



Effects of Teaching Methodology on the Students' Academic Performance in an Introductory Course of Programming

Patricia Compañ-Rosique^(✉), Rafael Molina-Carmona^(✉),
and Rosana Satorre-Cuerda^(✉)

Cátedra Santander-UA de Transformación Digital, Universidad de Alicante,
Alicante, Spain

{patricia.company,rmolina,rosana.satorre}@ua.es

Abstract. The work of a teacher is dynamic. Year after year it is necessary to adjust the contents and the methodology to the features of the students and the changes in the profession. The authors of this paper are aware of these needs and have been adapting over time a basic programming subject of the degree in Computer Engineering. The objective of this work is to analyse how the different teaching methodologies used in an introductory course to programming during several academic years affect the students' performance. For this purpose, the students' academic performance has been collected (the final grade in the first call of the subject) and they have been confronted with different input variables: methodology used (three methodologies: lecture, flipped learning, hybrid methodology), gender and university access grade. The article shows the results of this analysis and establishes the possible correlations between the variables studied.

Keywords: Programming teaching · Teaching methodologies · Flipped learning · Lecture

1 Introduction

The teaching-learning process is highly complex as many factors influence its success. Basically it involves two agents, teachers and students [1].

From the point of view of teaching, the main protagonist is the teacher while the student acquires all the protagonism considering the facet of learning. In any teaching-learning model, a structured plan must be drawn up that can be used to configure a syllabus, to design teaching materials and to guide teaching in the classroom. Many authors present different classifications of teaching methodologies [2]. Among them, Bruce Joyce and Marsha Weil [3] argue that there is no perfect or unique methodology, as there is no model capable of coping with all learning styles. It is evident that each person has a different learning

style according to his or her individual abilities. This means that the teacher must be aware of these differences and establish different teaching strategies in order to reach each student [4]. Even so, within a course the same activity is seen differently by each student.

These characteristics oblige the teacher, and in our case the university teacher, to try to improve not only their knowledge on the subject, but also the way of teaching it. Year by year, they research and put into practice different methodologies and techniques so that their students can better assimilate the knowledge, competences and skills corresponding to the subject [5]. This motivation on the part of the teacher is important because as time goes by it improves the way in which the subject is taught.

This continuous change in teaching methodologies generally leads to greater success in student learning. Although this is the case in most situations, it is important to analyse the academic results of students year after year. In recent times, lecture have been widely denigrated in favour of more active methodologies such as project-based learning, gamification or flipped learning [6].

Computer programming is a basic subject in any computer curriculum. In the specific case of an introductory course to programming, the teacher asks many questions about how to teach it. Many doubts arise about which programming language to use, which paradigm, and so on. The teaching methodology to be used is no exception: lecture? flipped learning? project-based learning? Regardless of the methodology used, it is essential to analyse the results obtained with it. Sometimes, it is thought that more modern techniques are better than more traditional procedures. This is not always so, it is important to review the results obtained to see if effectively using new methodologies makes students learn better.

This paper analyses different teaching methodologies that have been used from the academic year 2011-12 to the academic year 2017-18 to determine if there is a correlation with the academic results obtained. Although the main objective of this work is to determine whether the methodology influences the results, other possible influencing factors have also been analysed, such as the university access grade or the student's gender.

The document is organised as follows. Section 2 presents the context of the research. The teaching methodologies are presented in Sect. 3. Section 4 is devoted to explain how each methodology has been applied. The studies carried out to determine the possible relationships between the variables analysed (methodology, gender, access grade) and the results obtained (subject grade) are shown in Sect. 5. Finally, the conclusions resulting from this work, as well as the possible lines of future work, are presented in Sect. 6.

2 Context

The subject of this research is an introductory subject to programming. It is taught in the first year of the Bachelor's degree in Computer Engineering. It is a degree with a large volume of students, so there is a wide variety of data, about

a thousand samples. With these data, a study is developed in which the results obtained in the subject are analysed according to the access grade, the gender and the main focus of this study, the methodology used.

This subject is the first contact of many students with computer programming. Some of the students, a minority, have notions of this subject because they have taken a programming course before starting university studies. However, for most students this subject is the first encounter with this field. This study will focus on the latter, that is to say, on those students who come from secondary school and who, therefore, have had little or no contact with the subject analysed in this work.

It is an introductory subject with the following objectives:

- Ability to solve problems with initiative, decision making, autonomy and creativity.
- Analyze problems that can be solved by computer and design algorithms to solve them.
- Implement algorithms using structured programming techniques.
- Understand and know how to use a high level language.
- Understand the implications of the work of a programmer, individually and as a member of a team.

The content of the course includes data types, control sentences (conditionals and loops), modular programming, recursion, arrays, records and basic notions of how to measure the computational cost of an algorithm. The programming language used is C. Development environments are not used because the subject teachers are interested in students learning to differentiate basic concepts such as source code, executable code, compiler, and so on. A Linux terminal is used, using the commands to call the compiler and to run the program. Although the use of an integrated development environment would be more comfortable for them, from the point of view of the teachers, the student learns better the basic concepts using this system.

The essential objective of the course is that students learn to solve problems and are able to propose an efficient solution by means of a high level language.

3 Teaching Methodologies

There are multiple teaching methodologies that can be put into practice in the classroom: flipped classroom [7], micro flip teaching [8], project-based learning [9], cooperative learning [10], gamification [11], problem-based learning [12], design thinking [13], thinking-based learning [14], and so on. Normally, the teacher chooses the methodology that he or she believes will work best with the students. This would be the ideal case but it is not always possible and does not always work as expected. The teacher is subject to a number of constraints that also determine the methodology that can be used:

- Temporary restrictions: the time dedicated to the subject, as well as the moment in which it is taught, cannot be modified.

- Quantitative restrictions: the number of students per classroom limits in many cases the methodology to be used.
- Spatial constraints: for instance, the configuration and location of the tables in the classroom often does not facilitate the use of some methodologies.

For example, more active and participatory methodologies on the part of students often require small groups. There are also methodologies that, by their nature, require the student to have a significant amount of time to research and usually it is necessary to adhere to an academic calendar that severely restricts deadlines.

In the subject Programming 1, since the academic year 2011-12, two well-differentiated teaching methodologies have been used and a combination of both. The main characteristics of these methodologies are listed below.

3.1 Lecture

This method could be defined as the set of theoretical or practical knowledge given to apprentices by the master of a science, art or craft [1].

The lecture is the traditional method used in teaching. Although it has drawbacks [15] (it favours passivity in the student, reduces information sources to the teacher's word etc.), it also offers the advantage that it allows the quick transmission of a large volume of knowledge. In addition, it facilitates the comprehension of complex knowledge, since the teacher is in charge of simplifying the concepts to adapt them to the students' level of knowledge. Part of the teacher's job in preparing the lectures is to document himself or herself in an appropriate manner by consulting various authors on the subject. Therefore, although the student does not consult other sources of information, the teacher will have done so and the lessons should show this diversity in knowledge. Another important advantage of the lecture is that the students usually feel more secure in the knowledge conveyed by the teacher than in their own research on the subject.

A good lecture consists of four phases:

1. Preparation and design: objectives formulation, contents organization, activities preparation for students. This phase would be prior to classroom action.
2. Introduction: capturing the audience's attention, establishing relationships with the group, arousing interest, motivating towards the task, objectives presentation, introductory general summary, and so forth.
3. Body: content structuring, maintenance of attention and interest, adequate speed and rhythm, expressiveness, etc.
4. Conclusion: intensified retention, emphasis on main ideas, questions, summary, etc.

3.2 Flipped Learning

Flipped learning is a methodology widely used today [16,17]. With this type of technique, students are provided with a series of materials to prepare the course.

Normally, face-to-face sessions are left to solve doubts or do exercises. Before the face-to-face session, the student must have worked on the subject in order to be able to ask the teacher any doubts that may have arisen. It is also common practice to establish some milestones (report delivery or exercises) during the teaching period of the subject. These methodologies rely on techniques that actively involve students and give them a leading role in their own learning.

By means of flipped learning techniques, the student can establish his own work rhythm. In the specific case of a subject belonging to a university degree, as it is the subject we are talking about, this is so to some extent. The student is provided from the beginning with all the teaching materials (both audiovisual and written). It is obvious that the student can establish a learning rhythm higher than the one initially set by the teachers, but it is also true that the student has to respect a minimum speed set by the different deliverable activities, each of which has a deadline.

4 Practical Implementation

The authors of this work have been teaching programming for more than twenty years. This section explains in a concrete manner the way in which the subject has been taught from the academic year 2011-12 to the academic year 2017-18. Different methodologies and approaches have been followed in each academic year. All this with the very complicated aim of ensuring that students acquire the knowledge, skills and abilities involved in computer programming. Different techniques have been used in both theory and practice over the years, but the main changes have occurred in theory. The practical classes have always been taught in a similar way: the students are dedicated to doing programming exercises, so that from a given problem, they find an efficient solution in which they use the appropriate structures, asking the practice teacher when they have doubts. From one course to another there are some variations in the way of evaluating the practice. There are always two practice exams, but in addition, some years some additional control has been incorporated. In addition to this, there have been courses in which a competition of practices has been organised to foster the motivation of the students. Table 1 shows in a schematic way the methodology used in theory and practice classes, as well as other data related to the way of teaching the subject.

4.1 Lecture

Computer programming is an eminently practical subject. In addition, it is characterized by a double aspect: it presents a rigorously technical character with methods characteristic of an engineering as well as requiring an important creative component, as Andy Hertzfeld, co-designer of Macintosh, comments in *Programmers at Work* [18]. These characteristics of programming make the lecture, in its strictest sense, not the most appropriate methodology for this discipline.

Table 1. Methodology used each academic year

Year	Theory	Practice	Competition
2011/12	Lecture	-5 paper controls -2 practice exams	No
2012/13	Lecture	-3 paper controls -2 practice exams	No
2013/14	Lecture	-peer reviewed paper control - access to a practice requires that a particular exercise of the previous practice has been delivered -2 practice exams	No
2014/15	Lecture	-peer reviewed paper control -2 practice exams	Yes
2015/16	Lecture	-peer reviewed paper control -2 practice exams	Yes
2016/17	Flipped learning	-2 practice exams	No
2017/18	Hybrid methodology	-2 practice exams	No

It is necessary to incorporate multiple examples during the exposition, as well as to intercalate practical activities to be developed by the students.

From year 2011/12 to year 2015/16 the lecture with examples and practical exercises has been the methodology used for the theoretical sessions of the subject. It is important to point out that although all teachers share the same teaching material, the characteristics of the specific group of students to whom the session is addressed, as well as the character of the teacher, greatly influence the way the session is given. Working with a small group of 20 students is not the same as working with a group of 100 students. Similarly, the attitude of the students, which has a great influence on the final result, may also be very conditioned by their profile: age, access studies, family situation, etc.

Although the lecture usually requires little or no student participation, this is not really how it develops in our classes. During the session, concepts are explained and student participation is encouraged through exercises, comparisons with real situations, comments on habits and even exposition by students of their own proposed solutions.

4.2 Flipped Learning

A class based on flipped learning requires prior involvement of the students, which seems to be ideal for this subject. For this reason, in the academic year 2016/17 the teaching staff of the subject decided to implement this methodology for the theory sessions. At the beginning of the course all materials were provided, as well as the planning of deliverables. Approximately every week, the students had to deliver some basic programming exercises corresponding to the theoretical concepts on which they were going to work during the face-to-face session. It is important to point out that the students had to review the materials and send the deliverables before the face-to-face session, that is, without the teacher having explained anything about this topic. The first part of the session was dedicated to solving doubts, as well as to commenting on the main errors detected in the deliverables sent. This meant that the teacher had to review the deliverables before the face-to-face session in order to be able to address errors in the exercises. The second part of the session was devoted to solving more difficult exercises. Specifically, 12 deliverables were planned for the period between September and December.

An important part of the teacher's task during the correction was based on providing feedback to the student. This task was very laborious, given the volume of students enrolled in the subject, but very useful in motivating the delivery of the exercises.

4.3 Hybrid Methodology

The implementation of flipped learning during the 2016/17 academic year presented some drawbacks. The first topics corresponding to simple concepts such as conditionals or loops did not present much difficulty. The students managed quite well with the supplied materials and were able to solve the deliverables without major problems. The complications started when they started working with the modular programming materials. In spite of the fact that the materials had a special emphasis on the parameter passing, as well as on the decomposition into modules, the students did not assimilate it correctly and had difficulties in making the deliverables. It is also true that in previous courses, in which lecture were used, this topic had always generated many complications.

The other major problem that arose was due to the enormous amount of exercises that the teacher had to review before each face-to-face session in order to be able to address the main faults detected.

Given the volume of students each teacher had, an excessive amount of time had to be devoted to reviewing deliverables properly and providing students with feedback on their work. As mentioned above, individualised feedback is essential for the students to learn from their mistakes and be able to progressively improve.

The main problem, though, arose when students, lacking the habit in this type of methodology, considered that it was essential to provide correct exercises and frequently resorted to copying them from other classmates or other sources. They had been strongly insisted that the important issue was to perform them,

even if they were not functioning well, since the aim of this type of task is to think, to analyse the problem and thus see whether or not one is capable of providing a solution. However, if the student did not make his own developments, the main advantages of this methodology were completely lost. Actually, the problem of applying this methodology is not the methodology itself, but the fact that since they have started school they have been taught to memorise and then release what has been memorised. Rarely have they been taught to think [19].

Analysing the problems derived from the use of this methodology, it was decided to implement in the academic year 2017/18 a hybrid methodology in which for the initial topics (conditionals and loops) the flipped learning scheme was followed. Two deliverables corresponding to these two topics were programmed. For the rest of the topics of the subject, a lecture scheme was followed, complemented with 4 deliverables corresponding to modular programming, recursion, arrays and records. In addition, the students were provided with an automatic corrector that they could apply to the modular programming and recursion deliverables. If they used the template supplied for these deliverables they could run their programs with different test cases and the corrector indicated whether or not there were operating errors.

Both teachers and students have perceived that this hybrid methodology provides more satisfactory results. However, the problem of copying was still present. This is an issue for which no solution has been found, since the aim of the deliverables is for students to work and make mistakes, but they prefer to copy an exercise that works from another colleague.

5 Results

This section shows a detailed analysis of the results obtained by students from the academic year 2011-12 to 2017-18. Although each course has followed different techniques to teach the subject, it could be said that basically three techniques have been used:

- Lecture: from year 2011-12 to 2015-16. Data are available for 888 students.
- Flipped learning: year 2016-17. Data are available for 148 students.
- Hybrid methodology: year 2017-18. Data are available for 141 students.

The technique followed to find out if there is a relationship of dependence between two variables has been the technique ANOVA (ANalysis Of VAriance) [20, 21]. Following the terminology of this technique, the independent variable will be called factor and the dependent variable will be denoted as response. In this work, a one-way ANOVA will be applied since only one factor is analysed for each case presented. The application of the ANOVA technique is based on a hypothesis contrast. The null hypothesis that is contrasted is that the population means are equal:

H_0 : means are equal

H_1 : means are not equal

If the null hypothesis is accepted, it means that the population groups do not differ in the mean value of the dependent variable and therefore the mean value can be considered independent of the factor. In order to contrast the null hypothesis, the value of Snedecor’s distribution F is calculated and if it is greater than the critical value, the null hypothesis is rejected [22].

Given that there are three methodologies to be analysed, the data have been grouped into three groups: one for the years in which lectures have been used, another for the year in which flipped learning was used and another for the hybrid methodology.

The objective of the study is to detect whether the teaching methodology is related to the results obtained in the subject. In addition, it also aims to analyse other factors of possible influence such as the student’s gender and the university access grade. In order to be able to apply the ANOVA technique, the groups under study must follow a normal distribution. Figure 1 shows the distributions of the grades for the three teaching methodologies and for the gender of the student. As can be seen, the distributions of population groups can be assumed to be normal.

Each academic year there are two calls to pass the subject, so that students who fail the first call have the opportunity to pass in the second. In order to make the contrast in the most similar conditions between years, only the first call of each academic year analysed has been considered. In addition, repeat students have not been considered, since they could affect the result since the second year they take the subject they already have knowledge that the rest of the students do not have.

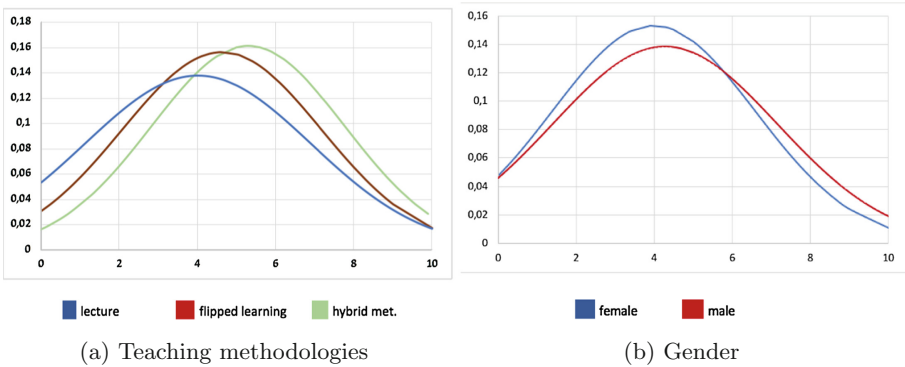


Fig. 1. Grade distributions

5.1 Teaching Methodology - Final Grades

This section shows the study carried out between the teaching methodology used and the final grade obtained by the student.

Applying the ANOVA technique gives a value $F = 14.6742124$, the critical value is 3.0033766 . Since the value F is greater than the critical value, the null

hypothesis is rejected and it is concluded that there is a relationship between the teaching methodology used each academic year and the grades obtained by the students. Figure 2 shows a diagram representing the final grade for the three methodologies analysed. The line joins the average grade obtained for each methodology. It can be seen that the average grade has increased slightly with the hybrid methodology.

These results confirm the initial impression of the teachers of the subject that the use of a combined methodology presents the advantages of the two methodologies used and at the same time compensates for their shortcomings. This subject is developed during the first four-month period of the first year. The students have just begun their university life, are a little disoriented and do not know how the University works. A pure flipped learning methodology is too radical for them, while the lecture makes them maintain a too passive attitude. After trying different strategies, it seems that what works best is to combine both techniques to take advantage of the lecture for complicated concepts while simpler concepts can be assimilated with a more student-dependent methodology. In addition, the fact of introducing some content through flipped learning prepares the students so that in later courses they can introduce this methodology in a more complete way.

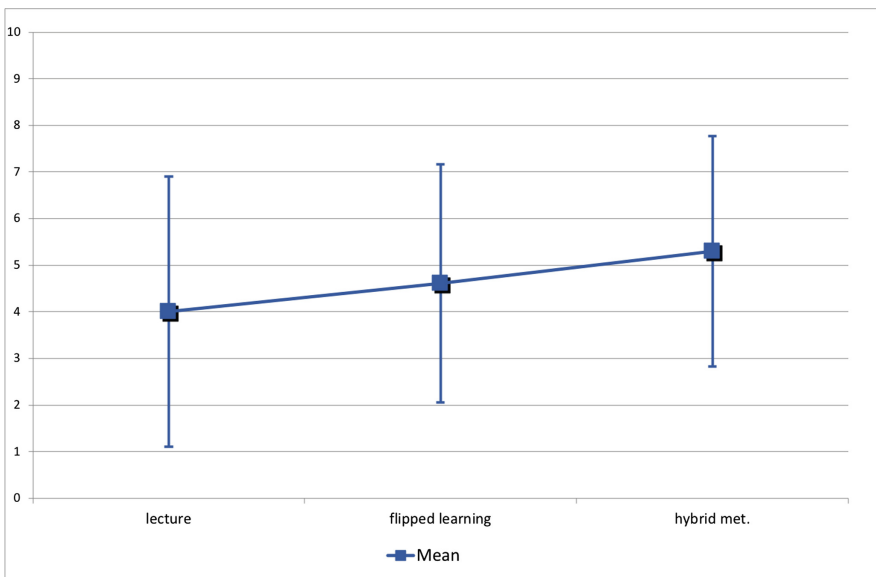


Fig. 2. Mean and standard deviation: Teaching methodology - Final grades

5.2 Gender - Final Grades

In this case, the study carried out between the student’s gender and the final grade obtained is shown. Data are available for 162 women and 1015 men. Computer Engineering is a degree whose students are mostly men, due to this fact there is so much disparity in the number of samples.

When applying the ANOVA technique, a value of F 1.43 has been obtained, the critical value being 3.84. Given that the value of F is lower, the null hypothesis is accepted, which means that the population groups do not differ in the mean value of the dependent variable and therefore this mean value can be considered independent of the factor.

Figure 3 shows a diagram representing the mean and standard deviation obtained in the subject grade for the two population groups: men and women. There are no significant differences between the two groups.

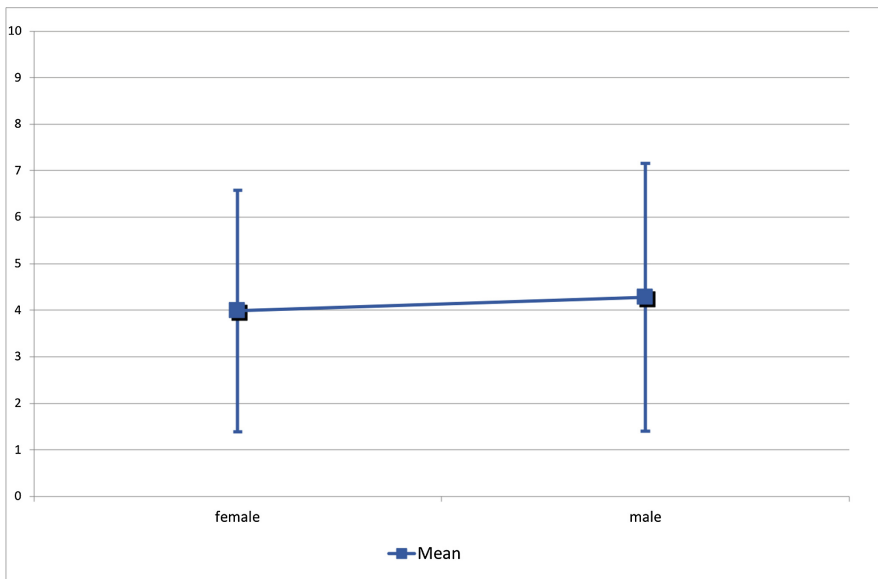


Fig. 3. Mean and standard deviation: Gender - Final grades

5.3 University Access Grade - Final Grades

This section shows the study that has been carried out between the student’s university access grade and the final grade obtained by the student. The number of samples available is 1177.

In this case, since both variables are quantitative, a linear correlation analysis has been performed to detect possible relationships between the variables [23].

There are several coefficients that allow quantifying the degree of linear relationship between two variables. In this case the correlation coefficient of Pearson [24] has been used. The value obtained was 0.5. This is a middle value that indicates that there is a moderate linear relationship.

Figure 4 shows a diagram made for the two variables object of this analysis: university access grade and final grade obtained in the subject. Students who have a very high access grade (12–13 points), obtain high grades in the subject. On the other hand, students with low access grades (5–7) obtain low average results in the subject. In the latter cases, the variance has high values. This may be due to the fact that there are students who have a low access grade because the access grade measures the average of the subjects that the students have taken in their pre-university stage, where there are subjects that they do not like or they are not good at. When the students begin their university stage, they are taking a discipline closer to their preferences.

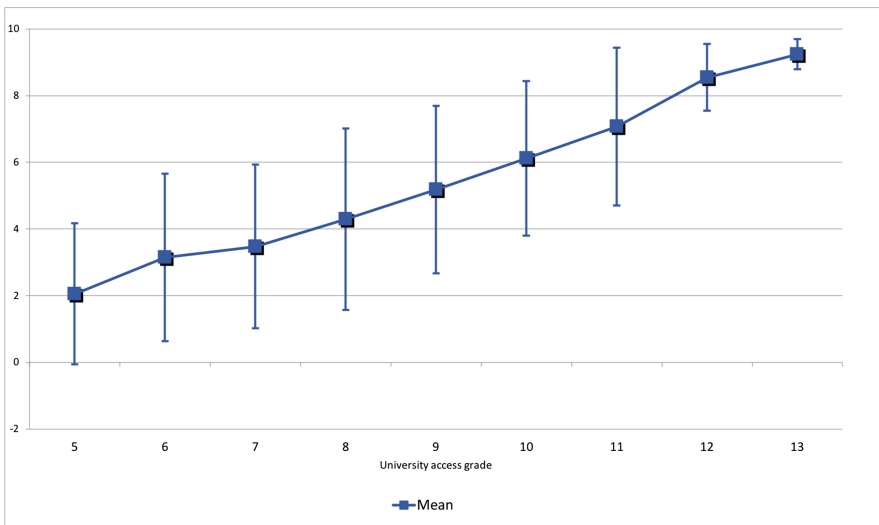


Fig. 4. Mean and standard deviation: University access grades - Final grades

6 Conclusions and Further Work

This study has served to verify possible factors of influence on the final grades obtained by the students in a course of introduction to programming.

In order to carry out the study, the ANOVA technique was used, based on a hypothesis contrast. After compiling the data of the students of the degree in Computer Engineering from the academic year 2011-12 to the academic year 2017-18, different analyses have been carried out to determine the relationship

between the final grade obtained by the students and variables such as the teaching methodology used, the gender of the students and the university access grade.

The first study has shown the results of the ANOVA technique between the teaching methodology and the final grade. Analysing the results obtained, it is detected that the use of a hybrid methodology (lecture and flipped learning) seems to be more convenient for the subject under consideration.

The second study analysed the relationship between the gender of the students and the final grade. In this case, there is a great difference in the number of samples between both populations, since it is a degree in which the majority of students are men. However, the analysis carried out allow us to state that there are no significant differences in the subject grade between both groups.

The last case shown analyses the relationship between the university access grade and the final grade. For this last study, a linear correlation analysis has been carried out since both variables are quantitative. The correlation coefficient obtained indicates that there is a moderate linear relationship between both variables.

This work is considered the seed for future research where other possible influencing factors will be analysed. In addition, it is also intended to perform factorial variance analysis with ANOVA models where the influence of two or more factors on the dependent variable is studied. In these experiments the interaction of several factors can be considered.

References

1. Compañ-Rosique, P., Satorre-Cuerda, R., Llorens-Largo, F., Molina-Carmona, R.: Enseñando a programar: un camino directo para desarrollar el pensamiento computacional. *Revista de Educación a Distancia* (46), October 2015
2. Anderson, L.W., Krathwohl, D.R. (eds.): *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*, Complete edn. Longman, New York (2001)
3. Joyce, B.R.: *Models of Teaching*, 9th edn. Pearson, Boston (2015)
4. *Journal of Learning: Journal of Learning Styles* (2016)
5. Castaño, A., Marquós, M., Satorre Cuerda, R., Jaume i Capó, A., López Álvarez, D.: Tengo una respuesta para usted sobre estilos de aprendizaje, creencias y cambios en los estudiantes, Universidade de Santiago de Compostela. *Escola Técnica Superior d'Enxerñaria*, pp. 275–282, July 2010
6. Knight, K.: *Book Reviews: The Manual of Learning Styles* Peter Honey and Alan Mumford, 83 p. Peter Honey, Maidenhead (1982). £25.20, ISBN 0 9508444 0 3. *Manag. Educ. Dev.* **14**(2), 147–150 (1983)
7. Bloom, B.S., Hastings, J.T., Madaus, G.F.: *Handbook on Formative and Summative Evaluation of Student Learning*. McGraw-Hill, New York (1971)
8. Fidalgo-Blanco, Á., Sein-Echaluce, M.L., García-Peñalvo, F.J.: Micro flip teaching with collective intelligence. In: Zaphiris, P., Ioannou, A. (eds.) *LCT 2018. LNCS*, vol. 10924, pp. 400–415. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91743-6_30
9. de Los Ríos, I., Cazorla, A., Díaz-Puente, J.M., Yagüe, J.L.: Project-based learning in engineering higher education: two decades of teaching competences in real environments. *Procedia - Soc. Behav. Sci.* **2**(2), 1368–1378 (2010)

10. Spencer, K.: Kagan Cooperative Learning. Kagan Publishing, San Clemente (2009)
11. Llorens-Largo, F., Villagr -Arnedo, C., Gallego-Dur n, F., Satorre-Cuerda, R., Compa n-Rosique, P., Molina-Carmona, R.: LudifyME. In: Formative Assessment, Learning Data Analytics and Gamification, pp. 245–269. Elsevier (2016)
12. Delisle, R.: How to Use Problem-based Learning in the Classroom. ASCD, Alexandria (1997)
13. Naiman, L.: Design thinking as a strategy for innovation
14. Yusuf, M.: Infusing thinking-based learning in twenty-first-century classroom: the role of training programme to enhance teachers' skilful thinking skills. In: Amzat, I.H., Valdez, N.P. (eds.) Teacher Empowerment Toward Professional Development and Practices, pp. 211–220. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-4151-8_14
15. Fidalgo, A.: Metodolog as. Lecci n Magistral: Qu  es y c mo mejorarla (2016)
16. Lage, M.J., Platt, G.J., Treglia, M.: Inverting the classroom: a gateway to creating an inclusive learning environment. *J. Econ. Educ.* **31**(1), 30–43 (2000)
17. Fidalgo-Blanco, A., Sein-Echaluce, M.L., Garc a-Pe alvo, F.J.: APFT: active peer-based flip teaching. In: Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM 2017, C diz, Spain, pp. 1–7. ACM Press (2017)
18. Lammers, S.M.: Programmers at Work: Interviews with 19 Programmers who Shaped the Computer Industry. Tempus Books of Microsoft Press, Redmond (1989)
19. Swartz, R.: ENSE NAR A PENSAR: 9 PRINCIPIOS BASICOS-IX - INED21
20. Sthle, L., Wold, S.: Analysis of variance (ANOVA). *Chemom. Intell. Lab. Syst.* **6**(4), 259–272 (1989)
21. Smalheiser, N.R.: ANOVA. In: Data Literacy, pp. 149–155. Elsevier (2017)
22. Yoo, S.K., Cotton, S.L., Sofotasios, P.C., Matthaiou, M., Valkama, M., Karagianidis, G.K.: The Fisher-Snedecor \mathcal{F} distribution: a simple and accurate composite fading model. *IEEE Commun. Lett.* **21**(7), 1661–1664 (2017)
23. Taylor, R.: Interpretation of the correlation coefficient: a basic review. *J. Diagn. Med. Sonogr.* **6**(1), 35–39 (1990)
24. Sedgwick, P.: Pearson's correlation coefficient. *BMJ* **345**, e4483–e4483 (2012)