DATA ANALYTICS OF STUDENTS CONTINUOUS ASSESSMENT ACTIVITY DATA

Kevin Calderón, María Serrano, Nicolás Serrano, Carmen Blanco

University of Navarra, TECNUN School of Engineering, San Sebastián, Spain

ABSTRACT

Codex is a web-based tool for the online edition of theoretical and practical teaching content, and for assessment of STEM subjects. It has been developed at TECNUN, the Engineering School of the University of Navarra. The Codex application helps teachers to promote active learning (Standard 8) and continuous assessment (Standard 11) without increasing the teacher's workload. Codex is being implemented in the classrooms, which opens another door for the improvement of the learning experience. Codex stores a significant amount of data from each student, which can be used both by the teacher to adapt his or her teaching method and by the student to see what his or her strengths and weaknesses are and be counseled personally. All of this is supported by up-to-date data. The aim of this project is to apply different Data Analytics and Machine Learning methods to the obtained data in the application from a subject called Digital Technology, in order to obtain a prediction of students' grades and performance at each moment of the course, based on his/her behavior and that of previous years' students. This allows the teachers to know information related to performance of their class, and the students, to see towards what result they are heading.

KEYWORDS

Learning analytics, continuous assessment, automated assessment, Standards: 8, 11

INTRODUCTION

The important progress in information technologies and the growth of their use in different areas of life allows the automated generation and capture of large amounts of data automatically. From the analysis of this enormous volume of data, very valuable information and knowledge can be extracted, future events can be predicted (and therefore prevented) and existing processes and tools can be optimized.

Data analytics benefits in education

In the last decades the use of ICT has also been introduced in education, not only for online education but also as a support to face-to-face teaching. The analysis of educational data can provide new insights about the educational process and allows improving the teaching-learning process and more specifically the performance of students and teachers (Larruson & White,

2014). The areas in which it is applied can be at the level of a student, a class or even one or more institutions.

Research in this area has acquired relevance in recent years, giving rise to Educational Data Mining (EMD) and Learning Analytics (LA) (Romero & Ventura, 2020).

Each of the data analysis methods applied to education helps to obtain different types of information (Bogarín, Cerezo, & Romero, 2018). One of the first applications of data analysis is the ability to predict future outcomes. Applied to the field of education, one can predict the grades or performance of students based on the results of previous years or the same student throughout the course.

These predictions provide information that helps measure the quality of the teaching-learning process. At the same time, different actors in the educational process can take advantage of this information and make changes with the intention of improving learning outcomes. For example, knowing the data from their classes, teachers can modify both the way of teaching and the way of grading their subject. In the case of the students, they can modify their attitude or their study method to face the subject.

Data analytics drawbacks in education

Despite the benefits that LA can bring to the improvement of the teaching-learning process, there are different risks. In a classroom setting, results could be prioritized over student learning, resulting in a surface-learning (Jordan, 2009). Instead, the information shown to the students could contribute to their lack of motivation, and therefore, to their abandonment of interest in the subject. It is the role of teachers to make sure that the use of LA is always for the benefit of the students and to avoid this kind of situations (Arnold & Pistilli, 2012), filtering the information and being aware that some data are not considered in the prediction, such as a student's personality.

Data analytics barriers in education

The difficulties we may encounter when applying data analysis in education focus on two aspects: data collection and data standardization. In order to collect the data, it is necessary to use a tool that stores the students' grades. However, it is also highly recommended that the tool not only stores the grades, but also corrects and evaluates the students, and assigns the grades. This saves the teacher a lot of time. However, the data collected in education is very variable due to several factors. An example is often that students do not take a test, and therefore are not assigned a grade. There are also problems when data from other academic courses are required, since it is normal that the exercises used to evaluate vary every year.

MOTIVATION

Our thesis is that continuous assessment can be considered an enhancer of student learning if, through an adequate and easy-to-use LA tool, teachers and students can obtain valuable information extracted from the data provided by the different assessment actions. We choose LA as the next step in the development of the assessment tool, as a research line with a huge growth in recent years and with great potential to improve education. We emphasize the easy-to-use aspect of the tool, since the existing tools are oriented to experts.

To facilitate continuous assessment, an automated evaluation tool has been developed and is already being used in several subjects of our engineering degrees (cf. Serrano *et al.*, 2018). In another paper submitted to 17th International CDIO Conference (Nicolás Serrano, Blanco, Calderón, Gutiérrez, & Serrano, 2021) the continuous assessment method implemented in a second-year programming subject is described.

The study is the first step in the current development of a Learning Analytics assistant that, fed with the continuous assessment data, facilitates and improves the learning process: providing information on the situation of the class and of each student and proposing recommendations. Once the LA tool is available, we will proceed to research on the impact of its use in improving learning.

MODEL DEVELOPMENT

Our goal is to conduct a study that allow us to know if it is possible to predict the results of students in a subject at our university. This prediction should be based on the data that has been collected on the subject in the past and during the course. We clarify that each teacher could use this method if the teaching topic is evaluated quantitatively. At the organizational level, it is recommended to apply the same evaluation to the whole class. In this case, the organization of the data for each student will be the same, facilitating the development of an LA model.

Data preprocessing

As mentioned in the previous section, data collection is the first difficulty one encounters in data analysis. Therefore, we have decided that the chosen subject is Digital Technology (DT), a subject in which students learn to program in the Java language. The goal of the course is that the students use the knowledge they have acquired during the course to solve real problems. For example, to design and program a simple web application. The reason for choosing this subject is that a system of continuous assessment has been introduced in previous years (Nicolas Serrano et al., 2018). In this system the student had to take tests and exercises every week, which contributes to the fact that the amount of data is significantly higher than in other subjects. The students get points with the tests and exercises, which take part in the final grade among exams and a final project.

In addition, these activities were carried out through Codex, a platform with different online teaching resources. The platform allows for automatic grading of activities and storage of student grades. Another advantage of the platform is that the activities, called items in the platform, usually follow the same structure from year to year as they are copied and modified. As each copied item has the ID from the original one, we are able to link those items in the model. To start with the analysis, the only step required is to import the data from the database to be able to use it in our program.

At this point it should be clarified that the questions and exercises that appear on the platform are called items. And the items are grouped in notebooks, which are the ones presented to the students. Related to the analyzed subject, the platform had in the moment 216 items, grouped into 46 notebooks. There are two kinds of items used in the chosen subject. The first kind of items are tests where the student must choose the correct statement. These items are used to evaluate the theory of the subject. The second kind of items consist of programming problems

and the students must introduce their code. Each student's answer is recorded on the platform, and the data obtained for each answer includes: the answer's ID, the student's ID, the item's ID, the grade obtained, the timestamp of the answer, and the maximum grade that can be obtained. A sample of this data is shown in Table 1. Before using the data, we decided to anonymize the student's ID so the information cannot be related to any real student.

Table 1. Original format of dataset.

	USERID	ITEMID	MARK	ANSWERDATE	MAXGRADE
0	1058277809	4214	0.0	2020-01-24 15:51:53.373	1.0
1	1058277809	4214	0.0	2020-01-29 22:08:48.055	1.0
2	1058277809	4498	1.0	2020-01-30 17:35:54.352	1.0
3	1058277809	4499	0.0	2020-01-24 16:22:35.452	NaN
4	1098327170	4499	0.0	2020-01-24 17:40:56.535	NaN
...	...	...	...	...	...
7416	1260649978	5386	0.0	2020-06-02 17:46:34.055	14.0
7417	1203634691	5383	12.0	2020-06-02 17:05:48.319	12.0
7418	1203634691	5384	0.0	2020-06-02 17:41:02.427	12.0
7419	1203634691	5385	12.0	2020-06-02 17:19:37.827	12.0
7420	1203634691	5386	0.0	2020-06-02 17:37:51.224	14.0

[7421 rows x 5 columns]

To begin with, we need to export the data from the database into our python program as a new dataset. We are getting answers from January to September 2020, as the database may contain information which corresponds to a new course. Also, we are only getting the last students' answer for each item. This last filter is applied because the student can give multiple answers for the same item if the teacher allows it.

In this research we decided to use only data from the grades, as it is considered as the most significant to predict the final grade (Arnold & Pistilli, 2012). Then, the input variables to our model are the item grades and the target variables are the final grades, so our desired structure is the one shown in Table 2. In the table, we have an initial column with the student IDs, several columns with the student grades (one for each item), and a final column with the final grade, computed with the item grades. Then, each row shows us all the required data from each student.

Table 2. Desired format of dataset.

Student ID	Item 1	Item 2	...	Item m	Final grade
Student 1	0.2	0.5	...	1	0.8
...	...	...	...	...	...
Student n	1	0.7	...	0.8	0.75

The next step is the standardization of the grades. As it was mentioned earlier, the variability of educational data forces us to adapt the data before applying any data analytics method. So, we normalize the grades into a 0-1 range, as they have different formats for each item. This helps optimize the algorithm's performance. We do so by dividing each grade by the maximum grade for that item. If it is missing for an item, we find the maximum grade achieved by any student and use it as the maximum grade.

However, there are a few more changes to be done which improve the algorithm's performance. On the one hand, we remove the items that do not give us any information. These are the ones with a 0.0 grading. Either they were items with only a theoretical explanation (without evaluation), or they were asked not to be answered by the professor. On the other hand, we have missing values that are stored as NaN values, which represent the questions with no answer. Thus, we convert all the NaN values to a 0.0 mark. An example of the final result is shown in Table *3*.

Table 3. Final dataset.

	USERID	4214	4499	4498	4217	4500	4215	4218	4252	4219	...	\
0	1001878881	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
1	1008562537	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	0.50	...	
2	1009074790	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
3	1009358503	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	0.75	...	
4	1010498129	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
5	1011762717	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
6	1034515793	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
7	1039454863	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
8	1042775892	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	0.00	...	
9	1059658959	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	0.75	...	
10	1062206845	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
11	1069422119	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
12	1077598052	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
13	1083182240	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
14	1087192798	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.00	...	
15	1098327170	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.00	...	

Now the data has the correct shape, and we can easily visualize a student's grades progress throughout the year. Figure 1 shows the data from a random student. In the figure, the x axis indicates the items ordered in time and the y axis represent the normalized grade. Then, blue crosses represent each item's grade, the red line is the final grade, and the green curve represents an approximation of the accumulated grade or the amount of the final grade obtained with the answered items.

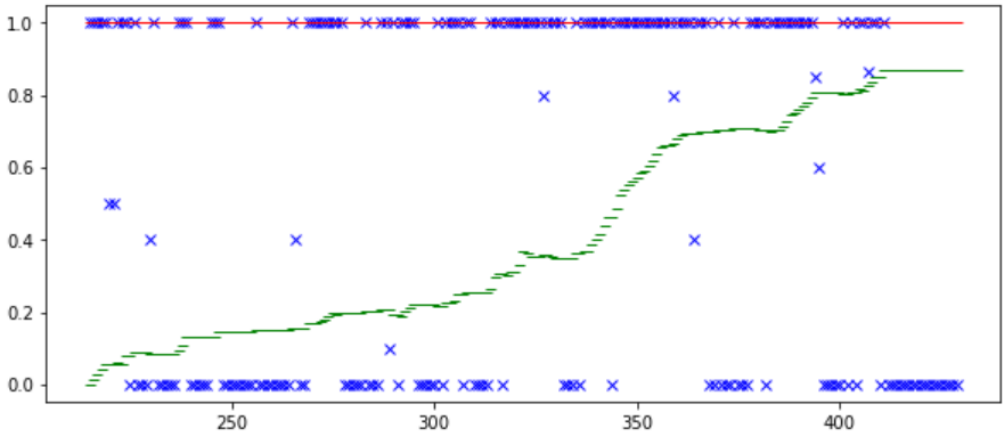

Figure 1. Random student's progress.

Model fitting

The next step is to train the linear regression model with scikit-learn's LinearRegression model (Raschka & Mirjalili, 2017). The multiple linear regression model takes m input variables, x_m $\forall m \in (0,1)$, and tries to find the coefficients, w_m $\forall m \in (0,1)$, to produce the outcome y that best fits the actual solution:

$$y = w_0 x_0 + w_1 x_1 + \cdots + w_m x_m = \sum_{i=0}^{m} w_i x_i = w^T x$$

Then, we need to split our data into a training and test dataset. Using scikit-learn's `train_test_split` method, we define the percentage of the samples that are going to make it to the test dataset. We are going to train the model to predict the final grade based on 150 items completed, out of all the 216 items that compose the course. Once both training and test datasets are created. We get the slope and intercept values of the fitted model by calling the `coef_` and `intercept_` methods from the LinearRegression object. The slope is defined by the coefficients of the model, and the intercept refers to the independent term. As we are working on a high-dimensional feature space, it is not really helpful to visualize the solution.

Using the trained model with linear regression, we predict the final grades and compare them with the ones obtained by students. The results are shown in Figure 2 for a group of students. In the figure, x axis represents the students, and the y axis the grade between 0 and 1. Blue crosses represent the actual grades, whereas red crosses are the predicted ones. Considering a figure similar to Figure *1* for any student, the maximum value of the green curve would be represented in this figure as a red cross.

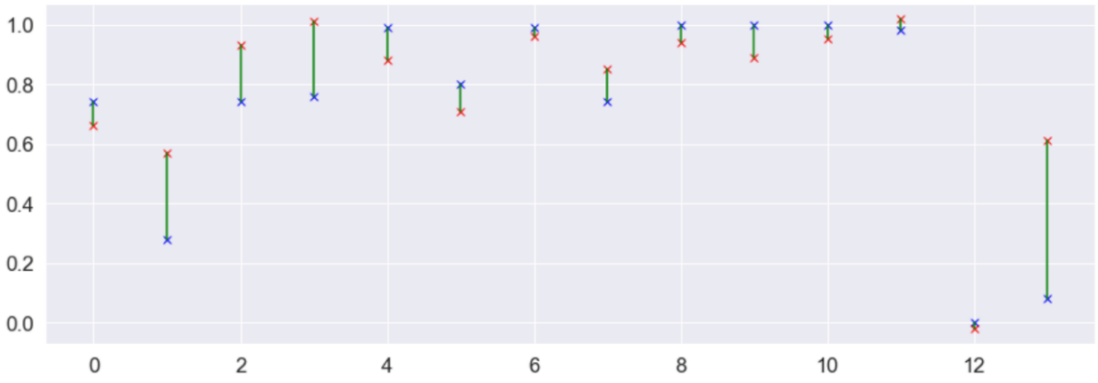

Figure 2. Comparison of actual and predicted final grades.

Figure *2*. Comparison of Figure 2 shows a general good accuracy for most of the students, but it fails to predict correctly two of those students (the second and last one) by a large error. Following this problem, we checked to see if those two students had anything in common between them and whether they differed much from the rest of the students. Comparing the plot from Figure 1 and the ones from both “outliers” progress, shown in Figure 3 and Figure 4, we discovered that both the students with a greater error at the prediction are students who dropped the course before answering 150 items. It is understandable that this situation causes the error from Figure 2, with both students having a much higher predicted grade than the achieved one. This is because the model is only trained with the first 150 items, where the students had a better performance than the one in the second half of the course. As the model does not know that, it predicts a much higher value.

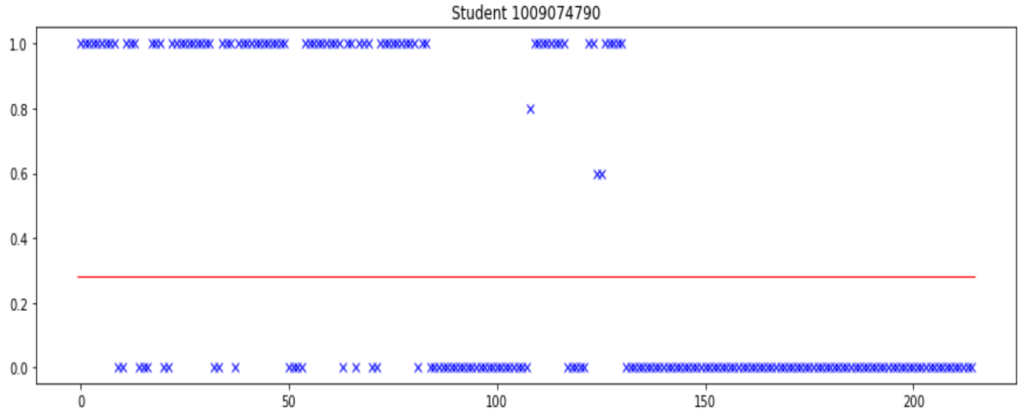

Figure 3. Course progress of the first outlier.

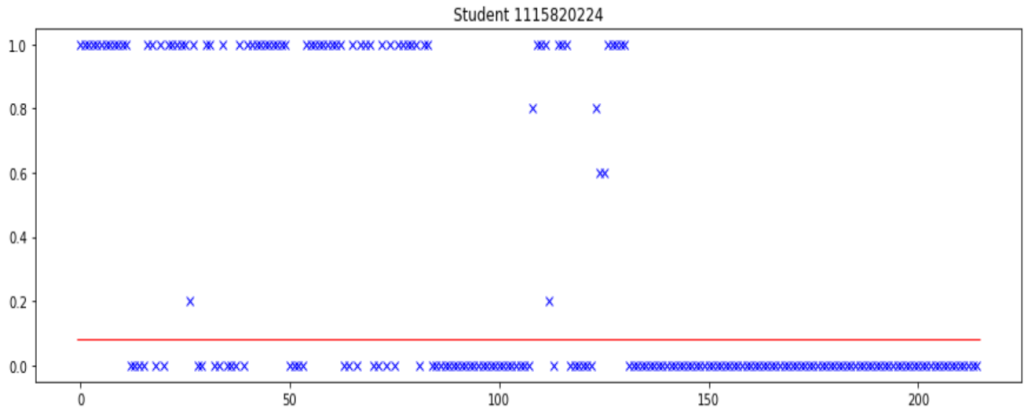

Figure 4. Course progress of the second outlier.

Finding a way to target which students are going to drop is a matter that needs to be assessed properly to prevent it. In addition, it is important as these cases must be leaved out of the training dataset so they do not affect the model's accuracy. A comparison of the accuracy of the model with and without outliers, with a test set of 20% of the items, is shown in Table 4. Despite that the mean squared error is greater without outliers, the R2 score increases significantly, which means that the prediction is more reliable. At the moment, we tackle this problem by manually selecting those students who are outliers and leaving them out of our data frame.

Table 4. Accuracy comparison of the model with and without outliers.

	With outliers	Without outliers
Explained variance	0.44	0.78
Max error (grade pt)	0.27	0.40
Mean absolute error (grade pt)	0.13	0.12
Mean squared error (grade pt^2)	0.02	0.03
Median absolute error (grade pt)	0.11	0.06
R2 score	0.43	0.71

RESULTS AND DISCUSSION

Using the model that has been developed and described in the previous section, we have made predictions by simulating that students are in different moments of the course. This is equivalent to having stored a different number of answered items from students. We have made predictions using 25%, 50% and 75% of the available items. A comparison table is shown in Table 5. The table shows the accuracy scores of the model for the three cases. As Table 5 shows, the scores improve when we use half of the available items compared to only using 25% and it slightly gets worse using 75%. This is a common behavior in data analysis, as the prediction improves as we include more data, which explains the difference between using 25% and 50%. The difference between using 50% and 75% is more difficult to explain. Our

guess is overfitting, as the added information fits the result in the model and it does not have into account possible variations at the end of the course.

Table 5. Accuracy scores of the model for 25%, 50% and 75% of the available items.

	25%	50%	75%
Explained variance	0.56	0.78	0.74
Max error (grade pt)	0.28	0.37	0.40
Mean absolute error (grade pt)	0.16	0.09	0.11
Mean squared error (grade pt^2)	0.03	0.02	0.02
Median absolute error (grade pt)	0.19	0.06	0.05
R2 score	0.55	0.76	0.68

CONCLUSIONS

The results have shown us that it is possible to predict a student's final grade in the subject quite accurately. This phenomenon occurs even if only a quarter of the course has been graded. The data show us that the prediction improves with respect to the progress of the course, since more data is obtained from each student. It has also been found that those cases in which a good prediction has not been achieved are those in which the student has decided to leave the subject. In the future, a way to automatically detect these special cases could be investigated in order to prevent them.

Considering the obtained model, it was discovered that some items have a negative coefficient to compute the prediction. Therefore, they probably do not evaluate correctly the knowledge of the students. For example, a good grade in one of those items means that the predicted final grade for the student will be lower.

These conclusions encourage us to show this information to teachers and students during the next course in the platform. Then, as it is previously mentioned, we will proceed to research on the impact to enhance the model, considering that the results may vary. For example, the students may modify their performance and attitude when they know this information. If this situation occurs, it is possible to calculate if the additional information has contributed to improve the students' performance.

Finally, we pretend to apply the same assessment method in other subjects. Then, it would possible to study if the predictions of the final grades have similar accuracy.

REFERENCES

Arnold, K. E., & Pistilli, M. D. (2012). Course signals at Purdue: Using learning analytics to increase student success. *ACM International Conference Proceeding Series*, (May), 267–270. http://doi.org/10.1145/2330601.2330666

Bogarín, A., Cerezo, R., & Romero, C. (2018). A survey on educational process mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(1). http://doi.org/10.1002/widm.1230
Jordan, S. (2009). *Assessment for learning: pushing the boundaries of computer-based assessment*. *Cumbria* (Vol. 3).
Larruson, J., & White, B. A. (2014). *Learning analytics: From research to practice* (1st ed.). Berlin, Germany: Springer.
Raschka, S., & Mirjalili, V. (2017). *Python Machine Learning* (2nd ed.). Packt.
Romero, C., & Ventura, S. (2020). Educational data mining and learning analytics: An updated survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *10*(3), 1–21. http://doi.org/10.1002/widm.1355
Serrano, N., Blanco, C., Calderón, K., Gutiérrez, I., & Serrano, M. (2021). Continuous assessment with flipped learning and automated assessment. In *Proceedings of 17th International CDIO Conference*. Bangkok, Tailand.
Serrano, N., Blanco, C., Carias, F., & Reina, E. (2018). Information from Automated Evaluation in an Engineering School. http://doi.org/10.4995/head18.2018.8132

BIOGRAPHICAL INFORMATION

Kevin Calderón is a PhD student graduated in Telecommunication Engineering in the Engineering School of the University of Navarra, Spain. His research is focused on the development and application of data analytics techniques in the educational and learning area.

María Serrano is an industrial engineer and has completed the master's degree project at the School of Engineering of the University of Navarra, Spain. His area of interest focuses on information systems, data analysis and machine learning.

Nicolás Serrano is a professor of Languages and Computer Systems at the School of Engineering of the University of Navarra, in San Sebastian. He has worked at the CEIT research center and in industrial and service companies. His experience covers the areas Computer Science, Software Engineering, Digital Technology and Information Systems. He has directed and developed several information systems such as the academic management system of the University of Navarra, an ERP in web platform for business management and applications for technological innovation.

Carmen Blanco is an Associate Professor of Mathematics, Engineering School of the University of Navarra (Tecnun), Spain. Currently, her research is focused on the analysis, development and application of the new technologies and methodologies for the improvement of the learning process.

Corresponding author

Nicolás Serrano
University of Navarra, TECNUN School of Engineering, San Sebastián, Spain
Paseo de Manuel Lardizabal, Nº 13
20.018, Donostia - San Sebastián
Spain
nserrano@tecnun.es