost. By indicating with (size $_1$, size $_2$, size $_3$) the number of neurons present in each of the three hidden levels, we have chosen: size$_1$ $\in \{5,10,15,20\}$, size$_2$ $\in \{5,10,15,20\}$ and size$_3$ $\in \{0,5,10,15,20\}$, for a total of 80 distinct models. The fact that the outermost layer can have a number of neurons equal to zero obviously means that the architecture of the network can be simplified to include only two hidden layers. Regarding the algorithmic parameters, the final results that we present in the next section do not include any form of weight decay (since, in our experience, its use did not improve accuracy in any way). Furthermore, by manual tuning we set $\eta = 0.2$ for the learning parameter and $d_{max} = 0$ for the maximum output difference. The weights of the connections between the units have been randomly initialized. We used the `RSNNS` library (ver 0.4-11; [6]).

The results obtained are reported in Table 1. Since for the MLP we had 80 possible models, a graphical summary of the sensitivity assessed on the training set by CV for each of the possible models is reported in Figure 2. As seen in Table 1, the logistic classifier has a sensitivity not exceeding 32% (on the test set T): in other words, although the Auc is roughly equal to 0.79 the algorithm does not manage to identify more than 32% of the students who actually leave the study program. We observe a better behavior using an SVM with linear kernel: when the value of the cost parameter is high, the sensitivity fluctuates around 35%; when, instead, the value of the cost parameter becomes lower than $C = 1$, the sensitivity increases up to 55%.

Logistic classifier	**Param**	**Acc**	**Auc**	**Sens**	**Spec**
(training set)	(none)		0.7861	0.3050	0.9659
(test set T)	(none)	0.7795		0.3244	0.9683
SVM with linear kernel					
(training set)	$C = 0.25$	← best	0.7864	0.5528	0.9466
	$C = 0.5$		0.7870	0.5042	0.9664
	$C = 1$		0.7878	0.4213	0.9880
	$C = 2$		0.7881	0.3752	0.9943
	$C = 4$		0.7882	0.3600	0.9954
	$C = 8$		0.7883	0.3555	0.9959
	$C = 16$		0.7884	0.3515	0.9962
	$C = 32$		0.7884	0.3502	0.9962
	$C = 64$		0.7884	0.3491	0.9964
	$C = 128$		0.7885	0.3531	0.9962
(test set T)	$C = 0.25$	0.8329		0.5463	0.9518
SVM with radial kernel					
(training set)	$C = 0.25$		0.7807	0.5351	0.9413
	$C = 0.5$		0.7852	0.5338	0.9659
	$C = 1$		0.7866	0.5284	0.9727
	$C = 2$		0.7887	0.5320	0.9744
	$C = 4$		0.7908	0.5392	0.9739
	$C = 8$		0.7922	0.5530	0.9747
	$C = 16$		0.7932	0.5584	0.9749
	$C = 32$		0.7942	0.5615	0.9752

	$C =$ 64		0.7953	0.5665	0.9749
	$C =$ 128	← best	0.7967	0.5682	0.9747
(test set T)	$C =$ 128	0.8532		0.5577	0.9757
Multilayer perceptron					
(training set)	(15,20	← best	0.8636	0.5975	0.9683
(test set)	(15,20	0.8625		0.5504	0.9919

Table 1. Detail of the results. The final value of tuning parameters (Param) has been determined by grid search, optimizing with respect to sensitivity

We know that when the value of C is high the training error assumes a significant weight within the cost term $C(\sum_i \xi_i)$ included in the objective function (8). Therefore, the number of training errors is reduced at the expense of the width of the geometric margin, which decreases. However, sensitivity and specificity are evaluated by CV: this means that the reduction in the number of training errors causes overfitting, which is immediately noticeable.

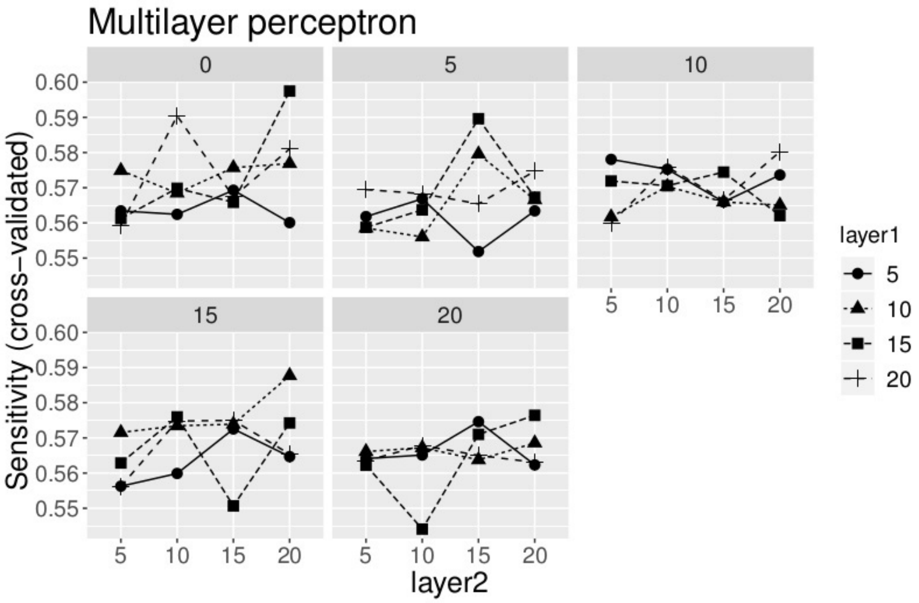

Figure 2. Determination of the optimal sensitivity for the Multilayer Perceptron (MLP). The architecture of the network includes three hidden layers having a number of neurons respectively equal to (size$_1$, size$_2$, size$_3$), with size$_1$ $\in \{5,10,15,20\}$, size$_2$ $\in \{5,10,15,20\}$ and size$_3$ $\in \{0,5,10,15,20\}$. For each of the possible models sensitivity was evaluated on the training set using a 10-fold Cross-Validation (CV).

If we use a SVM with a radial kernel we can observe a slightly higher accuracy. For high C values we do not run into overfitting problems and the sensitivity reaches a maximum of 57% at $C = 128$. In other words, in about six cases out of ten we are able to correctly identify the students who leave the course of study. The optimal Auc is roughly equal to 0.80, and therefore is much higher than that of the random classifier (for which Auc = 0.5). It is to be noted that the sensitivity of the final model is substantially preserved on the test set T (we have a sensitivity of 55% with an overall accuracy of 85%). Looking at the results related to the MLP (Table 1 and Figure 2) sensitivity is even slightly better. There are no obvious signs of overfitting as network complexity changes. It is important that the third level is automatically simplified at the model determination phase. Also in this case we have a sensitivity drop to 55% on the test set T.

Classifier	**Test set**	**Sens**	**Spec**
Logistic	T	0.3244	0.9683
	$T - 0.5$	0.3821	0.9454
	$T - 1.0$	0.4293	0.9201
	$T - 2.0$	0.4374	0.8793
	$T + U(-2,0)$	0.4285	0.9207
	$T + U(-3,0)$	0.4472	0.9008
	$T + U(-2,+2)$	0.3659	0.9528
	T_{boot}	0.1301	0.9285
Linear SVM	T	0.5463	0.9518
	$T - 0.5$	0.5984	0.9221
	$T - 1.0$	0.6463	0.8880
	$T - 2.0$	0.7138	0.7444
	$T + U(-2,0)$	0.6390	0.8668
	$T + U(-3,0)$	0.6715	0.7997
	$T + U(-2,+2)$	0.5317	0.9315
	T_{boot}	0.2244	0.8108
Radial SVM	T	0.5577	0.9757
	$T - 0.5$	0.6057	0.9417
	$T - 1.0$	0.6520	0.8712
	$T - 2.0$	0.7081	0.7137
	$T + U(-2,0)$	0.6455	0.8624
	$T + U(-3,0)$	0.6675	0.7976
	$T + U(-2,+2)$	0.5439	0.9207
	T_{boot}	0.1870	0.8779

MLP	T	0.5504	0.9919
	$T - 0.5$	0.6041	0.9572
	$T - 1.0$	0.6553	0.8759
	$T + U(-2{,}0)$	0.6512	0.8624
	$T + U(-3{,}0)$	0.6821	0.7703
	$T + U(-2, +2)$	0.5431	0.9261
	T_{boot}	0.2024	0.8030

Table3. Sensitivity and specificity calculated on the original test set, and on each of the artificial test sets we created (see text for more details).

In Table 2 and Figure 3 we compared the sensitivities and specificities calculated on the test set T. The age used for the prediction in each of these test sets is misaligned with the age at exit. The effect, however, is exactly what we already anticipated in Section 2. While the sensitivity and specificity on the set T test remain below 60%, as we move to the left along the AGE AT EXIT distribution, sensitivity increases consistently to the detriment of a modest reduction in specificity.

For example, using the MLP and making predictions up to six semesters before graduation ($T + U(-3{,}0)$), we find that sensitivity rises to 68%, while specificity drops to 77%. This means that almost seven students among those who dropout are identified correctly, while two out of ten graduating students are incorrectly classified as being at risk of dropout.

However, the latter (error of second type) is less serious, since it wpuld only lead to to an inefficient allocation of resources to students who would have reached the goal of graduating in any case. Error of first type, on the other hand, leads to not intervening in aid of those students who will abandon the course of study undertaken.

Note that using the test sets $T + U(-2, +2)$ and T_{boot} sensitivity drops markedly. This effect is particularly relevant when we test on T_{boot}, for which sensitivity falls below 20%. It is evident that the random reshuffling induced by the bootstrap sampling on the ages at exit, renders unusable the relation learned in the training phase between the labels and the ages at exit. The same problem is found with $T + U(-2, +2)$, even if the amplitude of the reduction in sensitivity is less, since the mixing of the ages at the exit caused by the forward shift is less extreme than the random bootstrap reshuffling.

Instead, when we move to the left along the AGE AT EXIT distribution, we always observe an increase in sensitivity in the test phase. Therefore, using the actual ages at the time of the forecast makes the classifier learned usable even in real situations, and we have an improvement in accuracy that is certainly not due to a *data dragging* phenomenon.

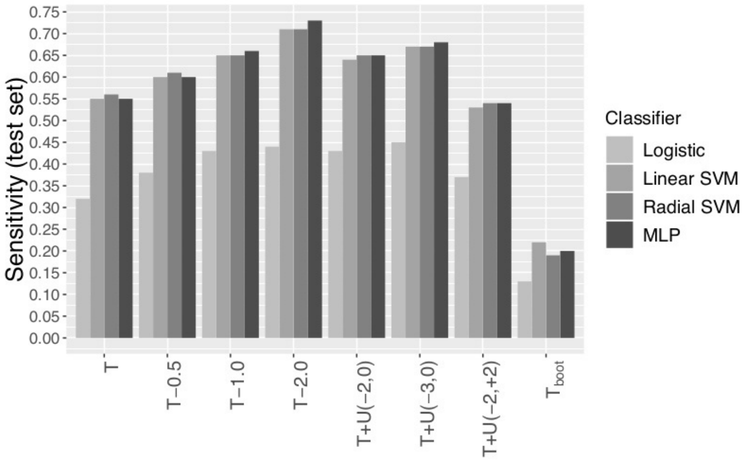

Figure 3. Sensitivity and specificity calculated on the original test set and on each of the artificial test sets we created (see the text for more details), represented graphically based on the type of classification algorithm used.

The fact that `AGE AT EXIT` has such a significant impact on classification can easily be confirmed by conducting an analysis of the importance of each input variable. To this end, we first applied the logistic regression model used for the logistic classifier in two distinct ways. First, we estimated the model on the entire training set and subsequently removed one input variable at a time (in-sample method). Then we tested the reduced model against the full model using an asymptotic likelihood ratio test (Analysis of Deviance; [2]). Once the relative p-value was obtained, the importance index of the tested variable was calculated as $100(1-p)\%$: the results obtained are shown in Figure 4. The second method (out-of-sample) consists in evaluating the accuracy on the test set using a set of models estimated on the training set in which, in turn, one of the input variables was excluded. The results shown in Figure 5 are an exemplification of the principle that goodness of fit and predictive accuracy are not correlated. In fact, all the input variables resulted to be important in terms of their contribution to goodness of fit, and all of them achieved in sample importance indices close to 100%. On the other hand, looking at predictive accuracy, the only variable that has the largest effect on accuracy is `AGE AT EXIT`. By eliminating this input variability the accuracy obtainable on the test set using the full model (approximately 78%) decreases to 70% (and the relative confidence intervals are not overlapping). A similar effect is not observed with all the other input variables.

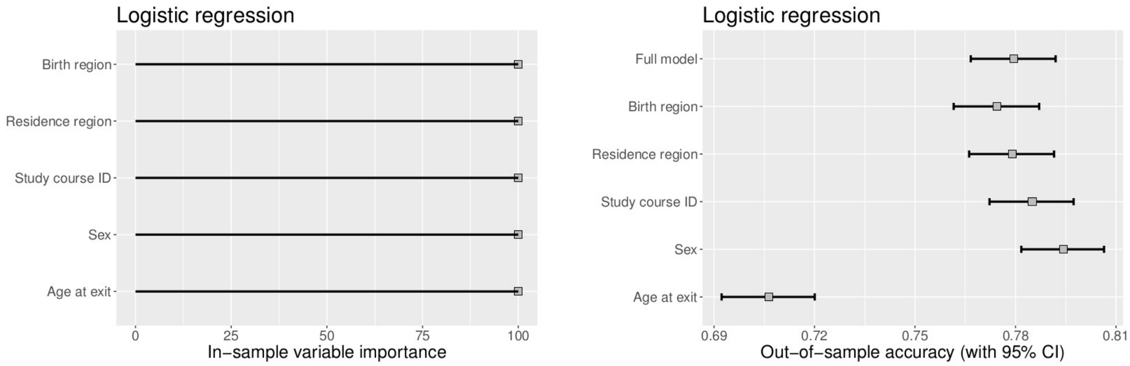

Figure 4. Importance of input variables. The importance was calculated using the logistic regression model used for the logistic classifier. See the text for details.

This parametric analysis was confirmed by the in-sample non-parametric analysis shown in Figure 5. It was obtained by calculating the ROC curve for the `CLASS` outcome variable, separately for each input variable (using the training set and varying the cut-off value to predict the two levels of the outcome variable). The reported measure of importance is the Auc. As it can be seen, once again `AGE AT EXIT` has the greatest importance, confirming that there are some critical moments during the university career linked to a typical chronological age (such as, for example, the passage between the first and the second years for students who matriculate at the University immediately after high school, which takes place on average around the age of 20), where the risk of dropout is maximum. The role of the other input variables is decidedly secondary, while the impact of `BIRTH REGION` and `RESIDENCE REGION` is absolutely irrelevant.

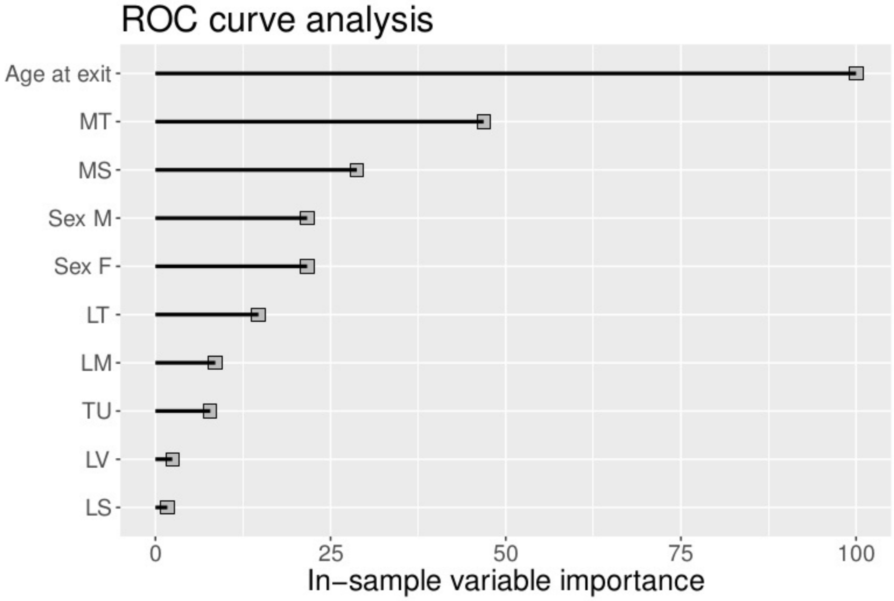

Figure 5. Importance of input variables. In this case, importance was obtained in a non-parametric way by calculating the ROC curve for the CLASS outcome variable, separately for each input variable. The reported measure of importance is the Auc.

5. The use of artificial intelligence and ML has caused a real paradigm shift in statistical science over the last 10 years [14]. The fusion of data science with the analysis of the empirical evidence coming from the education sector has generated a new research field known as Educational Data Mining (EDM). One of the areas in which EDM can play an important role is the early identification of students who are at risk of leaving university studies [12, 21]. In many countries, including Italy, this issue is particularly relevant and there is a vital need to develop information systems allowing for early decisions to be made to support students at risk.

In this work we applied some ML techniques to the early prediction of dropouts in a group of students of the University of Bari Aldo Moro. The results we have reported are preliminary, being based on a simulation study using a *minimal* set of input variables that included only a few variables immediately available at the time of matriculation, plus a time-dependent variable, i.e. the age reached during the academic career. We have shown that while the age at exit is used to learn the classification model, the actual age at the time of the prediction can be used for future instances. This covariate shift has no negative consequences, rather it systematically increases sensitivity to the detriment of a modest reduction in specificity. In this way we are able to correctly identify about seven out of ten students who drop out and eight out of ten students who graduate. We intend to test our proposal on cohorts of newly matriculated students including, among the testing variables, the ‘real’ age at matriculation and/or the time elapsed since matriculation, as soon as these new data become available.

Since sensitivity and specificity usually move in opposite directions, if we want to increase them both we will need more information. We must therefore recognize that the input variables used are insufficient for more accurate classifications. The inclusion of new input variables is, therefore, the first step to be taken. Among the variables immediately available at the time of matriculation that could prove useful, we can mention the grade obtained in the high school diploma, the deprivation index of the municipality of residence to take into account the socio-economic conditions of the environment in which the student lives, as well as the ISEE (Equivalent Economic Situation Indicator) to introduce individual economic conditions in classification models. The only individual academic performance measures already available at the time of matriculation concern the result of the entrance exam, which is mandatory for all students matriculation in free access Study Courses. Furthermore, one could think of including in the model the academic performance relative to the first semester only (so as to make it possible, in any case, to predict early dropout and to activate individual measures for the prevention of this eventuality). Experiments on these input variables will be the subject of future research.

From a merely algorithmic point of view, there would seem to be no room for further improvements. However, in the past much evidence has been accumulated that the feedforward networks with a large number of hidden levels, or networks with more complex topologies, but equally characterized by the presence of a very large number of compositions of non-linear functions to model the relationship between input and output, have a higher (and substantially not yet explained) generalization capability than traditional algorithms [31, 26]. The use of deep learning algorithms, in conjunction with the availability of an adequate amount of information, could therefore lead to a significant performance boost in terms of predictive accuracy and could represent a decisive step forward in building systems of early dropout prediction that can also be used from a practical point of view. These aspects will also be subject to future experimentation and research.

References

1. Aggarwal CC. *Neural Networks and Deep Learning*. Cham: Springer International Publishing; 2018. https://doi.org/10.1007/978-3-319-94463-0.
2. Agresti A. *An Introduction to Categorical Data Analysis*. Wiley-Blackwell; 2007.
3. Baker RSJD, Yacef K. The State of Educational Data Mining in 2009: A Review and Future Visions. *J Educ Data Min* 2009;1:3–17.
4. Belloc F, Maruotti A, Petrella L. University drop-out: an Italian experience. *High Educ* 2010;60:127–38. https://doi.org/10.1007/s10734-009-9290-1.
5. Belloc F, Maruotti A, Petrella L. How individual characteristics affect university students drop-out: a semiparametric mixed-effects model for an Italian case study. *J Appl Stat* 2011;38:2225–39. https://doi.org/10.1080/02664763.2010.545373.
6. Bergmeir C, Benítez JM. Neural Networks in R Using the Stuttgart Neural Network Simulator: RSNNS. *J Stat Softw* 2012;46. https://doi.org/10.18637/jss.v046.i07.
7. Burgalassi M, Biasi V, Capobianco R, Moretti G. The phenomenon of Early College Leavers. A case study on the graduate programs of the Department of Education of “Roma Tre” University. *Ital J Educ Res* 2016;17.
8. Caputo B, Sim K, Furesjo F, Smola A. Appearance-based object recognition using SVMs: which kernel should I use? Proc. NIPS Work. Stat. methods Comput. Exp. Vis. Process. Comput. Vis., Whistler: 2002.
9. Carci G. I Percorsi di Studio: Mobilità, Abbandoni e Consegumento del Titolo. In: ANVUR National Agency for the Evaluation of Universities and Research Institutes, editor. Rapporto Biennale sullo Stato del Sistema Universitario e della Ricerca 2018, ANVUR; 2018, p. 43–73.
10. Cerda P, Varoquaux G, Kégl B. Similarity encoding for learning with dirty categorical variables. Mach Learn 2018;107:1477–94. https://doi.org/10.1007/s10994-018-5724-2.
11. Dekker G, Pechenizkiy M, Vleeshouwers J. Predicting Students Drop Out: A Case Study. In: Barnes T, Desmarais MC, Romero C, Ventura S, editors. EDM, www.educationaldatamining.org; 2009, p. 41–50.
12. Delen D. A comparative analysis of machine learning techniques for student retention management. Decis Support Syst 2010; 49:498–506. https://doi.org/10.1016/j.dss.2010.06.003.
13. Di Pietro G. The determinants of university dropout in Italy: A bivariate probability model with sample selection. Appl Econ Lett 2004;11:187–91. https://doi.org/10.1080/1350485042000203832.
14. Dunson DB. Statistics in the big data era: Failures of the machine. Stat Probab Lett 2018;136:4–9. https://doi.org/10.1016/j.spl.2018.02.028.

15. Fan J, Ma C, Zhong Y. A Selective Overview of Deep Learning 2019. ArXiv:1904.05526 [stat.ML], http://arxiv.org/abs/1904.05526.
16. Fawcett T. An introduction to ROC analysis. Pattern Recognit Lett 2006;27:861–74. https://doi.org/10.1016/j.patrec.2005.10.010.
17. Georg W. Individual and institutional factors in the tendency to drop out of higher education: a multilevel analysis using data from the Konstanz Student Survey. Stud High Educ 2009;34:647–61. https://doi.org/10.1080/03075070802592730.
18. Ghahramani Z. Probabilistic machine learning and artificial intelligence. Nature 2015;521:452–9. https://doi.org/10.1038/nature14541.
19. Goodfellow I, Bengio Y, Courville A. Deep Learning. MIT Press; 2016.
20. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. 2nd ed. New York, NY: Springer New York; 2009. https://doi.org/10.1007/978-0-387-84858-7.
21. Hoffait A-S, Schyns M. Early detection of university students with potential difficulties. Decis Support Syst 2017;101:1–11. https://doi.org/10.1016/j.dss.2017.05.003.
22. Hofmann T, Schölkopf B, Smola AJ. Kernel methods in machine learning. Ann Stat 2008;36:1171–220. https://doi.org/10.1214/009053607000000677.
23. Ismail MR, Awang MK, Rahman MNA, Makhtar M. A Multi-Layer Perceptron Approach for Customer Churn Prediction. Int J Multimed Ubiquitous Eng 2015;10:213–22. https://doi.org/10.14257/ijmue.2015.10.7.22.
24. Karatzoglou A, Smola A, Hornik K, Zeileis A. kernlab - An S4 Package for Kernel Methods in R. J Stat Softw 2004;11. https://doi.org/10.18637/jss.v011.i09.
25. Karatzoglou A, Meyer D, Hornik K. Support Vector Machines in R. J Stat Softw 2006;15:1–28. https://doi.org/10.18637/jss.v015.i09.
26. Kawaguchi K, Kaelbling LP, Bengio Y. Generalization in Deep Learning 2017. ArXiv: 1710.05468v4 [stat.ML], http://arxiv.org/abs/1710.05468.
27. Khodabandehlou S, Zivari Rahman M. Comparison of supervised machine learning techniques for customer churn prediction based on analysis of customer behavior. J Syst Inf Technol 2017;19:65–93. https://doi.org/10.1108/JSIT-10-2016-0061.
28. Kumar B, Pal S. Mining Educational Data to Analyze Students Performance. Int J Adv Comput Sci Appl 2011;2. https://doi.org/10.14569/IJACSA.2011.020609.
29. Kumar M, Singh AJ, Handa D. Literature Survey on Educational Dropout Prediction. Int J Educ Manag Eng 2017;7:8–19. https://doi.org/10.5815/ijeme.2017.02.02.
30. Kuhn M. Building Predictive Models in R Using the caret Package. J Stat Softw 2008;28. https://doi.org/10.18637/jss.v028.i05.
31. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44. https://doi.org/10.1038/nature14539.
32. Manning CD, Raghavan P, Schütze H. Introduction to Information Retrieval. New

York, NY, USA: Cambridge University Press; 2008.

33. Márquez-Vera C, Cano A, Romero C, Noaman AYM, Mousa Fardoun H, Ventura S. Early dropout prediction using data mining: a case study with high school students. Expert Syst 2016;33:107–24. https://doi.org/10.1111/exsy.12135.
34. McCullagh P, Nelder J. Generalized Linear Models, Second Edition. Chapman and Hall/CRC Monographs on Statistics and Applied Probability Series, Chapman & Hall; 1989.
35. Meggiolaro S, Giraldo A, Clerici R. A multilevel competing risks model for analysis of university students' careers in Italy. Stud High Educ 2017;42:1259–74. https://doi.org/10.1080/03075079.2015.1087995
36. Miguéis VL, Freitas A, Garcia PJV, Silva A. Early segmentation of students according to their academic performance: A predictive modelling approach. Decis Support Syst 2018;115:36–51. https://10.1016/j.dss.2018.09.001.
37. Ministero dell'Università Istruzione e Ricerca. Osservatorio Studenti/Didattica – Anagrafe Nazionale degli Studenti. http://anagrafe.miur.it/index.php (accessed April 16, 2019).
38. Mitchell TM. Machine Learning. 1st ed. New York, NY, USA: McGraw-Hill, Inc.; 1997.
39. Montmarquette C, Mahseredjian S, Houle R. The determinants of university dropouts: a bivariate probability model with sample selection. Econ Educ Rev 2001;20:475–84. https://doi.org/S0272-7757(00)00029-7.
40. Ng A, Jordan MI. On generative vs. discriminative classifiers: A comparison of logistic regression and naive bayes. Proc Adv Neural Inf Process 2002;28:169–87. https://doi.org/10.1007/s11063-008-9088-7.
41. Parikh R, Mathai A, Parikh S, Chandra Sekhar G, Thomas R. Understanding and using sensitivity, specificity and predictive values. Indian J Ophthalmol 2008;56:45–50.
42. Polson NG, Sokolov V. Deep Learning: A Bayesian Perspective. Bayesian Anal 2017;12:1275–304. https://doi.org/10.1214/17-BA1082.
43. R Core Team. R: A Language and Environment for Statistical Computing. R Foundation
44. for Statistical Computing, Vienna, Austria; 2019. https://www.R-project.org/
45. Gil M, Reyes N, Juárez M, Espitia E, Mosqueda J, Soria M. Predicting Early Students with High Risk to Drop Out of University using a Neural Network-Based Approach. ICCGI 2013, Eighth Int. Multi-Conference Comput. Glob. Inf. Technol., IARIA; 2013, p. 289–94.
46. Rubin DB. Iteratively Reweighted Least Squares. Encycl. Stat. Sci., Hoboken, NJ, USA: John Wiley & Sons, Inc.; 2006.
https://doi.org/10.1002/0471667196.ess1296.pub2.

47. Şara N-B, Halland R, Igel C, Alstrup S. High-school dropout prediction using machine learning: a Danish large-scale study. In: Verleysen M, editor. Proceedings. ESANN 2015, i6doc.com; 2015, p. 319–24.
48. Scheuer O, McLaren BM. Educational Data Mining. In: Seel NM, editor. Encycl. Sci. Learn., Boston, MA: Springer US; 2012, p. 1075–9. https://doi.org/10.1007/978-1-4419-1428-6618.
49. Schmidhuber J. Deep learning in neural networks: An overview. Neural Networks 2015;61:85–117. https://doi.org/10.1016/j.neunet.2014.09.003.
50. Tekin A. Early Prediction of Students' Grade Point Averages at Graduation: A Data Mining Approach. Eurasian J Educ Res 2014;14:207–26. https://doi.org/10.14689/ejer.2014.54.12.
51. Teshnizi S, Ayatollahi S. A Comparison of Logistic Regression Model and Artificial Neural Networks in Predicting of Student's Academic Failure. Acta Inform Medica 2015;23:296–300. https://doi.org/10.5455/aim.2015.23.296-300.
52. Vapnik VN. An overview of statistical learning theory. IEEE Trans Neural Networks 1999;10:988–99. https://doi.org/10.1109/72.788640.
53. Willging PA, Johnson SD. Factors that Influence Students' Decision to Dropout of Online Courses. J Asynchronous Learn Networks 2004;8:105–18.
54. Wolpert DH, Macready WG. No free lunch theorems for optimization. IEEE Trans Evol Comput 1997;1:67–82. https://doi.org/10.1109/4235.585893.
55. Wolpert DH. The Supervised Learning No-Free-Lunch Theorems. Soft Comput. Ind., London: Springer London; 2002, p. 25–42. https://doi.org/10.1007/978-1-4471-0123-93.