Co-funded by the
Erasmus+ Programme
of the European Union

Student Profile
for Enhancing
Engineering Tutoring

ERASMUS + KA2 / KA203

Data Mining Tool for Academic Data Exploitation

Selection of most suitable Algorithms

J. L. Vicario (Coordinator), R. Vilanova, M. Bazzarelli, A. Paganoni, U. Spagnolini, A. Torrebruno, M.A. Prada, A. Morán, M. Domínguez, M.J. Varanda, P. Alves, M. Podpora and M. Barbu

March 2018

ISBN 978-989-20-8738-2

INTELLECTUAL OUTPUT #2

Student Profile for
Enhancing Tutoring Engineering

Data Mining Tool for Academic Data Exploitation

Selection of most suitable Algorithms

J. L. Vicario (Coordinator), R. Vilanova

Dept. de Telecomunicacio i Enginyeria de Sistemes
Escola d'Enginyeria, UAB
Carrer de es Sitges 08193 Bellaterra, Barcelona, Spain

M. Bazzarelli, A. Paganoni, U. Spagnolini, A. Torrebruno

Scuole di Ingegneria
Politecnico di Milano, Milano, Italy

M.A. Prada, A. Morán, M. Domínguez

Dept. de Ingeniería Eléctrica y de Sistemas y Automática
Escuela de Ingenierías Industrial e Informática
Universidad de León, León, Spain

M.J. Varanda, P. Alves

Escola Superior de Tecnologia e Gestao
Instituto Politecnico de Braganca, Braganca, Portugal

M. Podpora

Faculty of Electrical Engineering, Automatic Control and Informatics
Opole University of Technology, Opole, Poland

M. Barbu

Automatic Control and Electrical Engineering Department
"Dunarea de Jos" University of Galati, Galati, Romania

Final Version
Approved for public release; distribution is unlimited.

ERASMUS + KA2/KA203

Prepared for	SPEET Intellectual Output #2
Under	2016-1-ES01-KA203-025452
Monitored by	SEPIEE

Disclaimer: The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Contents

1 Executive Summary 2

2 Academic Data 3

2.1 Terms and Definitions 3

2.1.1 Background and Motivation 3

2.1.2 Confidentiality of the Academic Data 5

2.1.3 On Defining Different Profiles 5

2.2 Proposal for Dataset Format 6

2.3 Organization of the Document 7

3 Student performance Clustering and Classification 8

3.1 Data Base Format and Pre-Processing 8

3.1.1 Categorical vs Numerical/Performance data 8

3.1.2 Data Pre-Processing 9

3.2 Implemented algorithms 11

3.2.1 Clustering block 11

3.2.2 Classification block 15

3.3 Case Study Applications 17

3.4 Code Availability 21

4 Student drop-out prediction 24

4.1 Data Base Format and Pre-Processing 24

4.1.1 Data Cleaning Process 25

4.1.2 Categorical vs Numerical/Performance Data 26

4.1.3 Preliminary Analysis 28

4.2 Materials and Methods 29

4.2.1 Materials 29

4.2.2 Methods 31

4.3 Implemented Algorithms 34

4.3.1 Simple Logit Models 34

4.3.2 Logit Mixed-Effects Model 39

4.3.3 Final Model: Student Drop-out Predictor 44

4.4 Case Study Applications 45

4.4.1 Dealing with Random Effects 45

4.4.2 Prediction 46

4.4.3 Model Validation 49

4.4.4 Model Accuracy 51

4.5 Code Availability 52

Bibliography 53

1 Executive Summary

SPEET project is aimed at exploiting the potential synergy among the huge amount of academic data actually existing at universities and the maturity of data science in order to provide tools to extract information from students' data. A rich picture can be extracted from this data if conveniently processed. The purpose of this project is to apply data mining algorithms to process this data in order to extract information about and to identify student profiles.

In this document, the results obtained at SPEET project under the development of the data mining tools are presented. More specifically, two mechanisms have been developed: a clustering/classification scheme of students in terms of academic performance and a drop-out prediction system.

The document starts by addressing the motivation of the development of data mining tools along with the considerations taken into account for academic data gathering. These considerations include the proposed unified dataset format and some details about confidentiality issues. Next, the students' clustering and classification schemes are presented in detail. More specifically, a description of the considered machine learning algorithms can be found. Besides, a discussion of obtained results when considering data belonging to the different SPEET project's partners is addressed. Results show how groups of clusters can be automatically identified and how new students can be classified into existing groups with a high accuracy. Finally, the implemented drop-out prediction system is considered by presenting several algorithms alternatives. In this case, the evaluation of the drop-out mechanism is focused on one institution, showing a prediction accuracy around 91 %.

Algorithms presented at this document are available at repositories or inline code format, as accordingly indicated.

2 Academic Data

The international ERASMUS+ project SPEET (Student Profile for Enhancing Tutoring Engineering) aims at opening a new perspective to university tutoring systems. Before looking for its nature, it's recommended to have a look on the current use of data in education and on the concept of academic analytics basically defined as the process of evaluating and analysing data received from university systems for reporting and decision making reasons. As a matter of fact, accrediting agencies, governments, parents and students are all calling for the adoption of new modern and efficient ways of improving and monitoring student success.

2.1 Terms and Definitions

Data has always been a significant asset for institutions and has been used to inform their day-to-day operational decisions as well as long-term business and strategic decisions.

From a more purely educational point of view, the available academic data can be collected, linked together and analyzed to provide insights into student behaviours and identify patterns to potentially predict future outcomes. In this section, usually available data will be described as well as its potential use for the benefit of students. The use of academic data for supporting tutoring action is what we will put the focus on.

2.1.1 Background and Motivation

For the last 20 years, statistical analysis in education is growing as a profitable industry with prime objective of maximizing profit by delivering high quality education that produces well-educated, skilled, mannered students according to needs and requirements of the dynamically growing market. The use of statistical analysis in education has grown in recent years for four primary reasons: a substantial increase in data quantity, improved data formats, advances in computing and increased development of tools available for analytics.

In commercial fields, business and organizations are deploying sophisticated analytic techniques to evaluate rich data sources, identify patterns within the data and exploit these patterns in decision making. Recently researchers and developers from the educational community started exploring the potential adoption of analogous techniques for gaining insight into online learners activities. The academic assessment is defined as the systematic process of gathering and analyzing information about student learning to inform curricular decision-making and improve academic programs. Programs may collect data from students (e.g., survey, focus group), examine course documents (e.g., coursework, portfolios, capstone projects), or analyze student academic data (e.g., scores, grades, credentials) for academic assessment. Even the list of goals and objectives that can be pursued with the application of analytics to academic big data can be very long: it is possible to categorize the goals in terms of the students benefits as follows:

- Improve Student Results.

  The overall goal of big data within the educational system should be to improve student results. During his student life each student generates a unique data trail. This data trail can be analysed in real-time to deliver an optimal learning environment for the student himself and to ”gain” a better understanding in his individual behaviour. In addition, with the help of appropriate algorithms, it will be possible to determine the strengths and weaknesses of each individual. This will create stronger groups that will allow students to have a steeper learning curve and deliver better group results.

- Create Mass-Customized Programs.

  All the data will help to create a customized program for each single student. It will give students the opportunity to develop their own personalized program. Providing mass customization in education is a challenge, but thanks to algorithms it becomes possible to track and assess each individual student. We already see this happening in the Massive Open Online Courses (MOOCs) that are developed around the world now.

- Improve the Learning Experience in Real-time.

  Each student learns differently and, of course, the way a student learns affects the final grade. Some students learn very efficiently while others may be extremely inefficient. If available, this information could be used to provide a customized program or a real-time feedback to become more efficient in learning and, thus, improve the results.

- Reduce Dropouts, Increase Results.

  All the previous reasonings will improve the student results and, perhaps, also reduce dropout rates at the universities. Dropouts are expensive for educational institutes as well as for society. Using predictive analytics on all the data that is collected can give educational institute insights in future student outcomes. These predictions can be used to change a particular program if bad results are predicted. Universities and colleges will become more efficient in developing a program that will increase results thereby minimizing trial-and-error.

2.1.2 Confidentiality of the Academic Data

As these data are considered sensible (it contains student track records, personal information, etc), measures to preserve the individual anonymity have been ensured. When talking about academic data, it is questionable whether the personal/nonpersonal data distinction remains viable and whether anonymisation and aggregation remain effective in protecting users against tracking and profiling. Access to these data is limited to departmental faculty, campus administrators and accrediting agencies for the purpose of program review. Results of academic assessment are used only for program improvement. Student data is a university resource and must be used for university purposes only. As the custodian of academic data, the Office of the University Registrar, is responsible for establishing or enforcing policies and procedures to protect data and ensure its appropriate use, identifiers, such as names and ID numbers, have been removed/anonymised. The goal is to avoid the use of background knowledge and cross-correlation with other databases to re-identify student data records.

2.1.3 On Defining Different Profiles

Although the SPEET project goal is very clear (i.e. determine and categorize different profiles for engineering students across Europe), the approach to achieve student profiles in such a situation raises several questions and problems arising from the difficulty of the challenge assumed by the project partners, namely

- the official data reported by universities are quantitative/numerical. The social context of the student is not investigated because of the fact that it is related with the education level of the people he lives with, health habits and financial support.
- the phenomenon of dropout from university studies has multiple causes which can be grouped at least into two major categories of factors: internal factors related to the student's personality and his level of bio-psycho-social development and external factors related to the socioeconomic, cultural and educational environment in which the student lives.

However, the official data reported by universities about students are enough to 1) identify different patterns of students in terms of their performance and 2) detect

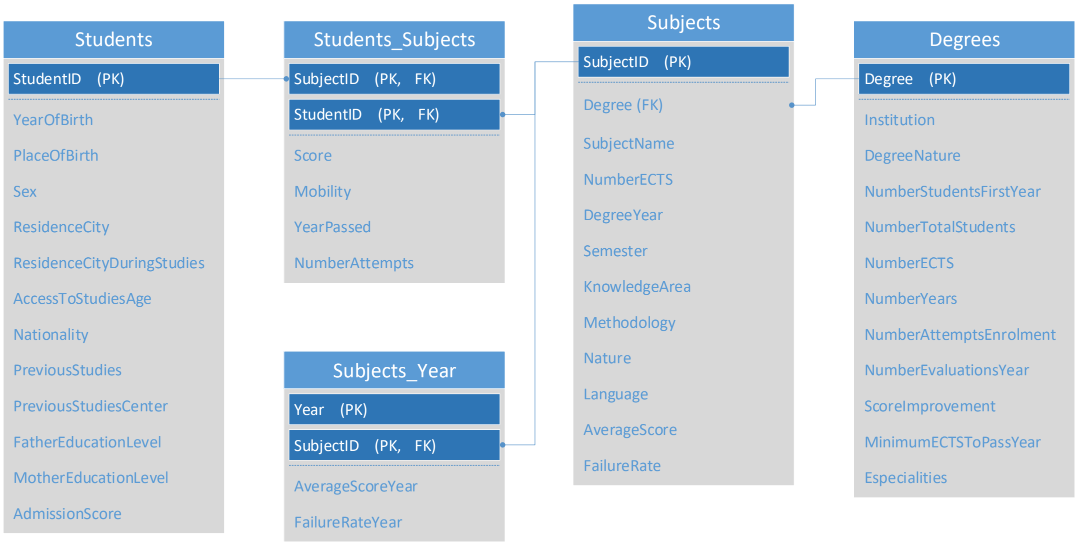


Figure 1. Relational model of the proposed structure of the dataset

students with educational *risk of dropout*, an information which, once obtained, then calls on the attention of the teachers and the management of the university to initiate some tutorial actions, counseling and failure avoidance. Tutoring and counseling will later complete the student profile by obtaining qualitative data about the student with dropout risk. Namely, for example, information generated by tools such as questionnaire, interview, checklist, structured essay, etc. The data collected will allow for a personalization of the profile and identification of other causes of socio-emotional and attitude-behavioral nature not found in official data statistically reported by universities.

2.2 Proposal for Dataset Format

One of the characteristics of the SPEET project is its transnational nature, since the fact of obtaining (or not) the same student classification and profiles will help identify common characteristics on engineering students coming from different EU institutions. The differences on a country/institution basis will be exposed and leads to deeper analysis. Due to its transnational nature, it is necessary to choose appropriate variables and representation to cover the differences in course organization at a country level. Additionally, the dataset must include students' personal information while complying with privacy regulations of the European Union. As a result, the proposed dataset uses variables obtained from the administrative records of the students, such as demographic data, courses taken and academic performance.

Figure 1 shows the initial, minimum core dataset, proposed to perform the analysis. It is also possible to enrich the dataset with other potentially useful additional data sources, e.g., the regional/metropolitan socio-economic indicators provided by organizations such as the Organisation for Economic Cooperation and Development. As a matter of fact the retrieved collections of data includes collateral information regarding the students' origin (year and place of birth, geographical info, previous studies, age, etc), degree information (degree nature, total number of students, number years, etc) as well as student performance on different subjects of the degree (subject score, subject year, subject language, subject nature, etc).

2.3 Organization of the Document

This document reflects the outputs of the SPEET project under the form of basic data mining tools. The next chapters present in detail:

- **Classification and Clustering tool**: this is a stationary-based tool consisting in the grouping of students at clusters based on their performance during their studies. This is presented in detail in Chapter 3.
- **Drop-out Prediction tool**: a dynamic tool based on the drop-out prediction of students based on their performance at the first semester of studies. Details are provided in Chapter 4).

These results are intended for qualified users with knowledge on programming and statistics. Therefore we put at their disposition the building blocks for performing direct data analysis or even generate their own IT tools.

3 Student performance Clustering and Classification

This Chapter is devoted to present the Clustering and Classification tooI. This tool has been implemented in Python and an overview of the architecture is presented in Fig. 2. By departing from the datasets presented in Chapter 2, the Pre-Processing blocks are in charge of adapting data to the Clustering and Classification blocks. The Clustering block, on the other hand, is aimed at generating three clusters of students based on their performance results. Besides, categorical information is analyzed to obtain profiles of students belonging to different clusters. Finally, the Classification block is in charge of classifying new students to the clusters generated at the Clustering block. As it will be shown later, this Classification procedure is also useful to obtain insights about the structures of plan studies at the different degrees.

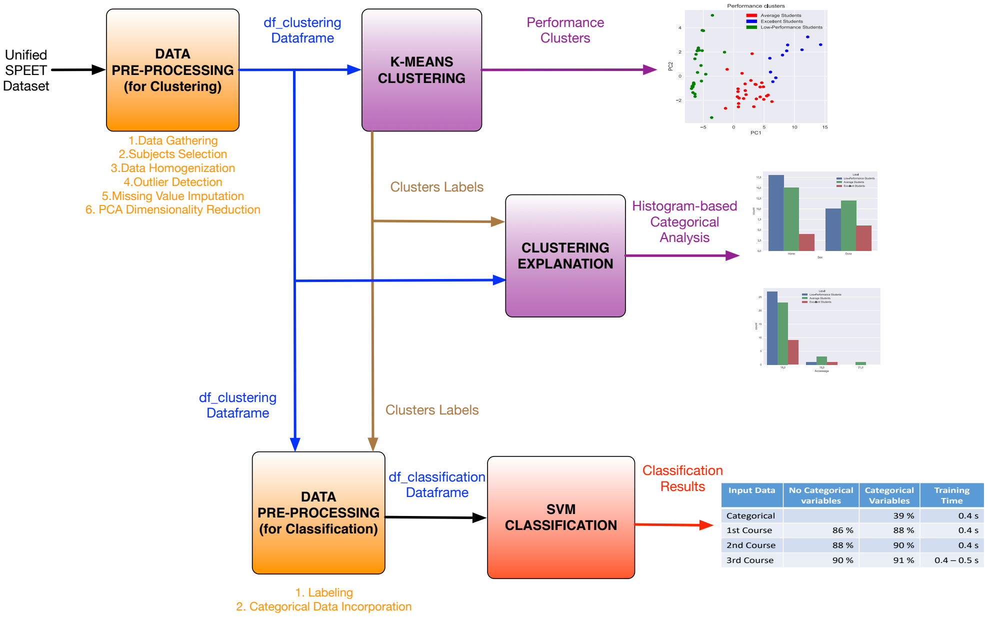
3.1 Data Base Format and Pre-Processing

As presented in Chapter 2, a unified dataset format has been considered for the project. Further details about this dataset format can also be found at the Intelectual Output # 1 document [BVV$^+$17]. Concerning the Clustering and Classification tool, some pre-processing of the data is needed to accommodate students' information to developed algorithms. Next, we present the specific aspects taken into consideration.

3.1.1 Categorical vs Numerical/Performance data

From the dataset presented in Fig. 1., the algorithms developed in this Tool focuses on the use of two kinds of data:

- **Performance data**: this data refers to the scores obtained by students at the different subjects. The nature of this data is numerical.
- **Categorical data**: this data refers to collateral information related to students. This includes features such as student demographic data (access age, gender, nationality), educational background (previous studies) or access conditions (access score).



Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		39 %	0.4 s
1st Course	86 %	88 %	0.4 s
2nd Course	88 %	90 %	0.4 s
3rd Course	90 %	91 %	0.4 – 0.5 s

Figure 2. Architecture of the Clustering and Classification tool.

3.1.2 Data Pre-Processing

The idea behind this procedure is to organize students in different groups (clusters) based on their performance results. To do so, a classical *k*-means clustering approach will be adopted, based on gathering in a cluster those elements with the highest similarity. The goal is to obtain three clusters.

As commented in the previous section, a unified dataset has been defined but, as observed during the project, data coming from different academic management databases could present differences. In some cases, we also found that datasets cannot be complete. Then, this data should be homogenized and processed to allow for the exploitation of the data mining algorithms developed in this tool.

The next steps are performed in order to generate the data frame used by the **Clustering Block** referred here as $df\_clustering$. This tool has been implemented in *Python* and dataframes creation and manipulation have been performed by means of *pandas* library:

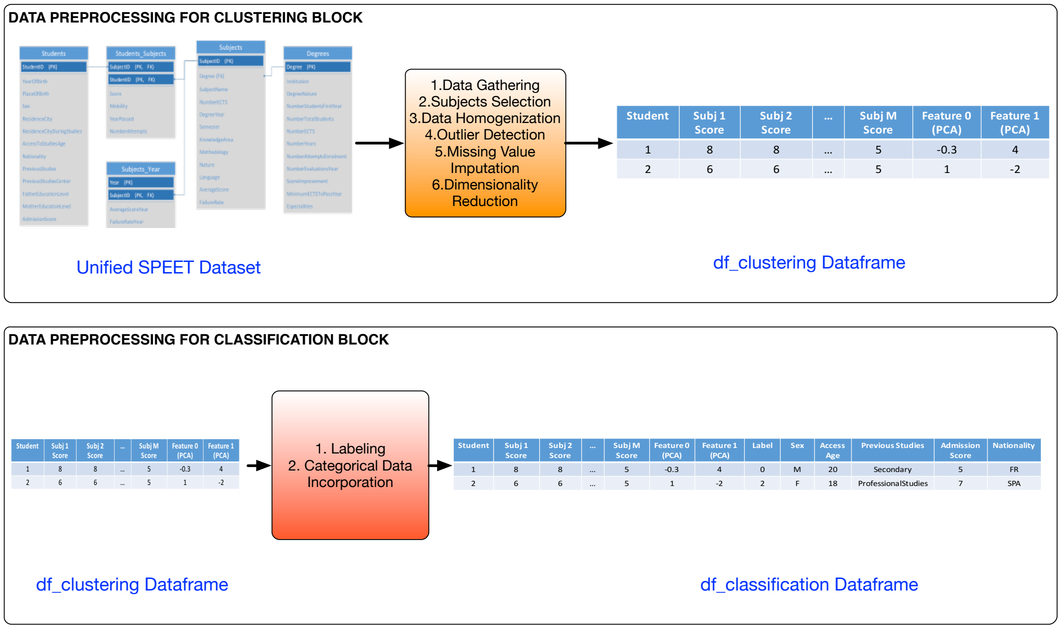
- **Data Gathering**: from the database structure received from the institution (see Fig. 1), the first step is to organize the data and generate a data mining-friendly dataframe format. Each entry of this dataframe belongs to a specific student and the scores obtained at the different subjects are considered as attributes.

- **Subjects Selection**: this block selects the set of subjects considered to generate the performance clusters. More specifically, only mandatory subjects are considered and the number of subjects differs depending on the mandatory subjects allocation on the study program. In this block, it is also verified that selected subjects belong to the set of mandatory subjects of the study programs, since some erroneous data entries were found.
- **Data Homogenization**: since different score ranges have been observed between some countries, subjects scores are normalized to 0-10 numerical evaluation.
- **Outlier Detection**: score data was previously analyzed and no outliers were detected.
- **Missing Value Imputation**: this block assigns reference score values when missing values are detected. These occurrences are due to procedures related to the recognition of subjects from previous studies. For this reason, the value of ”PASS” (numerical score equal to 5) is adopted as reference score.
- **Dimensionality reduction**: to reduce data complexity and provide a quicker execution of the clustering algorithm, this block is in charge of applying a Principal Component Analysis (PCA) algorithm [AH10]. This algorithm translates the original data from the set of mandatory subjects into a two-dimensional representation. In other words, each student entry at dataframe will have two additional attributes referred as *feature0* and *feature1*, which are the PCA components resulting from applying PCA to its subject scores. These components will be the data used by the clustering mechanism to generate the set of clusters.

Concerning the pre-processing required to perform the classification mechanism, the block departs from the dataframe $df\_clustering$ and the following steps are carried out to generate the dataframe used by the **Classification Block**, referred here as $df\_classification$:

- **Labelling**: once the clusters are obtained, a new attribute is included at the dataframe: the cluster label.
- **Categorical Data Incorporation**: a set of categorical variables are included at the dataframe as additional attributes. Again, depending on the institution, all the categorical variables may not be available.

As a summary, in Fig. 3, we present the different pre-processing steps carried out to generate $df\_clustering$ and $df\_classification$ dataframes.



Student	Subj 1 Score	Subj 2 Score	...	Subj M Score	Feature 0 (PCA)	Feature 1 (PCA)
1	8	8	...	5	-0.3	4
2	6	6	...	5	1	-2

Student	Subj 1 Score	Subj 2 Score	...	Subj M Score	Feature 0 (PCA)	Feature 1 (PCA)	Label	Sex	Access Age	Previous Studies	Admission Score	Nationality
1	8	8	...	5	-0.3	4	0	M	20	Secondary	5	FR
2	6	6	...	5	1	-2	2	F	18	ProfessionalStudies	7	SPA

Figure 3. Preprocessing steps to obtain dataframes used by the Clustering Block ($df\_clustering$ dataframe) and the Classification Block ($df\_classification$ dataframe).

3.2 Implemented algorithms

In this section, we present specific details about the Machine Learning Algorithms considered for Clustering and Classification and their implementation.

3.2.1 Clustering block

This block is in charge of grouping the different students based on their performance behavior. It is also in charge of providing explanations about the resulting clusters (i.e., identifying students' profiles belonging to the different clusters). Further details about these two functionalities are provided below:

- **K-means based Clustering**: As shown in Fig. 2, we adopt the *k-means* algorithm [Mac67] as Clustering algorithm. It is worth recalling that inputs to this block are based on the PCA components of subject scores for each student (see Fig. 3) to focus the clustering on a 2 dimensional problem. It is worth noting that we performed tests with clustering directly applied to the full dimensional problem and similar results were obtained. The clear advantage of working with PCA-based compressed data is that clustering computation time is significantly reduced, which is quite appropriate for the web-service deployment planned for *Intelectual Output # 5 - Front-end for End User Application*.

  Concerning the *k-means* algorithm, it was selected as it provides a good trade-off in terms of clustering performance vs. computational complexity. More

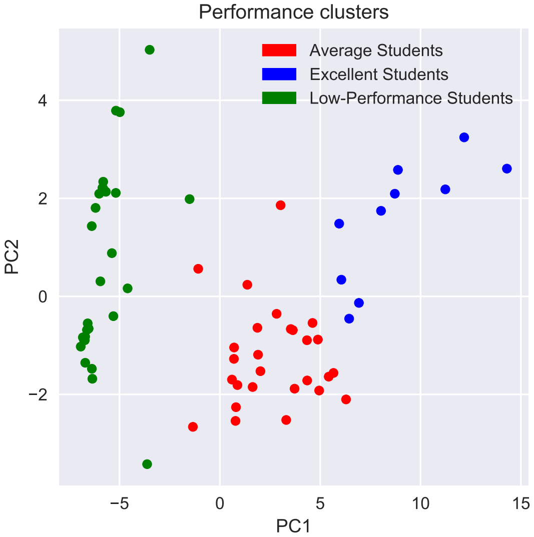


Figure 4. Clustering resulting from the k-means procedure (x- and y-axis are associated to PCA components).

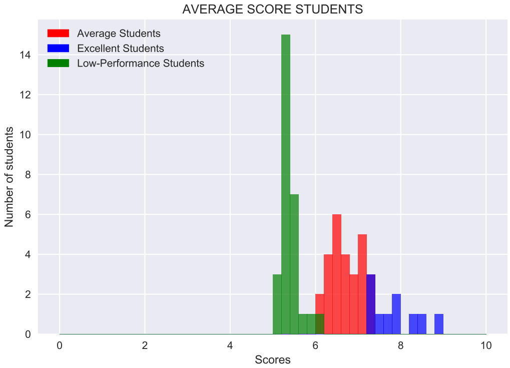


Figure 5. Histogram showing the average of the scores of all the subjects for each student belonging to the different clusters.

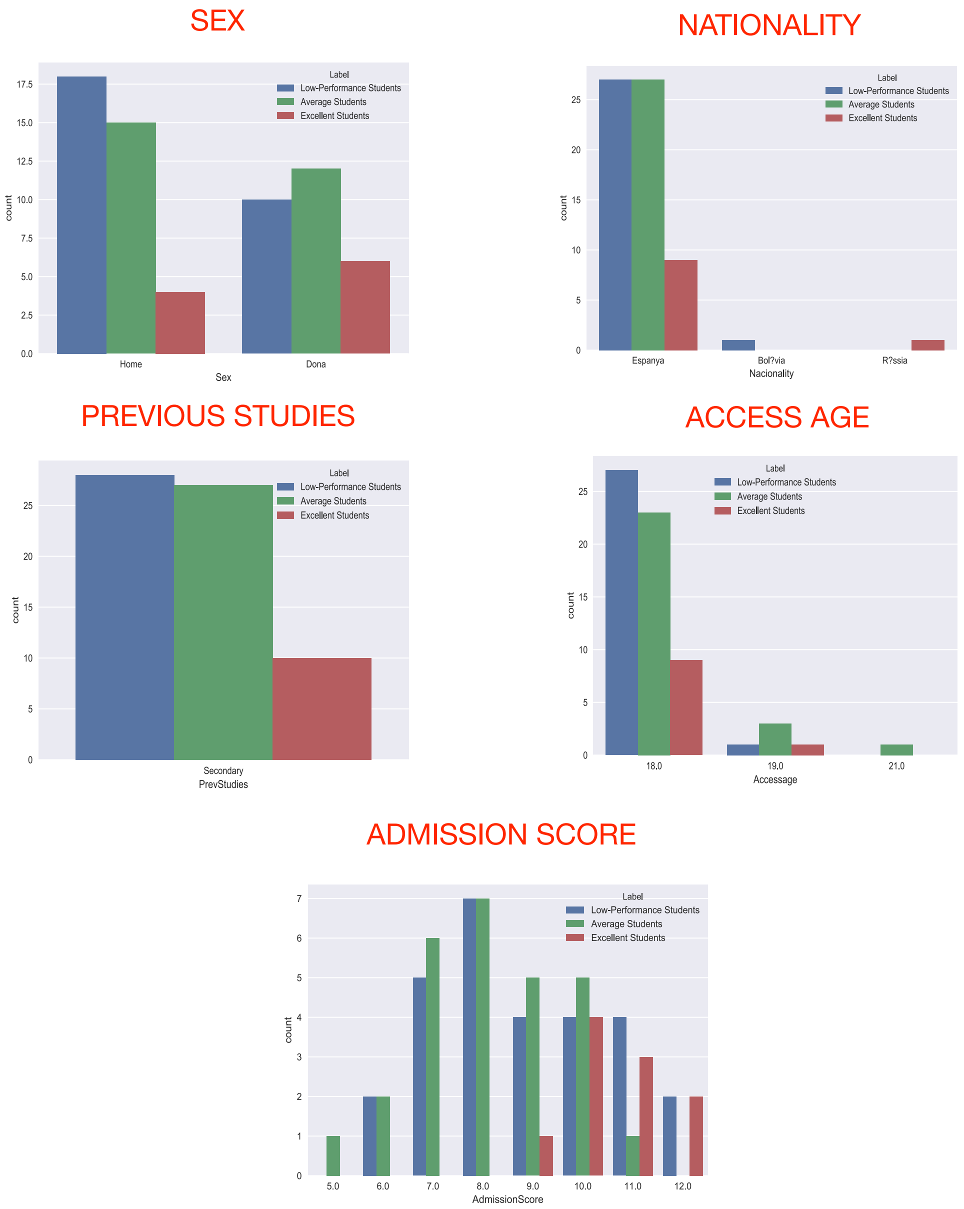


Figure 6. Example of Clustering Explanation based on histograms of Categorical Variables.

specifically, this project considers the k-means implementation of library scikit-learn version 0.18.2 in Python 2.7.13. As for the number of clusters, an Elbow Analysis was performed showing that three-four clusters significantly reduce the WSS (Within groups Sum of Squares). However, we considered three clusters for all the cases as the goal is to obtain a manageable number of clusters to obtain students' patterns. This number of groups also provides flexibility to adapt tutoring actions to segments of students, defined here as: ”Excellent Students”, ”Average Students” and ”Low-Performance Students”.

The final clustering step is to perform the Labeling process, i.e., to associate the three generated clusters with the three labels: Excellent, Average and Low-Performance. This is carried out by taking into account the average of the scores obtained by all students at each cluster. The highest average is associated to the Excellent label and the lowest one to the Low-Performance.

In Fig. 4, we show an example of the three clusters generated when considering Chemical Engineering degree at Universitat Autonoma de Barcelona (UAB). As shown in this case, clear groups are created. This is also reflected when we compute the average of the scores of all the subjects for each student and we present this by means of a histogram (see Fig. 5). Here one can clearly see how students belonging to the three different groups have different performance profiles.

- **Histogram based Clustering Explanation**: The second functionality of the Clustering tool is based on the generation of histograms to analyze the patterns of students at different clusters. More specifically, these patterns are analyzed by considering a set of categorical variables: Sex, Previous Studies, Admission Score, Access Age and Nationality. For each Categorical Variable, three histograms are generated to show the students' distributions associated to the different clusters. This methodology is inspired by the customer segmentation procedures applied in Marketing applications [AS01].

  In Fig. 6, we also also consider Chemical Engineering degree students from UAB to show the histograms obtained in this case. In this case, very homogeneous student patterns are found, but some conclusions can be extracted:

  – Sex: Excellent students tend to be Women (Dona in Catalan).

  – Access Age: Excellent students tend to be younger.

  – Admission Score: Excellent students tend to have higher admission scores.

3.2.2 Classification block

In this block the objective is to develop a classification mechanism able to classify new students in terms of the Performance Clusters obtained at the previous block. In this project, two methodologies have been evaluated:

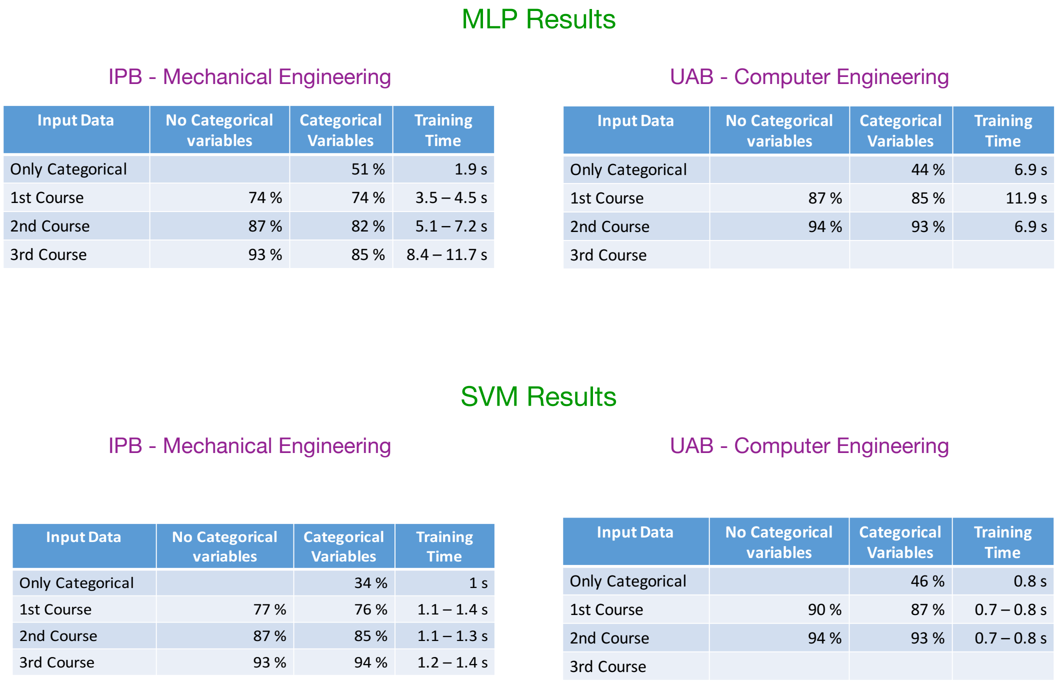
- **Multi-layer Perceptron (MLP)** [Bis95]: is a class of neural network that falls into the family of supervised learning algorithms that can learn a non-linear function approximator for either classification or regression. MLP utilizes a supervised learning technique called back-propagation for training and the neural network has three types of layers: Input layer, Hidden layers and Output layer. In our case, input layer neurons are directly the different attributes of $df\_clustering$ dataframe (subject scores and categorical variables - see Chapter 2), output layer give us the probability that students belong to the different performance clusters. Concerning hidden layers, the number of neurons and layers are configured to assess performance vs. computational trade-offs (more neurons-layers could provide better results at the expense of more complexity and training configuration problems). This project considers the Multi-layer Perceptron classifier implementation of library scikit-learn version 0.18.2 in Python 2.7.13. and different network configurations were tested with the following configuration:
  - **tol**: 1e-4. Tolerance for the stopping criterion (weights are iteratively optimized).
  - **learning rate init**: 0.1. The initial learning rate adopted by the optimizer.
  - **learning rate**: constant. A constant learning rate equal to learning rate init is considered.
  - **momentum**: 0.9. Momentum for gradient descent update.
  - **max iter**: 200. Maximum number of iterations. The solver iterates until convergence (determined by tol) or this number of iterations.
  - **activation**: relu. Activation function adopted at the hidden layers, in this case the rectified linear unit function, i.e., $f(x) = \max(0, x)$.
  - **batch size**: auto. Size of mini batches for stochastic optimizers. In this case, batch size=min(200, nsamples).
  - **solver**: sgd. This specifies the solver for weight optimization, stochastic gradient descent in this case.
  - **hidden layer sizes**: (7), (14,14), (14, 14,14). The number of neurons at each hidden layer, the i-th element represents the number in the i-th hidden layer. Three configurations were compared: with one hidden layer, two hidden layers and three hidden layers.

  - **Training and Test Ratios**: 80% and 20% for training and test, respectively.

- **Support Vector Machine (SVM)** [CST00]: is also a supervised algorithm which can be used for both classification and regression challenges. However, it is mostly used in classification problems. In this algorithm, each data item is plotted as a point in *n*-dimensional space (where *n* is number of features you have) with the value of each feature being the value of a particular coordinate. Then, a classification is identified by finding the hyper-plane that differentiate classes in the best way. This project considers the SVM implementation of library scikit-learn version 0.18.2 in Python 2.7.13. (C-Support Vector Classification) and different configurations were tested:

  - **tol**: 1e-3. Tolerance for the stopping criterion (weights are iteratively optimized).
  - **max iter**: -1 (no limit). Maximum number of iterations. In this case, the solver iterates until convergence (determined by tol).
  - **class weight**: balanced. Weights are adjusted by taking into account class frequencies (appropriate when the number of elements as each class is not homogeneous).
  - **degree**: 3. Degree of the polynomial kernel function (only when kernel is set to poly).
  - **probability**: True. It provides probability estimates to belong to the different classes (to allow for obtaining soft estimates to belong to a performance cluster).
  - **kernel**: linear, poly and rbf. This specifies the kernel type of the algorithm. In this project, we compare results obtained with these three kernel configurations.
  - **C**: 0.5, 1, 2 and 5. This is the penalty parameter C of the error term. Again, different configurations are considered to compare obtained results.
  - **Training and Test Ratios**: 80% and 20% for training and test, respectively.

MLP and SVM configurations presented above were compared by considering two degrees: Mechanical Engineering at Instituto Politecnico de Bragansa (IPB) and Computer Engineering at UAB. Obtained results can be observed at Fig. 7, where training time and classification performance were selected as performance metrics. Notice that different classifiers were derived based on the nature of attributes considered as inputs (only subject scores, only categorical variables and



MLP Results

IPB - Mechanical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Only Categorical		51 %	1.9 s
1st Course	74 %	74 %	3.5 – 4.5 s
2nd Course	87 %	82 %	5.1 – 7.2 s
3rd Course	93 %	85 %	8.4 – 11.7 s

UAB - Computer Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Only Categorical		44 %	6.9 s
1st Course	87 %	85 %	11.9 s
2nd Course	94 %	93 %	6.9 s
3rd Course			

SVM Results

IPB - Mechanical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Only Categorical		34 %	1 s
1st Course	77 %	76 %	1.1 – 1.4 s
2nd Course	87 %	85 %	1.1 – 1.3 s
3rd Course	93 %	94 %	1.2 – 1.4 s

UAB - Computer Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Only Categorical		46 %	0.8 s
1st Course	90 %	87 %	0.7 – 0.8 s
2nd Course	94 %	93 %	0.7 – 0.8 s
3rd Course			

Figure 7. MLP vs. SVM performance results. Only the results of the best configurations are shown (MLP with only one hidden layer and SVM with linear kernel and C=1).

categorical + subject scores) and number of courses (notice that computer engineering has only two courses with mandatory subjects).

Since MLP and SVM provide similar results but SVM is 10x faster, SVM was selected as the reference classifier for the rest of the project activities. More specifically, the selected SVM configuration adopts C equal to 1 and a linear kernel.

3.3 Case Study Applications

In this Section, we present a summary of results we obtained with both Clustering and Classification algorithms:

- **Clustering Evaluation**: Concerning the Clustering part, we first show two representative cases: Civil Engineering at IPB and Chemical Engineering at UAB (see Fig. 8). These two cases are representative as provide a bad and a good example in terms of Clustering behavior, respectively. This is reflected at the Average Score of Students histograms at Fig. 9. Clearly, the Civil Engineering case does not present as clear performance groups as the Chemical Engineering does. Indeed, one can readily observe that the Civil Engineering case is a

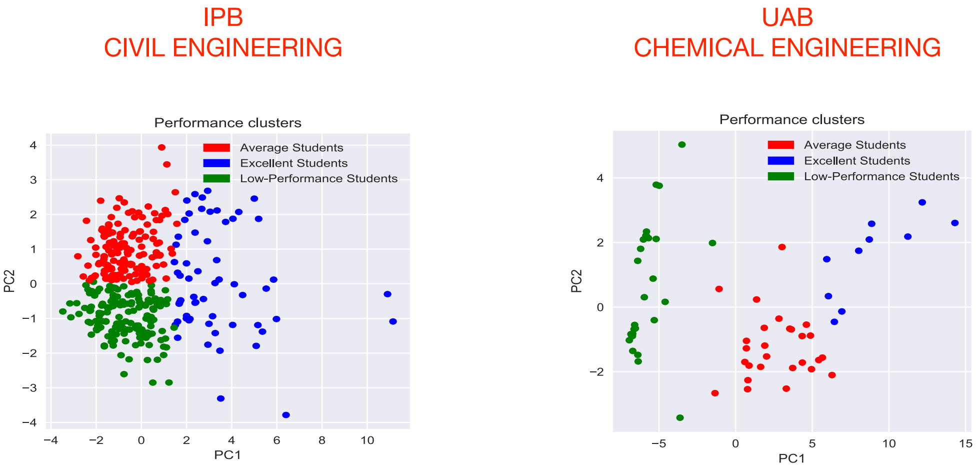


Figure 8. Performance Clusters for two cases in terms of Clustering Behavior: Left. Bad Clustering behavior, Right. Good Clustering behavior.

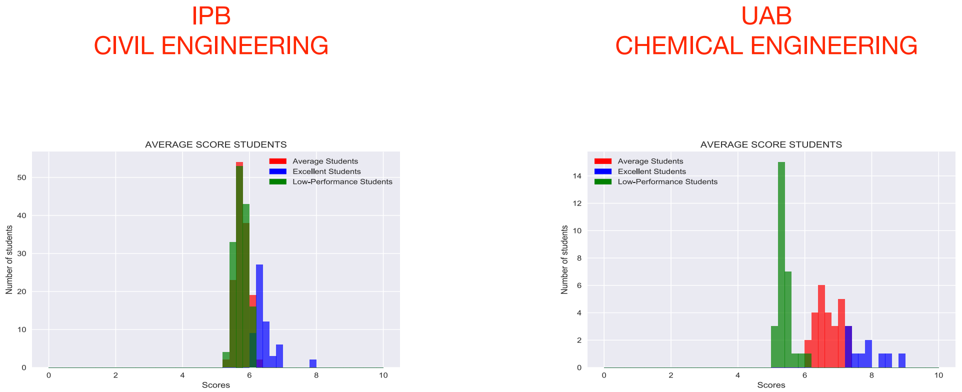


Figure 9. Histogram showing the average of the scores of all the subjects for two cases in terms of Clustering Behavior: Left. Bad Clustering behavior, Right. Good Clustering behavior.

scenario where students are better grouped by taking into account two Clusters (i.e., ”Low-Performance Students” and ”Average Students” should belong to the same cluster). This is a common pattern observed with the degrees considered at this project: when the Clustering behavior is bad, it means that two clusters is a better option. However, to improve the robustness of the proposed tool, we focus on the three cluster configuration as baseline.

As a Clustering results summary, we present the Silhouette coefficient for different degrees and Universities at Fig. 10. The reason values are so low is that performance of Students cannot be separated with the same good behavior as other Clustering problems allow[1] (e.g., rarely good students are very good in all the subjects). But our results showed that good student patterns can be obtained for the purpose of the tutoring actions: segment the actions to groups of students with similar needs. In particular, our analysis showed that Clustering levels when considering SPEET partners degrees are the following: Good Quality when Silhouette is higher than 0.2, Medium for 0.1-0.2 range and Bad for values lower than 0.1. We consider as good quality as the ability to see clear clusters (as in the example presented in Fig. 4). We also present an evaluation in terms of the ability to separate groups of students in terms of obtained scores. There we adopt color scales to show the quality, where a GREEN and RED examples could be the IPB Civil Engineering and UAB Chemical Engineering behaviors, respectively, presented above at Fig. 9.

- **Classification Evaluation**: in Figs. 11, 12, 13, 14, 15 and 16, we present classification results for different degrees and universities. As previously commented, the SVM-based (linear kernel with C=1) classifier is adopted. To understand these results, several points should be taken into account:

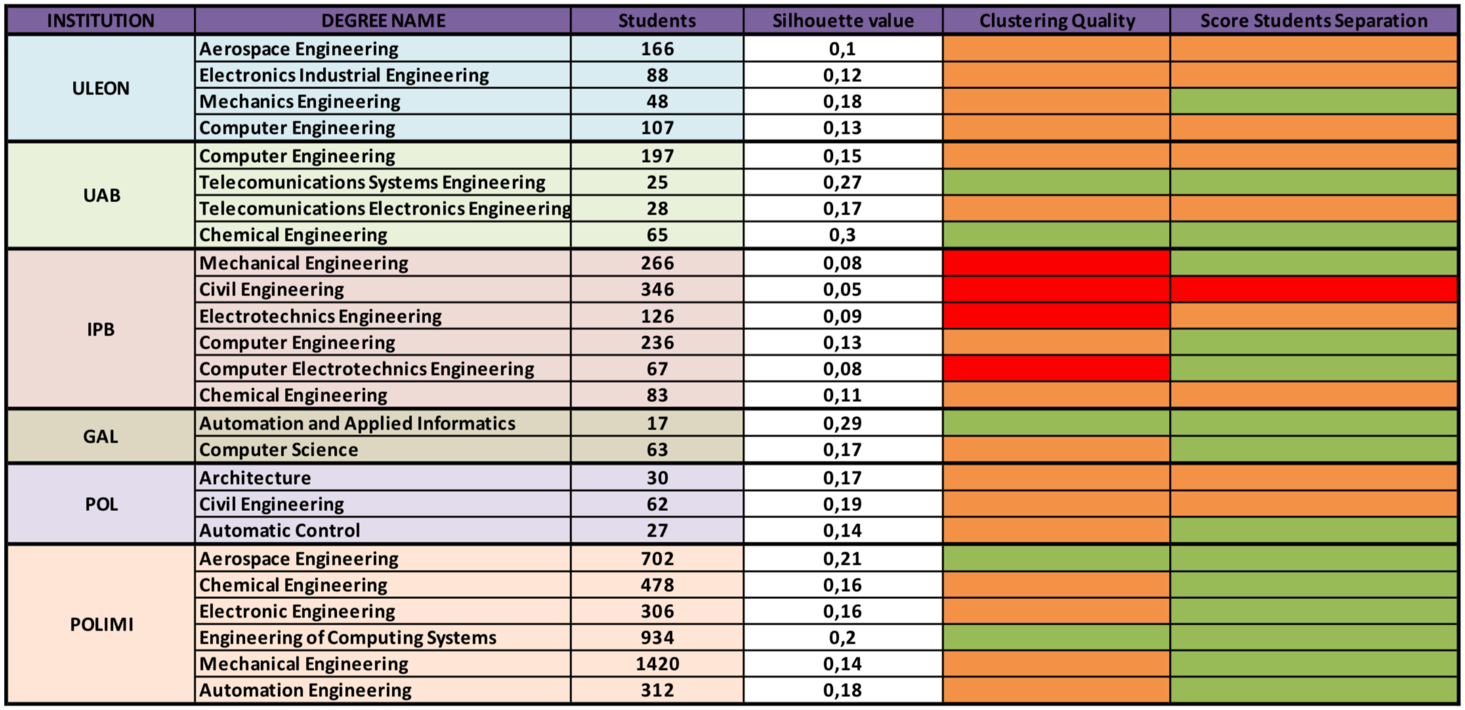
  – Number of courses: only mandatory subjects are considered. For this reason two or three courses are considered depending on the specific degree.

  – Input data: Classification is applied by considering different setups: only categorical value as input data, only the first course performance, the first + the second course performance or the first + the second + the third course performance.

  – No categorical or Categorical variables: Classification can be applied by considering only the subjects results or by considering the subject results along with the categorical variables of students (Categorical variables case).

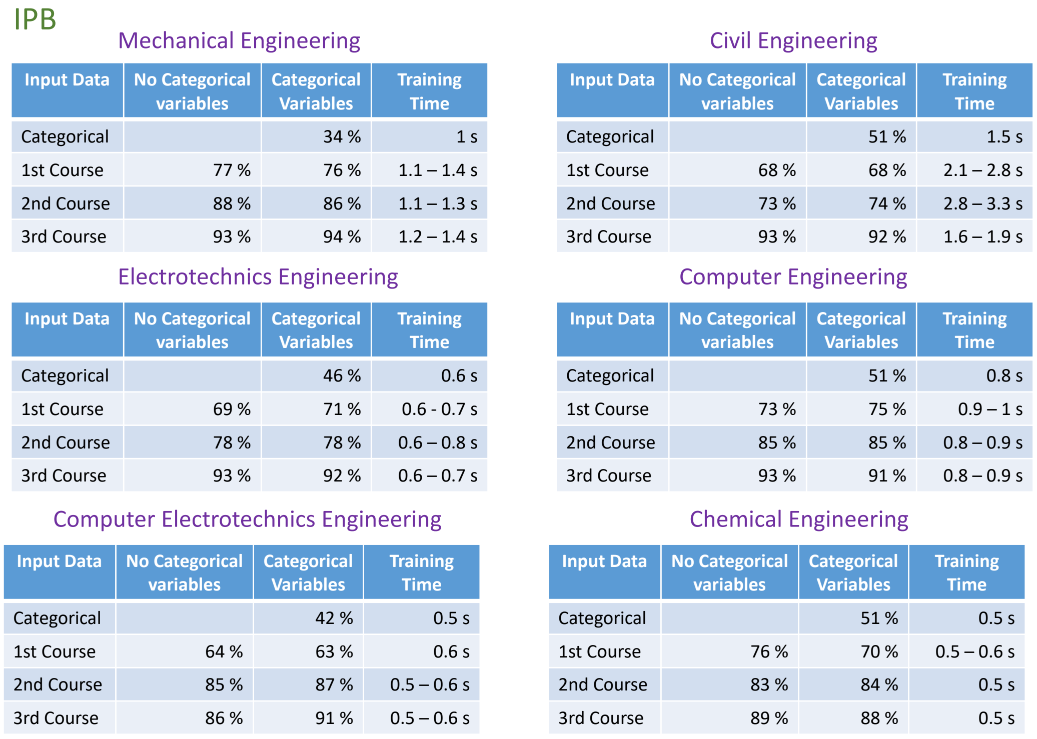
  Once the results are analyzed, one can verify that classification performance presents satisfactory results depending on the kind of degree/institution considered. But it is shown that the adoption of a SVM-based classifier provides a good trade-off in terms of accuracy vs. training time. On the other hand, one can also observe that categorical variables are not enough to classify students. Here it is worth pointing out that categorical variables are useful to understand profiles belonging to different clusters (by means of the Histogram-based Clustering explanation provided by the Clustering tool) but, however, this is somewhat different to try to accurately classify students beforehand. This is because performance at their studies are significantly affected by a complex set of factors not fully determined by the categorical variables.

[1] It is worth recalling that similar Silhouette results are obtained when k-means is directly applied to the complete set of subjects (i.e., without applying PCA dimensionality reduction) but at the expense of a higher computational complexity.



INSTITUTION	DEGREE NAME	Students	Silhouette value	Clustering Quality	Score Students Separation
ULEON	Aerospace Engineering	166	0,1	🟧	🟧
	Electronics Industrial Engineering	88	0,12	🟧	🟧
	Mechanics Engineering	48	0,18	🟧	🟩
	Computer Engineering	107	0,13	🟧	🟧
UAB	Computer Engineering	197	0,15	🟧	🟧
	Telecomunications Systems Engineering	25	0,27	🟩	🟩
	Telecomunications Electronics Engineering	28	0,17	🟧	🟧
	Chemical Engineering	65	0,3	🟩	🟩
IPB	Mechanical Engineering	266	0,08	🟥	🟩
	Civil Engineering	346	0,05	🟥	🟥
	Electrotechnics Engineering	126	0,09	🟥	🟧
	Computer Engineering	236	0,13	🟧	🟩
	Computer Electrotechnics Engineering	67	0,08	🟥	🟩
	Chemical Engineering	83	0,11	🟧	🟧
GAL	Automation and Applied Informatics	17	0,29	🟩	🟩
	Computer Science	63	0,17	🟧	🟩
POL	Architecture	30	0,17	🟧	🟧
	Civil Engineering	62	0,19	🟧	🟧
	Automatic Control	27	0,14	🟧	🟩
POLIMI	Aerospace Engineering	702	0,21	🟩	🟩
	Chemical Engineering	478	0,16	🟧	🟩
	Electronic Engineering	306	0,16	🟧	🟩
	Engineering of Computing Systems	934	0,2	🟩	🟩
	Mechanical Engineering	1420	0,14	🟧	🟩
	Automation Engineering	312	0,18	🟧	🟩

Figure 10. Clustering Quality in terms of Silhouette and Average scores separation (Score Students Separation colors: one can totally distinguish groups of students (GREEN), one can distinguish quite good (ORANGE) or not at all (RED)).



IPB

Mechanical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		34 %	1 s
1st Course	77 %	76 %	1.1 – 1.4 s
2nd Course	88 %	86 %	1.1 – 1.3 s
3rd Course	93 %	94 %	1.2 – 1.4 s

Civil Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		51 %	1.5 s
1st Course	68 %	68 %	2.1 – 2.8 s
2nd Course	73 %	74 %	2.8 – 3.3 s
3rd Course	93 %	92 %	1.6 – 1.9 s

Electrotechnics Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		46 %	0.6 s
1st Course	69 %	71 %	0.6 - 0.7 s
2nd Course	78 %	78 %	0.6 – 0.8 s
3rd Course	93 %	92 %	0.6 – 0.7 s

Computer Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		51 %	0.8 s
1st Course	73 %	75 %	0.9 – 1 s
2nd Course	85 %	85 %	0.8 – 0.9 s
3rd Course	93 %	91 %	0.8 – 0.9 s

Computer Electrotechnics Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		42 %	0.5 s
1st Course	64 %	63 %	0.6 s
2nd Course	85 %	87 %	0.5 – 0.6 s
3rd Course	86 %	91 %	0.5 – 0.6 s

Chemical Engineering

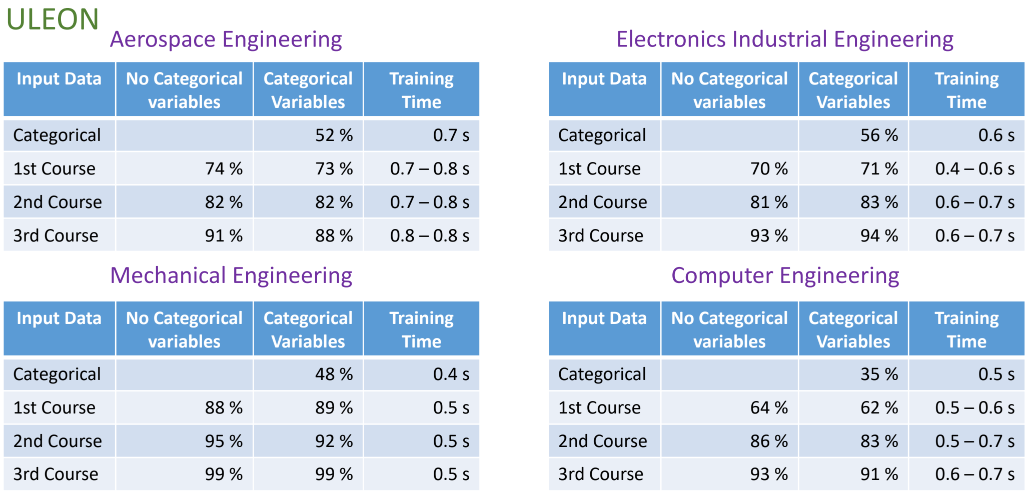
Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		51 %	0.5 s
1st Course	76 %	70 %	0.5 – 0.6 s
2nd Course	83 %	84 %	0.5 s
3rd Course	89 %	88 %	0.5 s

Figure 11. Classification Results obtained with Instituto Politecnico de Bragansa (IPB) degrees.

Besides the classification purpose of this tool (i.e., to classify a new student at the different groups), this tool shows a course-dependency behavior that can be exploited to understand the structure of degrees. In other words, different accuracy contributions are observed when comparing 1st Course, 1st + 2nd Course and 1st + 2nd + 3rd Course results. For instance, those cases reflecting a high accuracy level at 1st Course could mean that the first year of that degree is very important and clearly determines the kind of student. Indeed, this tool is quite useful to extract insights and patterns when comparing Institution and degrees, but this will be analyzed in detail in [BVV+18].

3.4 Code Availability

As previously commented, this tool has been developed in Python. All the code can be found at a bickbucket repository `https://bitbucket.org/SPEET_PROJECT/speet_code`. Due to confidentiality issues, data belonging to the different institutions are not provided. Instead, a toy example is considered.



ULEON

Aerospace Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		52 %	0.7 s
1st Course	74 %	73 %	0.7 – 0.8 s
2nd Course	82 %	82 %	0.7 – 0.8 s
3rd Course	91 %	88 %	0.8 – 0.8 s

Electronics Industrial Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		56 %	0.6 s
1st Course	70 %	71 %	0.4 – 0.6 s
2nd Course	81 %	83 %	0.6 – 0.7 s
3rd Course	93 %	94 %	0.6 – 0.7 s

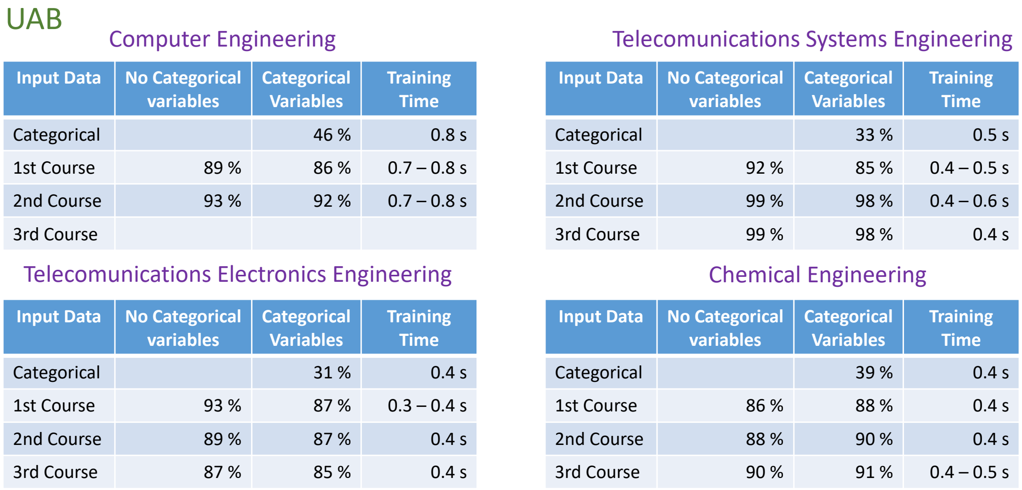
Mechanical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		48 %	0.4 s
1st Course	88 %	89 %	0.5 s
2nd Course	95 %	92 %	0.5 s
3rd Course	99 %	99 %	0.5 s

Computer Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		35 %	0.5 s
1st Course	64 %	62 %	0.5 – 0.6 s
2nd Course	86 %	83 %	0.5 – 0.7 s
3rd Course	93 %	91 %	0.6 – 0.7 s

Figure 12. Classification Results obtained with Universidad de Leon (ULEON) degrees.



UAB

Computer Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		46 %	0.8 s
1st Course	89 %	86 %	0.7 – 0.8 s
2nd Course	93 %	92 %	0.7 – 0.8 s
3rd Course			

Telecomunications Systems Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		33 %	0.5 s
1st Course	92 %	85 %	0.4 – 0.5 s
2nd Course	99 %	98 %	0.4 – 0.6 s
3rd Course	99 %	98 %	0.4 s

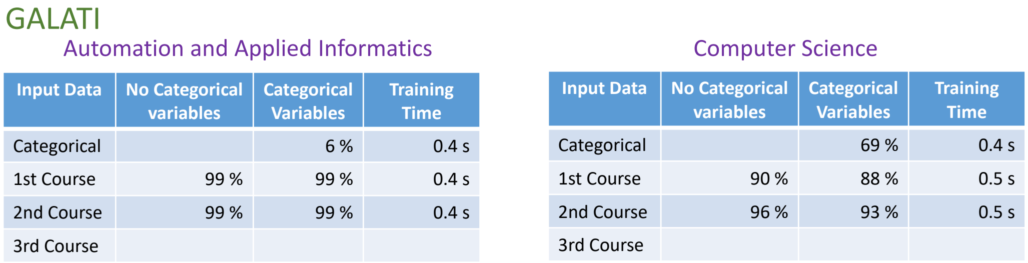
Telecomunications Electronics Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		31 %	0.4 s
1st Course	93 %	87 %	0.3 – 0.4 s
2nd Course	89 %	87 %	0.4 s
3rd Course	87 %	85 %	0.4 s

Chemical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		39 %	0.4 s
1st Course	86 %	88 %	0.4 s
2nd Course	88 %	90 %	0.4 s
3rd Course	90 %	91 %	0.4 – 0.5 s

Figure 13. Classification Results obtained with Universitat Autonoma de Barcelona (UAB) degrees.



GALATI

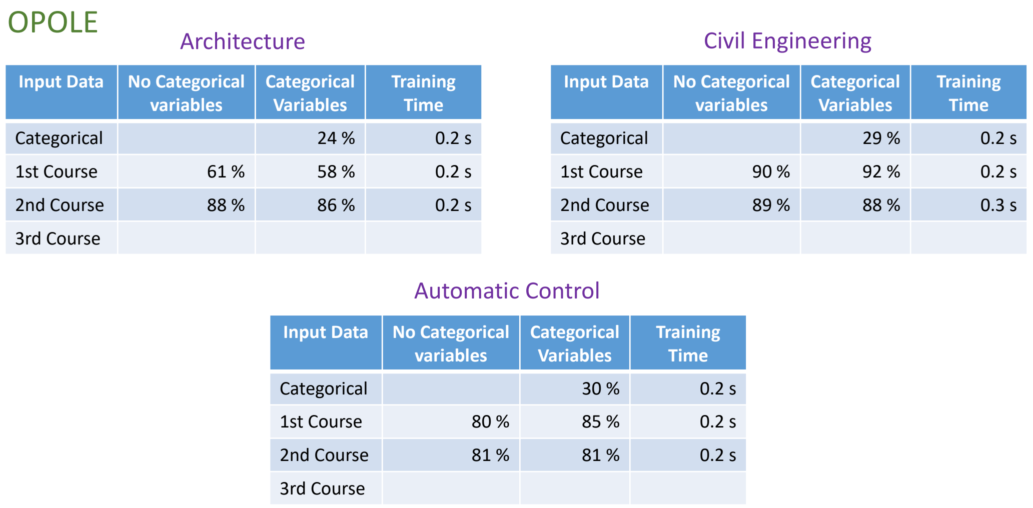
Automation and Applied Informatics

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		6 %	0.4 s
1st Course	99 %	99 %	0.4 s
2nd Course	99 %	99 %	0.4 s
3rd Course			

Computer Science

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		69 %	0.4 s
1st Course	90 %	88 %	0.5 s
2nd Course	96 %	93 %	0.5 s
3rd Course			

Figure 14. Classification Results obtained with Universitatea ”Dunarea de Jos” din Galati (GALATI) degrees.



OPOLE

Architecture

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		24 %	0.2 s
1st Course	61 %	58 %	0.2 s
2nd Course	88 %	86 %	0.2 s
3rd Course			

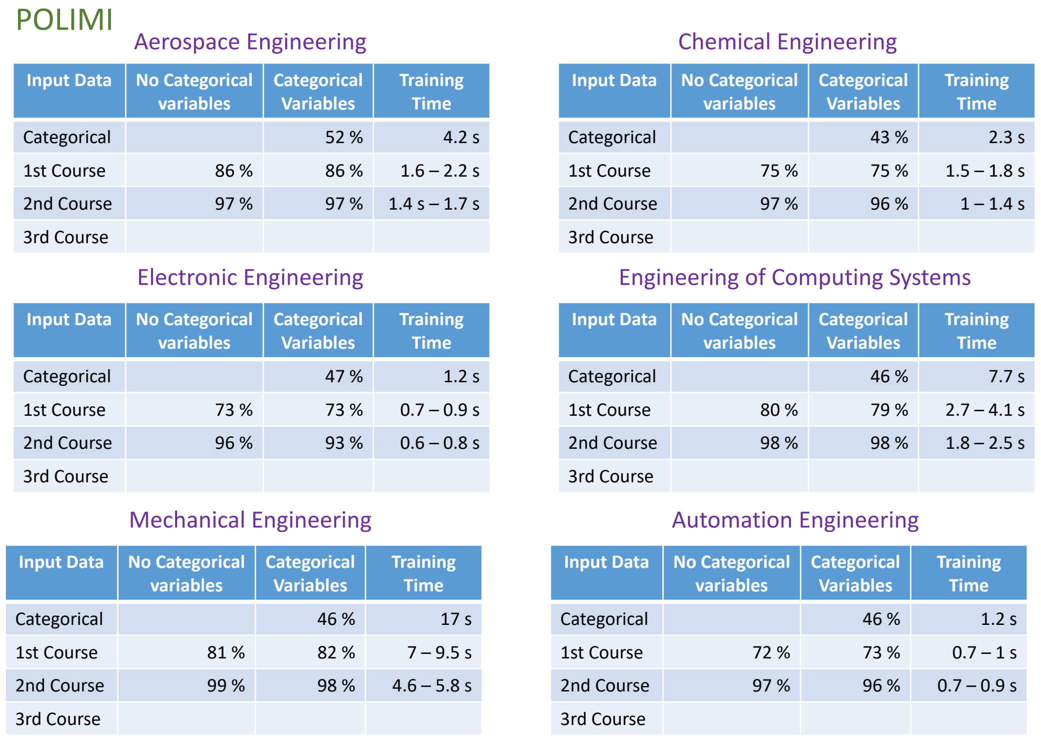
Civil Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		29 %	0.2 s
1st Course	90 %	92 %	0.2 s
2nd Course	89 %	88 %	0.3 s
3rd Course			

Automatic Control

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		30 %	0.2 s
1st Course	80 %	85 %	0.2 s
2nd Course	81 %	81 %	0.2 s
3rd Course			

Figure 15. Classification Results obtained with Politechnika Opolska (OPOLE) degrees.



POLIMI

Aerospace Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		52 %	4.2 s
1st Course	86 %	86 %	1.6 – 2.2 s
2nd Course	97 %	97 %	1.4 s – 1.7 s
3rd Course			

Chemical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		43 %	2.3 s
1st Course	75 %	75 %	1.5 – 1.8 s
2nd Course	97 %	96 %	1 – 1.4 s
3rd Course			

Electronic Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		47 %	1.2 s
1st Course	73 %	73 %	0.7 – 0.9 s
2nd Course	96 %	93 %	0.6 – 0.8 s
3rd Course			

Engineering of Computing Systems

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		46 %	7.7 s
1st Course	80 %	79 %	2.7 – 4.1 s
2nd Course	98 %	98 %	1.8 – 2.5 s
3rd Course			

Mechanical Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		46 %	17 s
1st Course	81 %	82 %	7 – 9.5 s
2nd Course	99 %	98 %	4.6 – 5.8 s
3rd Course			

Automation Engineering

Input Data	No Categorical variables	Categorical Variables	Training Time
Categorical		46 %	1.2 s
1st Course	72 %	73 %	0.7 – 1 s
2nd Course	97 %	96 %	0.7 – 0.9 s
3rd Course			

Figure 16. Classification Results obtained with Politecnico di Milano (POLIMI) degrees.

4 Student drop-out prediction

SPEET project aims to process the data in order to extract information about and to identify student profiles.

The choice of such identification profiles has been made at university level after analyzing all possible options. Despite the vastness of possibilities (for instance: students that finish degree on time, students that are blocked on a certain set of subjects, etc.), the SPEET consortium has decided to continue the analysis by analyzing the distinction between students completing their study programme graduating and those who instead decide to abandon studies. The student profiles we are referring to within the SPEET project scope are

(a) *dropout*: students that leave degree studies

(b) *graduate*: students that get the degree sooner or later

distincion which will be defined by a variable called ”status”.

The choice to analyze such factor can be justified by considering one example. In Italy, almost a student out of two renounces to his engineering degree before the end of the studies. The CNI[2] studies center has lately published some statistics related to the students that choose to study engineering and the numbers underline how the rate of abandonment is elevated, even if the graduates’ number in the sector continues to increase in the years.

4.1 Data Base Format and Pre-Processing

In this section database format and pre-proceesing steps are discussed: data gathering methods are often loosely controlled, resulting in out-of-range values, impossible data combinations, missing values, etc. Analyzing data that has not been carefully screened for such problems can produce misleading results.

[2] Consiglio Nazionale Ingegneri (National Council of Engineers). National organism of institutional representation for the remarkable affairs of the professional category of engineers [Giuseppe Latour, *Ingegneri, una matricola su due non arriva al termine degli studi*, Scuola24 (September 10, 2015) URL:http://www.scuola24.ilsole24ore.com/art/universita-e-ricerca/2015-09-09/ingegneri-matricola-due-non-arriva-termine-studi]

4.1.1 Data Cleaning Process

When working with a real dataset we need to take into account the fact that some data might be missing or corrupted, therefore we need to prepare the dataset for the analysis. As a first step, it is necessary to

- re-allocate students who have changed one or more Engineering Schools during their career: each of them has been ”identified” with the last attended school specifying if he has got the degree or not (in the last attended school).
- omit missing values, i.e. data values not stored for some variables in some observations. Missing data are a common occurrence and can have a significant effect on the conclusions that can be drawn from the data. Missing data can occur because of nonresponse: no information is provided for one or more items or for a whole unit.
- remove outliers and corrupted values, i.e. observations (or set of observations) which appear to be inconsistent with that set of data. The inconsistencies detected may have been originally caused by user entry errors, by corruption in transmission or storage, or by different data dictionary definitions of similar entities in different stores.
- omit some Engineering Schools because of two fundamental reasons:
  - negligible number of students (online programme university with few students, etc)
  - the sample of available data is not sufficiently reliable (Engineering School established few years ago)
- remove ”suspended” careers, i.e. students that have decided to interrupt their career just for the moment
- remove ”active” careers i.e. students that have not yet concluded their studies
- remove all careers that began in 2013, 2014 and 2015. This is done to avoid an increased number of dropouts without a counterbalance of graduate students. In other words, most of these careers are still active and therefore have already been excluded from the sample size but, however, several students registered in 2013, 2014 and 2015 ended their study programmes by choosing dropout.

These preprocessing steps, cleaning and formatting of the data, often is crucial for obtaining a good fit of the model and better predictive ability.

4.1.2 Categorical vs Numerical/Performance Data

Higher education institutions have always operated in an information-rich landscape, generating and collecting vast amounts of data each day. The academic records of students are stored in the offices of our Engineering Schools and they do not only include the performance of students on different subjects of the degree but also collateral information (geographical info, previous studies, age, etc). This information could be used to help characterise the student by means of data science techniques and, as a result, help tutors to better understand their students and improve counselling actions. To make easier the data import and to summarize the fundamental characteristics of each student, three data files are defined, namely

Student Explanatory Information

Variable	**Description**	**Type of variable**
YearOfBirth	year of birth	natural number
PlaceOfBirth	place of birth	factor
Sex	sex	factor (female, male)
ResidenceCity	city of residence	factor
Nationality	nationality	factor
PreviousStudies	high school studies	factor (sciences secondary, technological secondary, literature secondary, professional studies, etc)
PreviousStudiesCenter	high school location	factor
PreviousAcadStudies	has the student attended another university before and earned a Bachelor's degree?	factor (yes or not)
PreviousAcadStudiesNature	type of previous degree	factor (social sciences, medicine, etc)
PreviousAcadStudiesCenter	university where the student has previously studied	factor
AdmissionScore	admission test score	real number
AccessToStudiesAge	age at the beginning of the university studies	natural number
AccessToStudiesYear	enrolment year	natural number
Status	the way the student has finished the university	factor (Graduated, Dropouts, Momentary Interruption)
EndStudiesYear	year of the end of the career	natural number
StartingDegreeID	degree chosen by the student when he begun his career	factor (PhysicsEngineering, ElectricalEngineering, CivilEngineering, etc)
FinalDegreeID	degree at the end of the student career	factor (PhysicsEngineering, ElectricalEngineering, CivilEngineering, etc)

Table 1. List of variables related to the student explanatory information

Degree Information

Variable	Description	Type of variable
Institution	attended university	factor (UAB, POLIMI, ULEON, GALATI, OPOLE, BRAGANA)
DegreeArea	degrees in Engineering, Design, Architecture, ... ?	factor (Engineering, Design, Architecture, etc.)
Degree	type of degree	factor (Bachelor's degree, Master degree)
DegreeNature	degree study programme	factor (PhysicsEngineering, ElectricalEngineering, CivilEngineering, etc)

Table 2. List of variables related to the degree information

Student Performance Information

Variable	Description	Type of variable
YearsToFinishDegree	EndStudiesYear – AccessToStudiesYear	natural number
Mobility	indicator of the choice to spend a period abroad	factor (No, Erasmus, DoubleDegree, etc)
StartMobility	when (year) mobility started	natural number
EndMobility	when (year) mobility ended	natural number
Subject1NumberECTS	study credits of subject1	real number
Subject1Year	when (year) subject1 lessons have been attended	natural number
Subject1Semester	when (semester) subject1 lessons have been attended	natural number (1 or 2)
Subject1KnowledgeArea	knowledge area of subject1	factor (Area1, Area2, ... , AreaM)
Subject1Language	language of subject1	factor (Country Language, English, Other)
Subject1NumberStudents	number of students who have attended subject1 lessons	natural number
Subject1Score	score of subject1	real number
Subject1Lode	has the student gained the lode?	factor (yes or not)
Subject1NumberAttemps	number of attempts for subject1	natural number
Subject1AverageScore	average score of the students' class for subject1	real number
Subject1FailureRate	failure rate of the students' class for subject1	real number
Subject2 . . .	. . .	. . .
. . .	. . .	. . .
SubjectM . . .	. . .	. . .

Table 3. List of variables related to the student performance information

The variables chosen to fit the models have been selected taking into account the whole available descriptive framework. The nature of most of the variables in the dataset is self explanatory, with the following exceptions: YearOfBirth and AccessToStudiesYear will be treated as factors (i.e. as categorical predictors).

As each Engineering School envisages a differentiated study plan, the information on subjects scores can be grouped by computing:

- the weighted average (based on study credits associated with each subject) of the evaluations of each passed exam for each student. If the student did not pass any exams, the variable is set to 0;
- the average number of attempts for each subject that the student entered in his study plan (passed and not passed exams). Of course, for the calculation of the average, the exams that have never been ”tempted” by the student are not taken into account. If the student never attempted any exams, the variable is set to 0.

Determinants of students' performance have been the subject of ongoing debate among educators, academics, and policy makers. There have been many studies that sought to examine this issue and their findings point out to hard work, previous schooling, parents' education, family income and self motivation as factors that have a significant effect on the students behaviour.

The purpose of the SPEET investigation is to find out what are the factors that affect the performance of the students in terms of admission score, weighted average of the evaluations and average number of attempts per exam.

4.1.3 Preliminary Analysis

The main statistical tool used for this purpose is the ANOVA analysis. ANOVA is short for ANalysis Of VAriance. The main purpose of an one-way ANOVA is to assess for significant differences on a continuous dependent variable by a categorical independent variable (with two or more groups). It compares the means between the groups you are interested in and determines whether any of those means are statistically significantly different from each other.

Considering PoliMi dataset, fifteen (5 tested factors $*$ 3 responses) independent one-way anova analysis have been performed testing the significance of factors Access To Studies Year, Sex, Mobility, Previous Studies, Change of Faculty over admission score, weighted average of the evaluations and average number of attempts per exam.

The collection of data also includes geographic information that allows us to identify the origin of each student. Since cultural differences may play a role in shaping the factors that affect students' performance, this information is used to find out whether there are any differences among national and foreign students .

Before proceeding with the analysis of the results, it is important to consider the assumptions made by ANOVA:

I. the groups defined by the categorical variable have the same variance (homogeneity of variance)

II. the groups are normally distributed.

Since in most of the cases we have evidence that the variance of the groups differ significantly (Bartlett tests have been performed and almost all the p-values are < 0.05 proving that the hypothesis of homogeneity of variances is not verified), the ANOVA results (p-values, etc.) have not been considered as statistically valid. Indeed, the one-way ANOVA is considered a robust test against the normality assumption, this means that it tolerates violations to its normality assumption rather well; on the contrary variance heterogeneity is not admissible.

Kruskal-Wallis is an alternative, non-parametric (distribution free) test, and it is used when the assumptions of one-way ANOVA are not met. All the p-values (obtained with Kruskal-Wallis tests) are lower than 0.05 point out significant evidence of the factor on the response variable.

PoliMi results show that almost all the factors investigated affects student' performance with the exception of the variable concerning the previous studies of the student.

4.2 Materials and Methods

Logistic regression models and Logistic Mixed-Effects regressions models can be applied to examine the relationship between the success probability (getting the degree) and a set of attributes for each student such as sex, year of birth etc. With such classification that, of course, devise a more precise definition and categorisation the more usual student patterns will be depicted.

4.2.1 Materials

As a specification of the SPEET project, the focus of the analysis is concentrated on three pieces of information related to the first semester of the first year that the student spent in the univerisity with the hope that these could be significant in order to predict a student' status (for example graduate/dropout) through the first student' performance information. Namely

- the weighted average (based on study credits associated with each subject) of the evaluations of each passed exam for each student in the first semester of the first year that the student spent in his university. If the student did not pass any exams, the variable is set to 0
- the average number of attempts for each subject that the student entered in his study plan (passed and not passed exams) in the first semester of the first year that he spent in his university. Of course, for the calculation of the average, the exams that have never been ”tempted” by the student are not taken into account. If the student never attempted any exams, the variable is set to 0
- number of passed exams in the first semester of the first year that the student spent in his university according to the chosen study plan.

The overall collection of the selected predictor variables is the following

Variable	**Description**	**Type of variable**
YearOfBirth	year of birth	natural number
Sex	sex	factor (female, male)
Nationality	nationality	factor
PreviousStudies	high school studies	factor (sciences secondary, technological secondary, literature secondary, professional studies, etc)
PreviousAcadStudies	has the students attended another university before PoliMi and earned a Bachelor's degree?	factor (yes or not)
DegreeNature	degree study programme	factor (PhysicsEngineering, ElectricalEngineering, CivilEngineering, etc)
AdmissionScore	PoliMi admission test score	real number
AccessToStudiesAge	age at the beginning of the studies in PoliMi	natural number
AccessToStudiesYear	enrolment year in PoliMi	natural number

Variable	Description	Type of variable
Mobility	indicator of the choice to spent a period abroad	factor (yes or not)
WeightedAverageEvaluations	weighted average of the evaluations (passed exams, score $\geq$ 18)	real number
WeightedAverageEvaluations_11	weighted average of the evaluations in the first semester of the first year that the student spent in PoliMi (passed exams, score $\geq$ 18)	real number
AverageNumbAttemptsPerExam	average number of attempts for each subject that the student entered in his study plan (passed and not passed exams)	real number
AverageNumbAttemptsPerExam_11	average number of attempts for each subject that the student entered in his study plan (passed and not passed exams) in the first semester of the first year that the student spent in PoliMi	real number
NumbSubjectsPassed_11	number of passed exams in the first semester of the first year that the student spent in PoliMi	natural number
Change	has the student changeed Engineering School during his career?	factor (yes or not)

Table 4. Proposed covariates for the GLM models

4.2.2 Methods

As a first approach, despite of the vastness of classification algorithms, the SPEET consortium's choice is to use regression models. More specifically, the following mechanisms have been tested:

- **Simple Logistic Model**: In logistic regression, a categorical dependent variable y having two unique values is regressed on a set of k independent variables $x_1, x_2, \ldots, x_k$.
  The mean of the response variable p, in terms of explanatory variables x_1, x_2, $\ldots$, x_k, is modeled relating p and x_1, x_2, $\ldots$, x_k through the equation $p = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$. Unfortunately, this is not a good model be-

cause extreme values of $x_1, x_2, \ldots, x_k$ will give values of $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$ that does not fall between 0 and 1. The logistic regression solution to this problem is to use the logit. We can model the natural log odds as a linear function of the explanatory variables

$$l = logit(p) = ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \tag{1}$$

The `glm` function, in `stats`[3] R [4] package, implements such a logit model.

A simple logistic model can be applied to examine the relationship between the success probability (getting the degree) and a set of attributes for each student such as sex, year of birth etc., but the analysis has to be conducted independently for each one of the Engineering Schools within the same university, in order to not to lose the grouped nature of our database.
A prototype of the suggested model is the following:

$$\begin{aligned} p_j = P(status_j = graduate) = P(status_j = 1) \\ logit(p_j) = ln\left(\frac{p_j}{1-p_j}\right) = \beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j} + \cdots + \beta_k x_{kj} \end{aligned} \tag{2}$$

where p_j is the graduating probability for student j, $x_1, x_2, \ldots, x_k$ the explanatory variables and $\beta_1, \beta_2, \ldots, \beta_k$ the estimated coefficients.

- **Logistic Mixed-Effects Model**: Mixed-effects models are primarily used to describe relationships between a response variable and some covariates in data that are grouped according to one or more classification factors. Examples of such grouped data include longitudinal data, repeated measures data, multilevel data, and block designs. By associating common random effects to observations sharing the same level of a classification factor, mixed-effects models flexibly represent the covariance structure induced by the grouping of the data.

When dealing with response data that is binary in nature, we use what is often called logistic mixed-effects model. These are quite similar to ”ordinary” logistic models. The model form for a single observation y_{ij}, $j = 1, ..., n_i$ in group i, $i = 1, ..., M$ is

$$p_{ij} = P(y_{ij} = 1 | \boldsymbol{b}_i = [b_{i0} \quad b_{i1} \quad \ldots \quad b_{iq-1}]^T]) = \frac{exp(\sum_{k=0}^{p-1} x_{ijk}\beta_k + \sum_{h=0}^{q-1} z_{ijh} b_{ih})}{1 + exp(\sum_{k=0}^{p-1} x_{ijk}\beta_k + \sum_{h=0}^{q-1} z_{ijh} b_{ih})}$$

$$log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \sum_{k=0}^{p-1} x_{ijk}\beta_k + \sum_{h=0}^{q-1} z_{ijh} b_{ih} \tag{3}$$

[3] The `stats` package is part of R

[4] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

$$logit(p_{ij}) = \sum_{k=0}^{p-1} x_{ijk}\beta_k + \sum_{h=0}^{q-1} z_{ijh}b_{ih} \tag{4}$$

where y_{ij} and ε_{ij} denote observation and error j in group i (the number of observations may vary by group). x_{ijk} and z_{ijh} represent, respectively, the (j,k) element of matrix X_i (of size $n_i \times p$) and the (j,h) element of Z_i (of size $n_i \times q$), that is, the values of explanatory variables for fixed and random effects model parameters. While, $\boldsymbol{\beta} = [\beta_0 \quad \beta_1 \quad ... \quad \beta_{p-1}]^T$ is the p-dimensional vector of fixed effects and $\boldsymbol{b}_i = [b_{i0} \quad b_{i1} \quad ... \quad b_{iq-1}]^T$ is the q-dimensional vector of random effects.

Let $\mu_{ij} = E[y_{ij}|\boldsymbol{b}_i]$, the linear predictor for a Logistic Mixed-Effects model has the form

$$g(\mu_{ij}) = \sum_{k=0}^{p-1} x_{ijk}\beta_k + \sum_{h=0}^{q-1} z_{ijh}b_{ih} \qquad i = 1,...,M \qquad j = 1,...,n_i \tag{5}$$

in which g() is the logit link. The random effect vector $\boldsymbol{b}_i = [b_{i0} \quad b_{i1} \quad ... \quad b_{iq-1}]^T$ is assumed to have a multivariate normal distribution $N(0,\Psi)$. The covariance matrix Ψ depends on unknown variance components and possibly also correlation parameters.

Linear Mixed Models are based on maximum likelihood (ML) or restricted maximum likelihood (REML). Model fitting is rather complex for GLME (Generalized Mixed-Effects) models. Numerical methods for approximating it can be computationally intensive for models with multivariate random effects. In this project, several approaches have been explored to deal with these computational difficulties: Gauss-Hermite Quadrature Methods, Monte Carlo EM Methods, Penalized Quasi-likelihood Approximation and Bayesian Approaches.

A logistic mixed-effects model can be applied to examine the relationship between the success probability (getting the degree) and a set of attributes for each student such as sex, year of birth etc according to the grouping factor DegreeNature (= Engineering School within the same university).
A prototype of the suggested model is the following:

$$\begin{gathered} logit(p_{ij}) = \sum_{k=0}^{p-1} x_{ijk}\beta_k + \sum_{h=0}^{q-1} z_{ijh}b_{ih}, \qquad i = 1,...,M \qquad j = 1,...,n_i \\ \boldsymbol{b}_i = [b_{i0} \quad b_{i1} \quad ... \quad b_{iq-1}]^T \sim N(0,\Psi) \end{gathered} \tag{6}$$

where p_{ij} is the graduating probability for student j in group i, x_{ijk} and z_{ijh} represent, respectively, the (j,k) element of matrix X_i (of size $n_i \times p$) and the (j,h) element of Z_i (of size $n_i \times q$), that is, the values of explanatory variables for fixed and random effects model parameters. While, $\boldsymbol{\beta} = [\beta_0 \quad \beta_1 \quad ... \quad \beta_{p-1}]^T$ is the p-dimensional vector of fixed effects and $\boldsymbol{b}_i = [b_{i0} \quad b_{i1} \quad ... \quad b_{iq-1}]^T$ is the q-dimensional vector of random effects.

4.3 Implemented Algorithms

Next Subections will cover the implementation of considered algorithms and the analysis of their performance with Politecnico di Milano (PoLiMi) degrees. Before proceeding with the implementation of a generalized mixed-effects model, simple logistic regression modeling procedures are carried out for each one of the Engineering Schools considering them as independent samples. Finally, in order to obtain a single model that takes into account the ”grouped” nature of the data, a generalized linear mixed-effects model (GLME) is implemented with the purpose of describing the relationship between the success probability (getting the degree) and the covariates using exactly the data as ”grouped” according to one classification factor (the Engineering School).

4.3.1 Simple Logit Models

The goal of this section is to predict the ’survival’ probability to PoliMi Engineeering Schools (either 1 if the student gets the degree or 0 if not) based on some features. We will treat variables AdmissionScore, WeightedAverageEvaluations, WeightedAverageEvaluations_11, AverageNumbAttemptsPerExam and AverageNumbAttemptsPerExam_11 as continuous, while AccessToStudiesAge and NumbSubjectsPassed_11 as discrete. Both the variables YearOfBirth and AccessToStudiesYear are treated as factors (i.e. as categorical predictors).

The main goal whihin the SPEET purpose is to build a logit model for each of the 19 PoliMi Engineering Schools where p_j, such that

$$\begin{gathered} p_j = P(status_j = graduate) = P(status_j = 1) \\ logit(p_j) = ln\left(\frac{p_j}{1-p_j}\right) = \beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j} + \cdots + \beta_1 5 x_{15j} \end{gathered} \tag{7}$$

is the graduating probability for student j.

Implementation and Interpretation of the Results

Data of each Engineering School at Politecnico di Milano have been divided into two chunks: training and testing set. The training set (about 80%) will be used to fit the models which will be tested over the testing set (about 20%).

For the sake of simplicity we include in this section only the model associated with Mathematical Engineering.

The initial model include all the covariates expressed in Table 4. At each step, less significant covariates have been excluded (a significant p-value is usually taken as $\leq$ 0.05). These preliminary analysis suggest for Mathematical Engineering students a logit model including the main effects of Sex, AccessToStudiesYear, WeightedAverageEvaluations_11, AverageNumbAttemptsPerExam_11, NumbSubjectsPassed_11, WeightedAverageEvaluations and Change, namely

$$\begin{aligned} logit(p_j) = ln\left(\frac{p_j}{1-p_j}\right) = \quad & \beta_0 \\ & + \beta_1 Sex(male)_j \\ & + \beta_2 AccessToStudiesYear(2010)_j \\ & + \beta_3 AccessToStudiesYear(2011)_j \\ & + \beta_4 AccessToStudiesYear(2012)_j \\ & + \beta_5 WeightedAverageEvaluations\_11_j \\ & + \beta_6 WeightedAverageEvaluations_j \\ & + \beta_7 AverageNumbAttemptsPerExam\_11_j \\ & + \beta_8 NumbSubjectsPassed\_11_j \\ & + \beta_9 Change(yes)_j \end{aligned} \tag{8}$$

for student j.

The `glm` function, in `stats`[5] R [6] package, implements such a logit model.

Variable	**Estimate**	**P-value**
(Intercept)	-21.07423	6.32e-10
Sex(male)	-0.94167	0.03744
AccessToStudiesYear(2010)	1.48046	0.01074
AccessToStudiesYear(2011)	1.83650	0.00311
AccessToStudiesYear(2012)	0.99965	0.12033
WeightedAverageEvaluations_11	-0.16338	0.00110
WeightedAverageEvaluations	0.82720	6.37e-09
AverageNumbAttemptsPerExam_11	1.11412	0.00280
NumbSubjectsPassed_11	2.07738	1.47e-11
Change(yes)	-4.66472	9.77e-06

Table 5. P-values and coefficients estimates for the model of Mathematical Engineering students

As for the statistically significant variables, shown in Table5, NumbSubjectsPassed_11 has the lowest p-value suggesting a strong association of the number of exams

[5] The stats package is part of R

[6] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

passed in the first semester of the first year that the student spent in PoliMi with the probability of getting the degree. Interpreting the results:

- female students outperform their male counterpart, indeed being a man penalizes the log odds by 0.94167
- having been enrolled in a year rather than in another one changes the trend of the log odds. Basically the time increases the response. We would suggest a deeper investigation.
- a unit increase in the weighted average of the evaluations for the exams of the first semester of the first year reduces the log odds by 0.16338. This information suggests that students who face inexplicably the first exams probably thanks to some preliminary knowledges (high school), then, do not make it into the later ones
- a unit increase in the average number of attempts per exam of the first semester of the first year increases the log odds by 1.11412. The stubborn student is rewarded.
- one more exam passed in the first semester of the first year increases the log odds by 2.07738
- being an excellent student with a high weighted average of evaluations increases the log odds by 0.82720
- changing Engineering School during the career negatively affects the log odds reducing it by 4.66472. Probably the fact of not being determined just from the beginning does not give a positive contribution to the graduating probability

We can run the `anova` function, in `stats`[7] R [8] package, on the model to analyze the table of deviance (Listing 4.1).

```
> anova(fit_math, test=”Chisq”)
Analysis of Deviance Table

Model: binomial, link: logit

Response: status

Terms added sequentially (first to last)


                                       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                                                     528     589.96
```

[7] The stats package is part of R

[8] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

```
Sex                             1     0.559      527     589.40     0.4546
AccessToStudiesYear             3     5.700      524     583.70     0.1271
WeightedAverageEvaluations_11   1   260.328      523     323.37  < 2.2e-16 ***
AverageNumbAttemptsPerExam_11   1     0.833      522     322.54     0.3615
NumbSubjectsPassed_11           1    76.121      521     246.41  < 2.2e-16 ***
WeightedAverageEvaluations      1    53.730      520     192.68  2.300e-13 ***
Change                          1    18.865      519     173.82  1.403e-05 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Listing 4.1. Table of deviance for the model of Mathematical Engineering students

The difference between the null deviance and the residual deviance shows how our model is doing against the null model (a model with only the intercept). The wider this gap, the better. Analyzing the table we can see the drop in deviance when adding each variable one at a time. Therefore, adding WeightedAverageEvaluations_11, NumbSubjectsPassed_11, WeightedAverageEvaluations and Change significantly reduce the residual deviance. The other variables seem to improve the model less. A large p-value here, as for Sex, AccessToStudiesYear and AverageNumbAttemptsPerExam_11, indicates that the model without these variables explains more or less the same amount of variation.

Assessing the Predictive Ability

To validate the model we would like to see how it is doing when predicting status on a new set of data, i.e. the testing set selected at the beginning (100 test students). We need probabilities in the form of $P(y_j = 1|x_{1j}, x_{2j}, \ldots, x_{9j})$ where $y_j = status$ of student j. Our decision boundary will be 0.5:

If $P(y_j = 1|x_{1j}, x_{2j}, \ldots, x_{9j}) > 0.5$ then $y_j = 1$ otherwise $y_j = 0$.

Note that this choice is arbitrary, for some applications different thresholds could be a better option.

Observed	**Predicted**	
	1	0
1	78	1
0	4	17

Table 6. Test Sample Classification Table (100 test students)

Thanks to the misclassification error we can obtain an estimate of the model accuracy $= 1 - (4+1)/(78+1+4+17) = 0.95$. The 0.95 accuracy on the test set

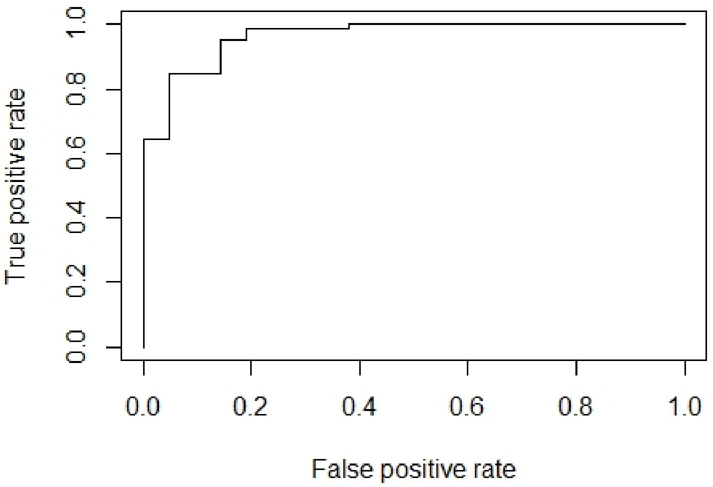


Figure 17. ROC curve, AUC = 0.9638336

is a good result. Moreover, we can consider sensitivity $= 78/(78+1) = 0.9873418$ and specificity $= 17/(4+17) = 0.8095238$. High sensitivity and specificity indicate a good fit of the model.

As a last step, we are going to plot the ROC curve and calculate the AUC (area under the curve) which are typical performance measurements for a binary classifier. The ROC is a curve generated by plotting the true positive rate against the false positive rate at various threshold settings while the AUC is the area under the ROC curve. The `ROCR`[9] R [10] package, draws such a curve (Figure 17).

As a rule of thumb, a model with good predictive ability should have an AUC closer to 1 (1 is ideal) than to 0.5. In our case $AUC = 0.9638336$, a great result for the SPEET purpose.

Overall Results

For each Engineering School a logit model has been worked out and evaluated. The one associated with Mathematical Engineering students is detailed and analyzed in the previous subsections. Let's summarize in Table 7 the significant covariates (with p-value ≤ 0.05) for each of the 19 models. The sign (+) implies that

[9] Sing T, Sander O, Beerenwinkel N and Lengauer T (2005). ?ROCR: visualizing classifier performance in R.? _Bioinformatics_, *21*(20), pp. 7881. URL: http://rocr.bioinf.mpi-sb.mpg.de.

[10] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

a unit increase in the variable increases the log odds, while (-) implies the opposite effect, namely a reduction in the log odds.

The covariates shared by all models are essentially two: WeightedAverageEvaluations and NumbSubjectsPassed_11. Both positively affect the log odds as it was expected. WeightedAverageEvaluations_11 (negative effect in most of the cases) and AverageNumbAttemptsPerExam_11 (positive effect in most of the cases) also appeared to be important factors for 8 logit models over 19. The AverageNumbAttemptsPerExam has a confused impact on the response: positive for Building and Mechanical Engineering, negative for Chemical and Management Engineering. Again, as in the logit model for Mathematical Engineering, female students outperform their male counterpart (significant for 3 models over 19) and having been enrolled in a year rather than in another one changes the trend of the log odds (significant for 4 models over 19). Basically going ahead in time the log odds is reduced (with the only exception of Mathematical Engineering students). Changing Engineering School during the career negatively affects the response, this is significant in 5 logit models over 19.

4.3.2 Logit Mixed-Effects Model

Mixed-effects models provide a flexible and powerful tool for analyzing grouped data. They have gained popularity over the last decade, in part because of the development of reliable and efficient software for fitting and analyzing them. The `lme4`[11] library in R [12] is an example of such software.

- **Initial Model**: Taking into account pre-processing steps (data cleaning process and variables redefinition), we will use a binomial GLMM, specifically, a binary logit mixed-effects model to analyze our data.

  To assess the strength and utility of the predictive relationship of our model we decide to split the data into two chunks: training (80%) and testing set (20%) as we have done for the previous nineteen simple logistic models. Data splitting is useful when we need a quick approximation of performance and we have a very large dataset so that the testing dataset can provide a meaningful estimation of performance.

  The initial model include all the covariates expressed in Table 4, complexity that can cause numerical problems as we will see shortly. Using the `glmer` function there is no provision for autocorrelated within-subject errors, and we

[11] Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67(1), 1-48. doi:10.18637/jss.v067.i01.

[12] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

	proposed covariates														
	YearOfBirth	Sex(male)	Nationality(not italian)	PreviousStudies	PreviousAcadStudies	AdmissionScore	AccessToStudiesAge	AccessToStudiesYear(2010,2011,2012)	Mobility(yes)	WeightedAverageEvaluations	WeightedAverageEvaluations_11	AverageNumbAttemptsPerExam	AverageNumbAttemptsPerExam_11	NumbSubjectsPassed_11	Change(yes)
Aerospace Eng.		-								+	-			+	
Automation Eng.										+	+				-
Biomedical Eng.										+	-		+	+	
Building Eng.										+		+	-	+	
Chemical Eng.										+		-	+	+	
Civil and Environmental Eng.						-				+				+	
Civil Eng.									-	+			-	+	
Electrical Eng.										+				+	
Electronic Eng.										+				+	
Energy Eng.		-								+			+	+	
Eng. of Computing Systems						+				+	-			+	-
Environmental and Land Planning Eng.			-							+				+	
Industrial Production Eng.										+			+	+	
Management Eng.							-	-		+	-	-	+	+	-
Materials and Nanotechnology Eng.										+	-			+	
Mathematical Eng.		-						+		+	-		+	+	-
Mechanical Eng.								-		+	-	+		+	
Physics Eng.										+				+	-
Telecommunications Eng.								-		+				+	

Table 7. Significant covariates for the nineteen logit models

don't have the alternative of using another package. Even without explicit autocorrelation, however, the random effects are complex for a fairly small dataset, and we will try to simplify this part of the model considering a single random effect, specifically on the intercept.

Our initial attempt to fit a GLMM produces a convergence warning:

```
> glmer(status ~ (1|DegreeNature) + YearOfBirth + Sex +Nationality
                 + AdmissionScore +PreviousStudies + Mobility
                 + WeightedAverageEvaluations + AverageNumbAttemptsPerExam
                 + AccessToStudiesYear+ WeightedAverageEvaluations_11
                 + AverageNumbAttemptsPerExam_11 + NumbSubjectsPassed_11
                 + AccessToStudiesAge + Change, data = speet_data, family =
                     binomial)

Warning messages:
  1: In (function (fn, par, lower = rep.int(-Inf, n), upper = rep.int(Inf,  :
    failure to converge in 10000 evaluations
  2: In checkConv(attr(opt, ”derivs”), opt$par, ctrl = control$checkConv,  :
    unable to evaluate scaled gradient
  3: In checkConv(attr(opt, ”derivs”), opt$par, ctrl = control$checkConv,  :
    Model failed to converge: degenerate  Hessian with 8 negative eigenvalues
```

Listing 4.2. First output for the Logit Mixed-Effects Model

Failure to converge is a common occurence in fitting a GLMM. In this case, the function to be maximized to find the estimates is 60-dimensional, with 59[13] fixed-effects parameters and 1 more parameter in Ψ (Ψ is mono-dimensional because we have a single random effect on the intercept).

There are multiple possible causes for the failure to converge; sometimes changing to a different optimizer in the computations can produce convergence. The glmer function makes provision for alternative optimizers, and after a bit of experimentation, we are able to obtain convergence using the bobyqa[14] optimizer in optimx[15] R [16] package. Optimizer bobyqa produces its own warning but nevertheless converges to a solution.

The convergence warning in our initial attempt was likely a false alarm; in general, glmer is conservative about detecting convergence failures. Existing methods are approximations because exact evaluation of the likelihood is

[13] as for categorical variables: YearOfBirth has 33 levels (32 + reference level(1954)); Sex has 2 levels (1 + reference level(Female)); Nationality has 2 levels (1 + reference level(Italian)); PreviousStudies has 14 levels (13 + reference level(Foreign High School)); Mobility has 2 levels (1 + reference level(No)); AccessToStudiesYear has 4 levels (3 + reference level(2009)); Change has 2 levels (1 + reference level(No))

[14] BOBYQA performs derivative-free bound-constrained optimization using an iteratively constructed quadratic approximation for the objective function.

[15] John C. Nash, Ravi Varadhan (2011). Unifying Optimization Algorithms to Aid Software System Users: optimx for R. Journal of Statistical Software, 43(9), 1-14. URL: http://www.jstatsoft.org/v43/i09/.
John C. Nash (2014). On Best Practice Optimization Methods in R. Journal of Statistical Software, 60(2), 1-14. URL: http://www.jstatsoft.org/v60/i02/

[16] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

intractable. The `glmer` function implements various numerical methods, and by default uses a Laplace approximation, which is a compromise between accuracy and computational speed.

- **Improved Model**: Before examining the estimated fixed effects coefficients, we will attempt to simplify the model, removing less significant covariates (a significant p-value is usually taken as ≤ 0.05) and performing a likelihood-ratio test relative to the initial model step-by-step.
  On the basis of these tests, we specify a model whose fixed-effects part is

$$\begin{aligned} logit(p_{ij}) = ln\left(\frac{p_{ij}}{1-p_{ij}}\right) = \quad & \beta_0 \\ & + \beta_1 Sex(male)_{ij} \\ & + \beta_2 Nationality(notitalian)_{ij} \\ & + \beta_3 AccessToStudiesAge_{ij} \\ & + \beta_4 AccessToStudiesYear(2010)_{ij} \\ & + \beta_5 AccessToStudiesYear(2011)_{ij} \\ & + \beta_6 AccessToStudiesYear(2012)_{ij} \\ & + \beta_7 WeightedAverageEvaluations\_11_{ij} \\ & + \beta_8 WeightedAverageEvaluations_{ij} \\ & + \beta_9 AverageNumbAttemptsPerExam\_11_{ij} \\ & + \beta_{10} AverageNumbAttemptsPerExam_{ij} \\ & + \beta_{11} NumbSubjectsPassed\_11_{ij} \\ & + \beta_{12} Change(yes)_{ij} \end{aligned} \tag{9}$$

for student j in group i.

This specification for the fixed-effects can be summarized in the matrix X with 13^{17} columns corresponding to the regressors multipying each of the 13 βs in the fixed-effects part of the model. Matrix Z will be a vector of ones, and the corresponding coviarance matrix has obviously only a variance term.

We obtain the following estimates for the fixed effects and variance component:

Fixed Effects

Variable	**Estimate**	**P-value**
(Intercept)	-8.79155	< 2e-16
Sex(male)	-0.26433	0.002619
Nationality(not italian)	-0.42273	0.019515

[17] as for categorical variables: Sex has 2 levels (1 + reference level(Female)); Nationality has 2 levels (1 + reference level(Italian)); AccessToStudiesYear has 4 levels (3 + reference level(2009)); Change has 2 levels (1 + reference level(No))

Variable	Estimate	P-value
AccessToStudiesAge	-0.14913	5.14e-07
AccessToStudiesYear(2010)	-0.05479	0.547818
AccessToStudiesYear(2011)	-0.35235	0.000131
AccessToStudiesYear(2012)	-0.99251	< 2e-16
WeightedAverageEvaluations_11	-0.03718	1.57e-07
WeightedAverageEvaluations	0.44834	<2e-16
AverageNumbAttemptsPerExam_11	0.40280	2.02e-08
AverageNumbAttemptsPerExam	-0.21780	0.033309
NumbSubjectsPassed_11	1.60767	< 2e-16
Change(yes)	-0.51002	0.000353

Table 8. P-values and coefficients estimates for the reduced Logit Mixed-Effects Model (Fixed Effects)

Random Effect

Grouping factor	Variable	Variance
DegreeNature	(Intercept)	0.9229

Table 9. Variance Ψ estimate for the random effect on the reduced Logit Mixed-Effects Model

As for the statistically significant variables (i.e. those with < 0.05), shown in Table 8, WeightedAverageEvaluations and NumbSubjectsPassed_11 have the lowest p-values suggesting a strong (positive) association of the exams performance (+0.44834) and the number of exams passed in the first semester of the first year (+1.60767) with the probability of getting the degree.

Interpreting the results, female students outperform their male counterpart, indeed being a man penalizes the log odds by 0.26433.

The results show that national students outperform non-national students (-0.42273).

As it was expected the relationship between dependent variable and student? age is negatively related, this is proved by the coefficient value -0.14913. Generally, the aged students have less time to devote to studies and this affects their performances.

Having been enrolled in a year rather than in another one changes the trend of the log odds. Basically the time has a negative influence on the response (note that this result is opposite with respect to the one of the simple logit model for Mathematical Engineering students; a further research is required to explore this relation).

A unit increase in the weighted average of the evaluations for the exams of the first semester of the first year reduces the log odds by 0.03718. It may appear that there is a contradiction between intuition and the sign of the estimated regression coefficient. This information could suggest that students who face inexplicably the first exams probably thanks to some preliminary knowledges (high school), then, do not make it into the later ones, but still it requires more research to explain this phenomenon.

A unit increase in the average number of attempts per exam of the first semester of the first year increases the log odds by 0.40280.

This result is in disagreement with that found for the variable on the average number of attempts throughout the student' career: a unit increase in the average number of attempts per exam reduces the log odds by 0.21780. It may depend on intelligence level, intellect, memory or method of learning of the student, although this value is not negligible yet it reflects the effect of personal characteristics of student.

Changing Engineering School during the career negatively affects the log odds reducing it by 0.51002. Probably the fact of not being determined just from the beginning does not give a positive contribution to the graduating probability.

4.3.3 Final Model: Student Drop-out Predictor

As our main goal is to predict the academic future of a student as soon as possible, we decide to further manipulate the model. In order to predict a student' status (graduate/dropout) through the first student' performance information, we decide to focus our attention on the information available at the end of the first semester of the first year that the student spent in PoliMi. Therefore we choose to delete variables WeightedAverageEvaluations and AverageNumbAttemptsPerExam from the model (WeightedAverageEvaluations_11 and AverageNumbAttemptsPerExam_11 are still considered together with the others).

On the basis of this setting, we specify a final model whose estimates for the fixed effects and variance component are the following:

Fixed Effects

Variable	**Estimate**	**P-value**
(Intercept)	-2.322716	2.47e-05
Sex(male)	-0.292086	0.000447
Nationality(not italian)	-0.423296	0.020026
AccessToStudiesAge	-0.054090	0.029240

Variable	Estimate	P-value
AccessToStudiesYear(2010)	-0.022170	0.801236
AccessToStudiesYear(2011)	-0.344638	8.32e-05
AccessToStudiesYear(2012)	-0.844056	< 2e-16
WeightedAverageEvaluations_11	0.060766	< 2e-16
AverageNumbAttemptsPerExam_11	0.028752	0.562176
NumbSubjectsPassed_11	1.709591	< 2e-16
Change(yes)	-0.373339	0.011962

Table 10. P-values and coefficients estimates for the final Logit Mixed-Effects Model (Fixed Effects)

Random Effect

Grouping factor	Variable	Variance
DegreeNature	(Intercept)	1.062

Table 11. Variance Ψ estimate for the random effect on the final Logit Mixed-Effects Model

The same considerations of the previous subsection on the significant variables can be considered as true. Note that a unit increase in the weighted average of the evaluations for the exams of the first semester of the first year increases the log odds by 0.060766. The effect is positive, opposed to the coefficient estimate of the previous model in Table 8. This result is certainly more reasonable and realistic. Probably the previous regression model was overspecified, i.e. the regression equation contained one or more redundant predictor variables. Redundant predictors can lead to problems such as inflated standard errors for the regression coefficients estimates.

This final model reflects all the key features within the project scope.

4.4 Case Study Applications

This section is aimed at showing results obtained with the final model presented above when applied to PoLiMI students' data.

4.4.1 Dealing with Random Effects

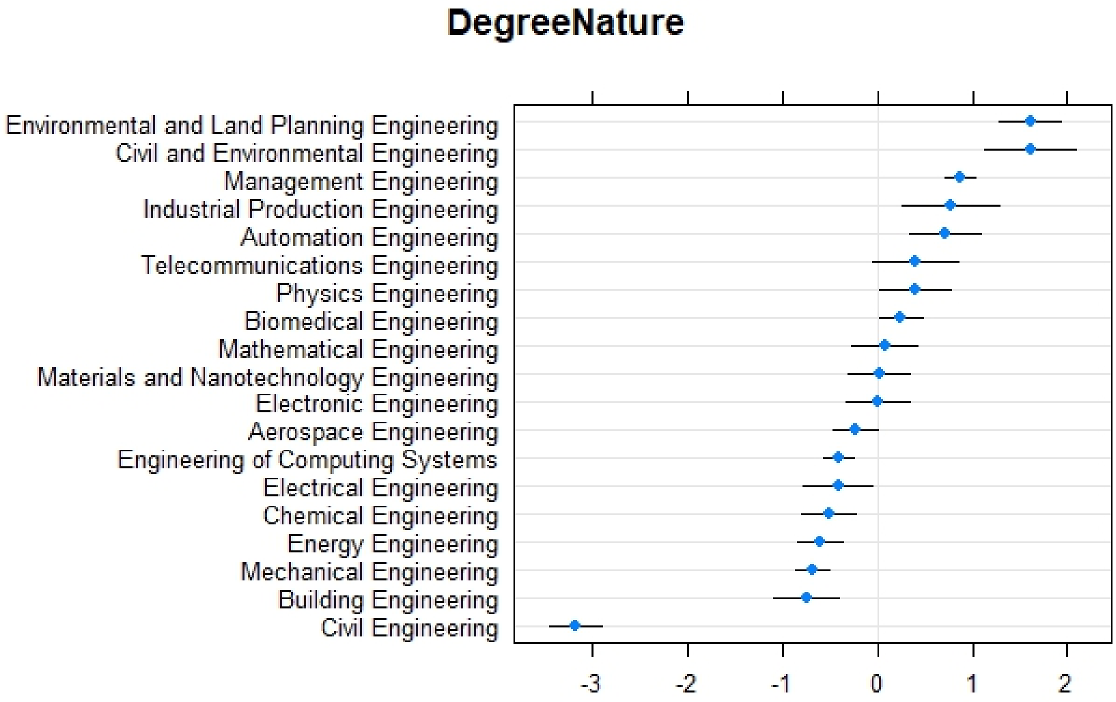


Figure 18. Random effects overview

We will now examine the estimate of the DegreeNature effect on the overall regression model.

Figure 18 shows the estimated random effects for all 19 communities in the dataset. By default, dotplot function from `lattice`[18] R[19] package reorders the random effects by their point estimate. In most of the Engineering Schools, the 95% confidence interval does not overlap the vertical line at zero, indicating that the graduating probability in these schools is significantly different from the average (crossing the zero line). Thus the choice of such mixed-effect regression model seems to be appropriate.

4.4.2 Prediction

So far, we have been worrying about coefficients, but the real model output are the fitted values. Our main interest is in ”predicting”. The predicted values are probabilities (p) and are therefore restricted to $(0,1)$. Our decision boundary will be 0.5. If $p > 0.5$ then $y = 1 := graduate$, otherwise $y = 0 := dropout$. Note that

[18] Sarkar, Deepayan (2008) Lattice: Multivariate Data Visualization with R. Springer, New York. ISBN 978-0-387-75968-5

[19] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

Variable	Value
Sex	'Male'
Nationality	'Italian'
AccessToStudiesAge	19
AccessToStudiesYear	'2012'
WeightedAverageEvaluations_11	23.5
AverageNumbAttemptsPerExam_11	1.8
NumbSubjectsPassed_11	3
Change	'No'

Table 12. Student A profile

this choice is arbitrary, for some applications different thresholds could be a better option.

Following estimation of effects from a generalised linear mixed-effects model, it is useful to form predicted values for certain factor/covariate combinations. This process has been well defined for simple linear models, but the introduction of random effects into the model means that a decision has to be made about the inclusion or exclusion of random model terms from the predictions. This section discusses the importance of analyzing predictions formed including rather than excluding the random term on the intercept of our model.

As an initial example we chose a random student profile: personal characteristics are defined in Table 12, any specifics about his Engineering School is indicated.

Considering the profile of student A and using the `predict` function from `lme4`[20] library in R [21], we can get an estimate of the predicted value for the logit quantity and for the probability of success (which of course depends on the logit value).

Excluding the random term on the intercept of our model thanks to `re.form`[22] = `NA` option we get

```
> newDat <- data.frame(Sex                           = 'Male',
                       Nationality                   = 'Italian',
```

[20] Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67(1), 1-48. doi:10.18637/jss.v067.i01.

[21] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

[22] `re.form` : formula for random effects to condition on. If NULL, include all random effects; if NA or 0, include no random effects

```
                           AccessToStudiesAge             = 19,
                           AccessToStudiesYear            = ’2012’,
                           WeightedAverageEvaluations_11  = 23.5,
                           AverageNumbAttemptsPerExam_11  = 1.8,
                           NumbSubjectsPassed_11          = 3,
                           Change                         = ’No’)
> predict(glme_2,newDat,re.form=NA, type=”link”)          # logit
  2.121949
> predict(glme_2,newDat,re.form=NA, type=”response”)      # success probability
  0.8930183
```

Listing 4.3. Prediction on student A profile without considering the random effect

i.e. student A has 89.30% chance of graduating.

The dropping random effects from predictions does not re-estimate the reduced model, it just sets the random effects to 0, then studying in an Engineering School rather than in another one does not change the final result.

Considering the random effect term and iterating the `predict` command on all Engineering Schools we get the following success probabilities:

```
> school <- c(’Aerospace Engineering’,’Automation Engineering’,
              ’Biomedical Engineering’,’Building Engineering’,
              ’Chemical Engineering’, ’Civil and Environmental Engineering’,
              ’Civil Engineering’,’Electrical Engineering’,
              ’Electronic Engineering’,’Energy Engineering’,
              ’Engineering of Computing Systems’,
              ’Environmental and Land Planning Engineering’,
              ’Industrial Production Engineering’, ’Management Engineering’,
              ’Materials and Nanotechnology Engineering’,
              ’Mathematical Engineering’,’Mechanical Engineering’,
              ’Physics Engineering’,’Telecommunications Engineering’)

> prob<-NULL
> for (i in 1:19) {
   newDat <- data.frame(Sex                            = ’Male’,
                        Nationality                    = ’Italian’,
                        AccessToStudiesAge             = 19,
                        AccessToStudiesYear            = ’2012’,
                        WeightedAverageEvaluations_11  = 23.5,
                        AverageNumbAttemptsPerExam_11  = 1.8,
                        NumbSubjectsPassed_11          = 3,
                        Change                         = ’No’,
                        DegreeNature                   = school[i],)
  prob <- c(prob,predict(glme_2,newDat,re.form=NULL,type=”response”))
  }

> prob
  0.8680009 0.9444031 0.9139439 0.7969962 0.8331766 0.9765795 0.2574750 0.8463671
  0.8935796 0.8202989 0.8466989 0.9766516 0.9474861 0.9523711 0.8949468 0.9003653
  0.8081273 0.9251595 0.9254372
```

Listing 4.4. Prediction on student A profile considering the random effect

For each Engineering School, different success rates have been estimated for student A. Note two almost opposing particular cases: success probability = 97.66%

for Environmental and Land Planning Engineering and success probability = 25.75% for Civil Engineering.

As an example of how a minimum difference can change the estimate of the percentage of success, I consider student B and C profiles defined in Table 13.

For student B the estimated success percentage is 25.75%, while for student C 31.71% (Listing 4.5). Note that the only difference between profile B and profile C is the sex of the student.

```
> newDat_male <- data.frame(Sex                            = ’Male’,
                            Nationality                    = ’Italian’,
                            AccessToStudiesAge             = 19,
                            AccessToStudiesYear            = ’2012’,
                            WeightedAverageEvaluations_11  = 23.5,
                            AverageNumbAttemptsPerExam_11  = 1.8,
                            NumbSubjectsPassed_11          = 3,
                            Change                         = ’No’,
                            DegreeNature                   = ’Civil Engineering’)
> predict(glme_2,newDat_male, re.form=NULL, type=”link”)          # logit
  -1.059134
> predict(glme_2,newDat_male, re.form=NULL, type=”response”)      # success
  0.257475                                                        # probability


> newDat_female <- data.frame(Sex                            = ’Female’,
                              Nationality                    = ’Italian’,
                              AccessToStudiesAge             = 19,
                              AccessToStudiesYear            = ’2012’,
                              WeightedAverageEvaluations_11  = 23.5,
                              AverageNumbAttemptsPerExam_11  = 1.8,
                              NumbSubjectsPassed_11          = 3,
                              Change                         = ’No’,
                              DegreeNature                   = ’Civil Engineering’)
> predict(glme_2,newDat_female, re.form=NULL, type=”link”)          # logit
  -0.7670477
> predict(glme_2,newDat_female, re.form=NULL, type=”response”)      # success
  0.3171181                                                         # probability
```

Listing 4.5. Prediction on student B and C profiles

4.4.3 Model Validation

Often it can be informative to say something about the model quality looking at the fitted values. Goodness-of-fit measures assess the relation between fitted (i.e. predicted) values and actually observed outcomes. In a logit model fitted values are predicted log-odds (and hence predicted probabilities) of outcome.

(a) Student B profile

Variable	**Value**
Sex	’Male’
Nationality	’Italian’
AccessToStudiesAge	19
AccessToStudiesYear	’2012’
WeightedAverageEvaluations_11	23.5
AverageNumbAttemptsPerExam_11	1.8
NumbSubjectsPassed_11	3
Change	’No’
DegreeNature	’Civil Engineering’

(b) Student C profile

Variable	**Value**
Sex	’Female’
Nationality	’Italian’
AccessToStudiesAge	19
AccessToStudiesYear	’2012’
WeightedAverageEvaluations_11	23.5
AverageNumbAttemptsPerExam_11	1.8
NumbSubjectsPassed_11	3
Change	’No’
DegreeNature	’Civil Engineering’

Table 13. Students B and C profiles

Standard model output in R[23] usually includes such measures:

```
    AIC     BIC   logLik deviance df.resid
 7068.7  7158.3  -3522.4   7044.7    12829
```

Listing 4.6. Measures built on data likelihood for the final Logit Mixed-Effects Model

Akaike information criterion (AIC), Bayesian information criterion (BIC) and deviance values should always be positive, smaller is better.

Taking as reference the initial model with all the covariates, AIC and BIC indexes have decreased their values (from $AIC = 7721.6$ to $AIC = 7068.7$, from $BIC = 8182.5$ to $BIC = 7158.3$), which implies a model improvement.

The variance partition coefficient (VPC) is calculated as

$$VPC_{ij} = \frac{\Psi}{\Psi + \sigma^2} \qquad i = 1,...,19 \quad j = 1,...,n_i \tag{10}$$

where VPC_{ij} is the percentage of variation explained by the Engineering School level differences for individual j in school i. Note that we simply have a random intercept then the VPC is constant across individuals.

From theory, the standard logistic distribution has variance

$$\sigma^2 = \pi^2/3 = 3.29 \tag{11}$$

Then with

$$\Psi = 1.062364 \qquad\qquad (Table 9) \tag{12}$$

we get

$$VPC = \frac{1.062364}{1.062364 + 3.29} = 0.244089 \tag{13}$$

Thus, 24.41% of the residual variation in the propensity to get the degree is attributable to unobserved community characteristics. Once again the choice of such mixed-effect regression model seems to be appropriate.

4.4.4 Model Accuracy

When developing models for prediction, the most critical metric regards how well the model does in predicting the target variable on out of sample observations. This is typically done by estimating accuracy using data that was not used to train

[23] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

the model such as a test set, as we have done with simple logistic models. The process involves using the model estimates to predict values on the training set. Afterwards, we will compared the predicted target variable versus the observed values for each observation.

We use 12859 of sampling for model development and keep 3200 sampling to check model accuracy. Based on the proposed model, we compute predicted graduating probability, then looking at the difference between observed and predicted, for those 3200 cases, we find

Observed	**Predicted**	
	1	0
1	2221	68
0	224	687

Table 14. Test Sample Classification Table (3200 test students)

Thanks to the misclassification error we can obtain an estimate of the model accuracy $= 1 - (68+224)/(2221+68+224+687) = 0.90875$. The 90.87% accuracy on the test set is a very good result. Moreover, we can consider sensitivity $= 2221/(2221+68) = 0.9702927$ and specificity $= 687/(224+687) = 0.7541164$. High sensitivity and specificity indicate a good fit of the model.

4.5 Code Availability

As previously commented, this tool has been developed in R. In this case, a repository is not considered as the code has been included at the previous sections in an inline format.

Bibliography

[AH10] L.J. Abdi. H., Williams. *Principal component analysis*. Wiley, 2010.

[AS01] P. Agell and J.A. Segarra. *Escuchando la voz del mercado: Decisiones de segmentacion y posicionamiento (original document in Spanish)*. EUNSA: Manuales IESE, 2001.

[Bis95] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford: Oxford University Press, 1995.

[BVV$^+$17] M. Barbu, R. Vilanova, J. L. Vicario, M.J. Varanda, P. Alves, M. Podpora, M.A. Prada, A.Moran, A. Torrebruno, S. Marin, and R. Tocu. Io1 - literature review and first arquitecture proposal. Technical report, ERASMUS + KA2 / KA203 SPEET Project, June 2017.

[BVV$^+$18] M. Barbu, J. L. Vicario, R. Vilanova, M.A. Prada, A. Moran, M. Domínguez, M.J. Varanda, P. Alves, M. Podpora, A. Paganoni, U. Spagnolini, and A. Torrebruno. Io4 - publication report on engineering students profiles. Technical report, ERASMUS + KA2 / KA203 SPEET Project, in preparation 2018.

[CST00] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, 2000.

[Mac67] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967.