Journal of Software Engineering Research and Development, 2021, 10:13, doi: 10.5753/jserd.2022.1977

OSS in Software Engineering Education: Mapping Characteristics of Brazilian Instructors

Fernanda Gomes Silva [**Federal University of Bahia** | ***nanda.gomes.si@gmail.com***]
Paulo Ezequiel D. Santos [**Tiradentes University** | ***paulo.ezequiel@souunit.com.br***]
Christina von Flach [**Federal University of Bahia** | ***flach@ufba.br***]

Abstract. Software Engineering is a crucial topic in undergraduate computing-related courses and provides the basic knowledge and skills necessary for professional practice in the software industry. Teaching Software Engineering principles, concepts, and practices and relating them to real-world scenarios are challenging tasks, and the adoption of Open Source Software (OSS) projects can help to face these challenges. On the other hand, adopting OSS projects as a didactic resource may introduce additional challenges to instructors who are not familiar with the OSS ecosystem. **Objective:** In this paper, we identified and mapped the profiles of instructors of Software Engineering courses concerning their classroom practices and use of OSS projects in Software Engineering Education. **Method:** We surveyed 90 higher education instructors in Brazil to collect data regarding their familiarity with the Software Engineering knowledge areas, pedagogical methods and resources used, and familiarity with and use of OSS projects in the classroom. Then, we resorted to data mining techniques, for instance, K-modes and Decision Tree algorithms, to identify instructors' characteristics according to their classroom practices and use of OSS projects in the course activities. **Results:** Our findings include the characterization of instructors who use and instructors that do not use OSS projects in Software Engineering Education and the grouping of instructors after the application of the K-modes algorithm, and after the application of the Decision Tree algorithm, with similar characteristics of the pedagogical practices. The main result of this work is that the familiarity with OSS projects and the use of active learning methods were characteristics present in the application of the K-modes and Decision Tree algorithms, that distinguished instructors who used OSS projects from those that did not use them in Software Engineering Education. Finally, we confirmed that familiarity with OSS projects could have a positive influence on the instructors' interest and potential for adopting this approach in Software Engineering Education.

Keywords: *Software Engineering Education, Open Source Software, Classroom Practices, Instructor Characteristics, Survey Study, Data Mining Techniques.*

1 Introduction

Proper education and training can significantly improve the Software Engineering (SE) practice and are considered prerequisites for advancing the state-of-the-art in the software industry (Leite and Werner, 2008). Nevertheless, there are several challenges to be addressed in Software Engineering Education, such as the need to convey the practical experience in the context of a rich and large body of theoretical knowledge (Nascimento et al., 2019). To support the alignment of theory and practice, SE instructors[1] should also consider skills and attitudes that go beyond concepts, methods, and techniques. They should consider the current software development scenario, the complexity of social interactions, and how collaborative software development occurs in a real-world environment (Marques et al., 2020; Gutica, 2018; Nascimento et al., 2018).

In Software Engineering Education (SEE), academic-industry collaborations can provide real-world environments, but instructors may depend on contracts and have to handle issues related to confidentiality of the proprietary code (Ferreira et al., 2018). On the other hand, Open Source Software (OSS) projects allow users to run, study, modify and redistribute the software with few eventual restrictions. The access to its community, the software's source code, and information about its development and evolution are appealing factors for the use of OSS projects in SEE (Nascimento et al., 2013; Smith et al., 2014; Pinto et al., 2017; Nascimento et al., 2018, 2019).

Using OSS projects in SEE brings new challenges for instructors, even the most experienced ones. For instance, the lack of familiarity with OSS projects may hinder the planning and execution of courses that use these resources and negatively influence their adoption in the classroom (Silva et al., 2020). However, the lack of knowledge about instructors teaching SE in higher education courses in Brazil and their use (or willingness to use) OSS projects in the classroom makes this problem even more challenging. Therefore, we consider the characterization of SE instructors who teach SE and their teaching-learning practices, with or without OSS projects. Such characterization is related to the general research question that guides this work:

> *What is the profile of the instructor who teaches Software Engineering in Brazil concerning their classroom practices and the use of OSS projects in the course activities?*

The instructor profile represents a group of instructors with similar characteristics such as familiarity with the SE body of knowledge, pedagogical methods and resources used in the classroom, familiarity and experience with OSS, and familiarity and experience with using OSS projects in SEE.

[1] In this paper, the term *instructor* denotes "teacher in a higher education course," and we will use it throughout the text.

We applied an online survey with instructors who teach Software Engineering in undergraduate courses in Brazil to collect evidence to support the identification of instructor profiles. Based on the responses of 90 participants, we carried out the characterization and classification of instructors with the support of data mining techniques to identify patterns, connections, and correlations in their answers. From the clustering of instructors, according to the similarity of the answers and the identification of characteristics and practices used, we identified and mapped the profiles of instructors who teach Software Engineering in undergraduate courses in Brazil.

Our work contributes to the state-of-the-practice of Software Engineering Education in Brazil: (i) by bringing information about instructors who teach the subject of Higher Education in the country, and (ii) by identifying instructor profiles concerning their knowledge and pedagogical use of OSS projects in their classes.

The instructor profile can be the basis for developing strategies to guide the training of instructors with little or no familiarity or who have not yet used OSS projects in their teaching activities. These contributions may mitigate the difficulties encountered and expand the adoption of OSS projects in SEE.

Section 2 presents the theoretical background on OSS projects and a summary of data mining techniques and the algorithms used in this work, while Section 3 presents the research methods used in to answer our research question. Section 4 describes and analyzes the results, and Section 5 discusses the findings, limitations, and threats to validity associated with our work. In Section 6 we present related work and compare it with our contributions. Section 7 presents the final remarks and paths for future work.

2 Theoretical Foundation

2.1 OSS Projects

Open source software promotes user freedom without discrimination about users and their uses, developed in the context of an open and collaborative process, which may involve from a few sporadic collaborators to various institutions with different interests and organizations (Wen et al., 2020). The Open Source Software Development (OSSD) workflow is supported by open repositories hosted by version control systems that allow multiple versions of the same software to coexist (Flach and Kon, 2021). Educators and students may use the OSS project's open repositories and access different assets (and their versions), such as source code, tests, reports, and workflows. While a repository is often publicly available for reading access, writing access is restricted to a limited group of developers selected by meritocracy. Collaboration among peers is frequent and enabled via shared access to source code and multiple communication channels. A fundamental practice, code review, fosters enhanced software quality through sharing, collaboration, and peer review and can be applied to other research assets. Automated testing increases reliability and maintainability and promotes agility in developing new features. Frequent and continuous documentation keeps user guides, manuals, and other relevant documents updated for the latest version of the software.

Thus, using OSS projects in Software Engineering is appealing to instructors and students, given the openness and availability of the software's source code, the development workflow and use of best practices, and often seamless support to first-time contributors. Furthermore, students are encouraged to participate and contribute/give back to the project community after the course term ends.

2.2 Data Mining

Data mining is an essential step in the knowledge discovery (Han et al., 2012). It comprises a set of methods, techniques, and tools for the effective and efficient analysis of data and extraction of data patterns (Yin et al., 2020; Han et al., 2012) that may represent knowledge. *Data mining techniques* perform a specific task linked to a goal. *Discovery-oriented goals* aim to find previously unknown phenomena in the data through description and prediction (Banimustafa and Hardy, 2020).

Data mining functionalities include characterization and discrimination, mining of frequent patterns, associations and correlations, classification and regression, clustering analysis, and outlier analysis. Data mining functionalities specify the patterns found in descriptive and predictive data mining tasks (Han et al., 2012).

2.2.1 Classification

Classification is the process of finding a model that describes and distinguishes data classes or concepts. The resulting model is based on the analysis of a set of training data (class-labeled data) and expressed in various forms, such as classification rules, neural networks, and decision trees (Han et al., 2012).

A decision tree is a flowchart-like tree structure, where each node denotes a test on an attribute value, each branch represents an outcome of the test, and the tree leaves represent classes or class distributions. We may use decision trees to classify tuples whose class labels are unknown. In this case, we test the attribute values against the decision tree, from its root to a leaf representing the class prediction Han et al. (2012).

2.2.2 Clustering analysis

Clustering is the process of grouping a set of data objects into relevant groups called *clusters* while maximizing the similarity within the cluster (*intracluster similarity*) and minimizing the similarity between clusters (*intercluster similarity*). Dissimilarities and similarities are assessed based on the attributes' values describing the objects and often involve distance measures (Park and Jung, 2020). Clustering can also support the generation of class labels for unknown data.

Clustering analysis handles data objects without consulting class labels; that is, it can deal with data that is not class-labeled. Partitioning is the simplest yet fundamental version of clustering analysis. The *K-means* is a popular partitioning

method that defines the center point of a cluster based on the *mean value* of the points within the cluster (Han et al., 2012).

The *K-modes* method extends K-means to address the clustering of nominal/categorical data. The K-modes method replaces the mean value with the *mode value* of the points within the cluster (Han et al., 2012) and uses new dissimilarities measures to deal with nominal objects.

3 Research Methods

The goal of this study is to identify profiles for instructors who teach Software Engineering in undergraduate courses in Brazil based on characteristics such as their classroom practices and the use of OSS projects in teaching.

This section presents the research methods used to reach this goal and answer the research question. First, we conducted a survey (Section 3.1). Next, we resorted to data mining techniques to identify the instructor profiles. We applied the K-modes (Section 3.2) to identify groups based on the similarity of the answers, and we used the data generated in the clustering to apply the Decision Tree algorithm (Section 3.3). Finally, we analyzed and consolidated knowledge to identify the profiles of software engineering instructors (Section 3.4). Figure 1 presents the methodological steps of this study.

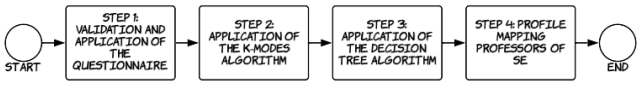

Figure 1. Methodological Steps

3.1 Survey

3.1.1 The Instrument

The SE instructor profile questionnaire[2] included 34 questions organized into six sections: (i) questions about familiarity with SE content; (ii) questions about familiarity with teaching methods and resources; (iii) questions related to familiarity with OSS projects; (iv) issues related to the use of OSS projects in the classroom; (v) questions related to aspects of OSS projects to instructors who used this approach in the classroom and questions related to aspects of classroom practices to instructors who have not yet used OSS projects in the classroom; and (vi) questions for data collection of a demographic nature.

3.1.2 Questionnaire Evaluation

After the construction of the questionnaire[3], we performed a pilot study with four SE instructors, two of them with experience using projects OSS in SEE and two with no experience. The instructors had access to the instrument and answered a form[4] evaluating (i) organization; (ii) objectivity; (iii) ease of reading and understanding the content; and (iv) time required to answer the questions.

After the instructors' feedback, we included new questions related to the instructors' motivation and new options for some questions. We also adapted the entire form to refer to the OSS project as an object of study in the classroom.

Figure 2 presents the result of the questionnaire validation. Instructors who used OSS projects in SEE and those who had not used them participated in the validation.

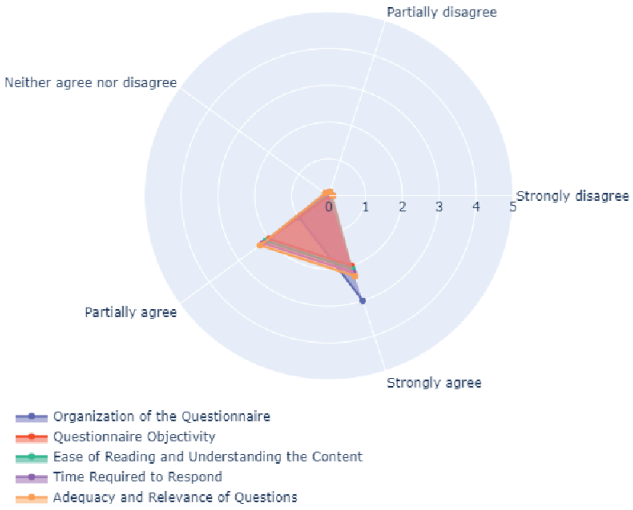

Figure 2. Questionnaire Validation

Figure 2 shows that 100% of the evaluators were satisfied with the organization, objectivity, ease of reading, understanding of the content, the time needed to answer the instrument, and adequacy and relevance of the questions.

Regarding the questionnaire organization, three instructors were "totally satisfied", and only one was "partially satisfied". Concerning the other criteria, two instructors were "totally satisfied", and the other two were "partially satisfied". We also considered suggestions for improvement and modified the instrument before its application.

3.1.3 Questionnaire Application

We sent the invitation to fill out the questionnaire via e-mail. Initially, a manual search identified 63 e-mails from instructors' websites and blogs. We also sent the invitation to the discussion list of the Special Committee on Software Engineering (CEES) of the Brazilian Computer Society (SBC)[5]. In 2020, the list had around 1200 subscribers. However, we could not find information about the number of members that were or have been instructors in SE courses. The questionnaire was available for 15 days with an extension for another 15 days.

We summarized the data collected through the survey with instructors, derived class/concept descriptions, and provided simple data summaries based on statistical measures. The research artifacts are available at `seed-br/profile-research.github.io`.

[2] `https://OSSgroup.github.io/project/index.html`
[3] `https://OSSgroup.github.io/project/index.html`
[4] `https://OSS2020.github.io/Instrumento/validarInst.html`
[5] `https://www.sbc.org.br/`

3.2 Clustering Analysis

To work with nominal/categorical data, we used the K-modes algorithm combined with the Hamming distance metric (or dissimilarity). For the application of data mining techniques, we use the Google Collaboratory[6] environment with pandas[7] and commands in Python[8].

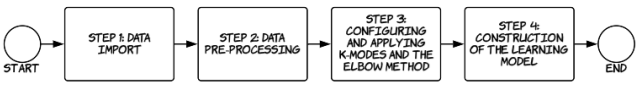

Figure 3. Steps for applying the K-modes algorithm

Figure 3 shows the steps taken to execute the K-modes algorithm. First, we imported the data collected by the questionnaire application into Google Collaboratory. Then, we prepared the data to align it with what the algorithm expected and solved problems related to the lack of values. We selected the relevant questions to provide a data set for applying the K-modes algorithm. We chose the columns corresponding to the questions about familiarity with SE contents, pedagogical methods, and resources with OSS projects and demographic aspects. We left aside the remaining columns because part of them was answered either by instructors who used OSS projects in the classroom or by instructors who did not use this teaching approach. This pre-processing step resulted in a data set in the matrix format, with dimensions of 90 rows by 36 columns.

In the pre-processing stage, we temporarily excluded the column with information related to the use of OSS projects to avoid influencing the distribution of instructors in the clusters. Then, we installed, updated, and imported the K-modes library into the Google Collaboratory environment. The K-modes algorithm requires a value of k to define the number of divisions in the data set. In order to minimize the subjectivity of choosing k, we used the Elbow Method. This method consists of executing the K-modes algorithm for different amounts of clusters. Such an approach tests the variance of the data concerning the number of clusters and considers the k value to be ideal when the gain (cost) does not represent a significant value.

Figure 4 presents the result of executing the Elbow Method. We can observe the distribution of the number of similar attributes within the cluster concerning the attributes outside that cluster. The graph shows a curve similar to an elbow from the 2 clusters. Therefore, we used the K-modes algorithm to generate two clusters. We build the K-modes model and the learning model training in the Google Collaboratory environment.

After executing the K-modes algorithm, we identified the centroid of the two generated clusters and the representatives of each cluster. Finally, for each instructor, we reintroduced the information about using OSS projects (previously removed) and the corresponding cluster.

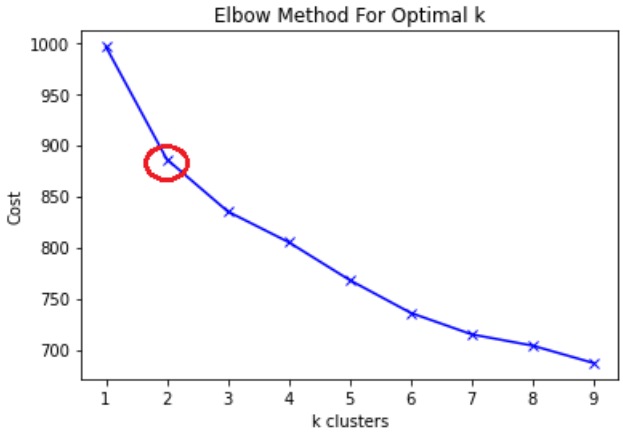

Figure 4. Result of the Elbow Method

3.3 Classification

Figure 5 shows the steps for the Decision Tree algorithm. First, we performed the processing and transformation of the results from the application of the K-modes algorithm. We created a new column to inform each respondent's cluster (group). Considering the nature of categorical data, we transformed the data set into a vector of 0's and 1's. The use of the pandas' library supported the automation of this operation.

Figure 5. Steps for Using the Decision Tree Algorithm

We split the data set into training (75%) and test (25%) subsets to measure performance. Once we found the solution based on the training set, the built classification model evaluated its performance based on the test data set. Then, we imported the `sklearn.tree` library, built the learning model, and trained it with the segmented part of the test data set.

We evaluated the generated model to check its generalizability. We computed measures of accuracy, coverage (recall), and precision. After evaluating the built learning model, we used the Feature Importance resource of Decision Trees. *Feature Importance* shows the attributes that better determine the classification. We extracted them from the questionnaire answers that influenced the classification regarding the use of OSS projects in the SEE.

3.4 Mapping the Characteristics of SE Instructors

We mapped the profile of Brazilian higher education instructors working in undergraduate SE courses. We used the clusters identified by the K-modes algorithm, the prediction resulting from the Decision Tree algorithm, and the instructors' responses concerning OSS projects' use (or not) in SEE.

We used `Google Data Studio`[9] to help map the instructors' profiles. `Google Data Studio` is a free data visualization tool that supports the creation of graphs, reports, and information panels, enabling integration with various data sources. Google Data Studio provided the visualization and representation of the data to support the analysis of the information generated by the machine learning algorithms.

[6] https://colab.research.google.com/
[7] https://pandaslibrary.pydata.org/
[8] https://OSSgroup.github.io/project/ComandoPyhton.html
[9] https://datastudio.google.com/

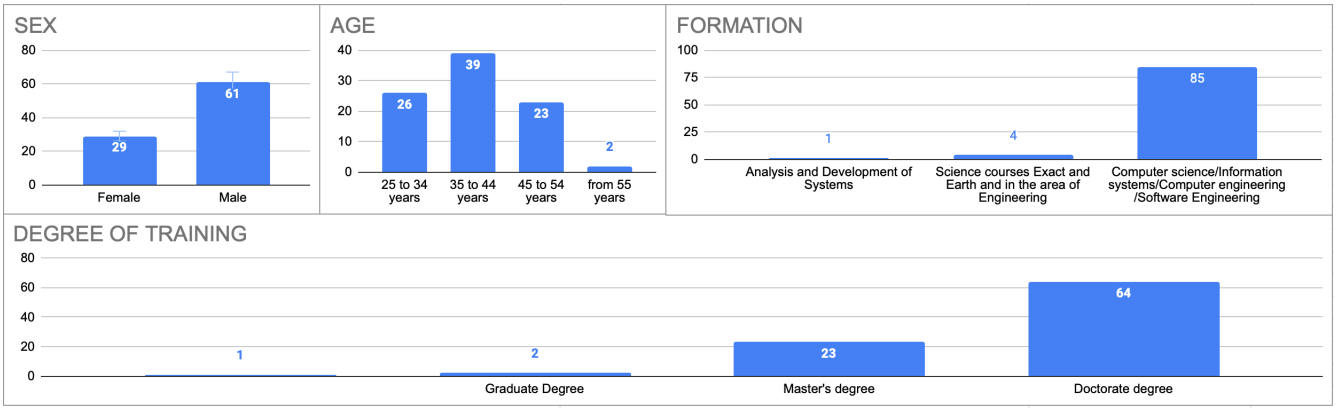

Figure 6. Demographic aspects of Software Engineering instructors

4 Results

This section presents the results of the survey study (Sections 4.1, 4.2 and 4.3), followed by the results of the profile mapping (Section 4.4).

4.1 Demographics

Demographic data included nominal attributes, such as gender, and ordinal attributes, such as age group. Figure 6 shows that 67.8% of the respondents are male, and 32.2% are female. Concerning age, 43.3% are 35 to 44 years old, 28.9% are 25 to 34 years old, and 25.6% are 45 to 54 years old. Only 2.2% of the respondents were 55 years old or more. Most respondents graduated in Computer Science courses, and only 4.4% graduated in related areas. Concerning the instructors' educational level, 71.1% held a Ph.D., 25.6% had a Master's degree, 2.2% Specialization, and 1.1% Bachelor's.

4.2 Classroom Practices

4.2.1 Instructors who use OSS in SEE

Concerning the frequency of using OSS in the classroom, 12 instructors (46.2%) used OSS five or more times; seven instructors (26.9%) used it 2 to 5 times, and seven instructors (26.9%) used OSS in the classroom only once.

Figure 7 shows information about the control level over OSS projects used in the classroom. Instructors could select one or more choices. Many instructors (53.8%) used "no control" at least once. They monitored the activities of the students during the use of an OSS project, but the community made the requests and approved (or not) the students' contributions. Less than a third of instructors (30.7%) used internal control with external approval from the OSS community at least once. Internal control with external approval means new features were proposed and built in the classroom and later submitted for the community's approval. Less than 46.2% of the instructors used internal control at least once. The instructors forked the project, defined the assignments, and evaluated the students' contributions. Only 7.7% of instructors had total control over tasks, developing and managing the core of the OSS projects.

Figure 8 presents the answers about selecting OSS project(s) for use in the classroom. Several instructors

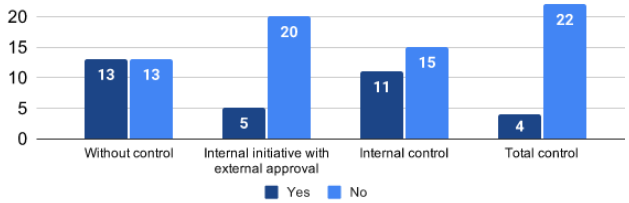

Figure 7. Control Level in the Classroom

(53.8%) selected the OSS project(s) to be used by all students at least once. Using a pre-defined list of projects was the preferred strategy of 65.5% of instructors, and students had to select the OSS project of their choice from such a list. Moreover, 42.3% of instructors allowed students to select an OSS project of their preference.

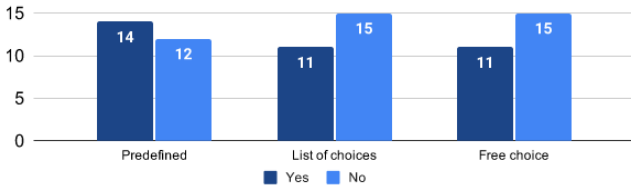

Figure 8. Choice of OSS Projects in the Classroom

Figure 9 shows the representation of perceptions regarding the benefits of using OSS in SEE. Most instructors who use OSS (83.5%) replied that its use is beneficial, as it brings real experience to the practice of SE content. Moreover, 65.4% of the instructors answered that using OSS is beneficial because it allows the freedom to run, study, share, and modify the software. Finally, 65.4% replied that using OSS is beneficial because it allows students to be in contact with third-party software, and 57.7% stated that the experience with OSS enables the active participation of students.

Figure 10 presents the instructor's perceptions regarding difficulties in using OSS projects. Among them, 38.5% had difficulties selecting the appropriate OSS project, and only 15.4% had difficulties adapting the OSS project practices to

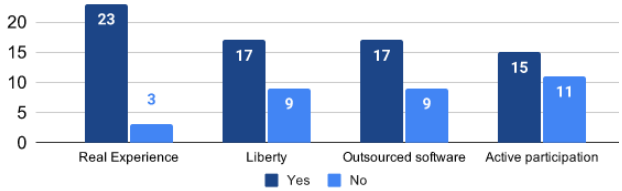

Figure 9. Benefits of using OSS in the Classroom

the course contents. Most instructors (61.5%) had difficulty understanding the OSS project code, artifacts, and practices, while 53.8% had difficulty configuring the environment. Finally, 38.5% had difficulty aligning the practice using OSS projects to the duration of the course's execution, and 46.2% had difficulty interacting with the open-source software community.

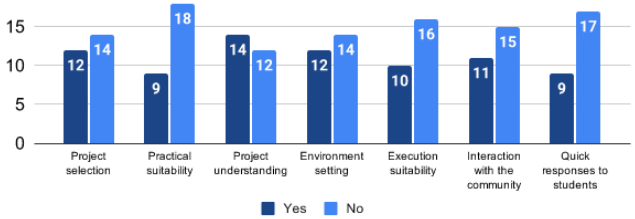

Figure 10. Difficulties in using OSS

4.2.2 Instructors who did not use OSS in SEE

Only 21 instructors (32.8%) were concerned about training their students to perform maintenance in real-world software systems in the classroom; and 19 (29.6%) were concerned with supporting their students to handle projects of different domains, sizes, and complexities. Concerning social skills, 53 instructors (82.8%) stated their interest in fostering the development of students' social skills in the classroom. Only 19 instructors (29.6%) declared to have the necessary technical knowledge to use OSS projects, and 18 instructors (28.1%) were concerned about time to plan classes using OSS projects. Finally, 37 instructors (57.8%) declared their interest in using OSS projects in SEE. Figure 11 presents the concerns and interests of instructors with no previous experience using OSS projects in SEE.

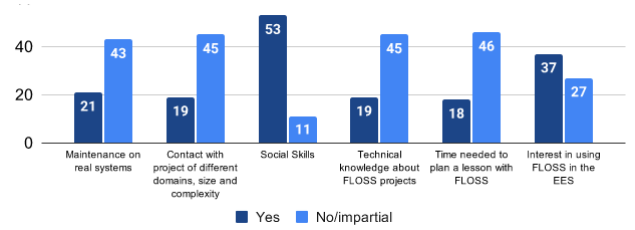

Figure 11. Concerns and interests of instructors with no previous experience in using OSS Projects in SEE

4.3 Mapping the Use and Interest in using OSS Projects

This Section presents an overview of the relationships between the instructors' responses from different perspectives. We present seven maps that combine information on the use of OSS projects, teaching experience in SE (in years), age group, learning approaches, assessment types, teaching strategies, familiarity and experience with OSS projects, and interest in using OSS projects in SEE.

4.3.1 Use of OSS and teaching experience (in years)

Figure 12 presents a map that combines OSS projects in SEE with teaching experience in SE courses (in years) and instructors' age groups. Most instructors who used OSS projects have 1 to 10 years of teaching experience in SEE and are 35–44 years old. Most of the participants who did not use OSS projects have 1 to 5 years of teaching experience in SEE and are 35–44 years old.

We will now move on to an analysis of each perspective presented in Figure 12. We noticed that 36.6% of instructors who used OSS projects in SEE have from 1 to 5 years of teaching SE, and the same percentage of instructors have from 6 to 10 years. 26.9% are over ten years, and only 3.9% are less than one year old. Considering instructors who did not use OSS projects in SEE, 35.9% have 1 to 5 years in teaching, 34.4% are over ten years, 23.4% have 6 to 10 years, and only 6,3% have less than one year of teaching experience. Regarding the age group, Figure 12 shows that most instructors who used OSS projects in the SEE (57.7%) are 35 to 44 years old, 26.9% are between 25 and 34 years old, and only 15,4% are 45 to 55 years old. No instructor who used an OSS project in SEE is over 55 years old. Considering instructors who have not used OSS projects in SEE, 37.5% are 35 to 44 years old, 29.7% are 25 to 34 years old, 29.7% are 45 to 54 years old, and only 3,1% of instructors are over 55 years old.

4.3.2 OSS projects, teaching approaches and assessment types

Figure 13 presents a map that combines the use of OSS projects in SEE with teaching approaches and assessment types used in the SE course. The majority of instructors who used OSS projects in SEE and instructors who did not use OSS, used Problem/Project/Research-based learning as a learning approach and applied Exams/Tests as a way to assess their students.

Concerning the teaching approaches (left), we noticed that 76.9% of instructors who used OSS projects in SEE used Problem/Project/Research-based learning. Moreover, 73.1% used Active learning (general), 61.5% used Case-based learning, 30.8% used Game-based learning, and only 7.7% used Formal learning. No instructor who did not use OSS in SEE opted for the OC2RD2[10] approach in the classroom. Considering instructors who **did not** use OSS projects in SEE, 76.6% used Problem/Project/Research-based learning, 51.6% used Active learning (general), 37.5% used Case-based learning, 20, 3% used Game-based learning, 9.4% used Formal learning and only 1.6% used OC2RD2.

Figure 13 shows that the majority of instructors who used OSS projects in SEE (84.6%) used Exams/Tests to assess their students, 76.9% applied Exercises, 73.1% used Seminars and Software Artifacts, 57.7% Reporting, 42.3% used SE Topic Search, 38.5% used Participation in Discussions, 19.2% used Interview and only 3.9% used Dynamics. No instructor who used OSS projects in SEE applied project development in the classroom. Considering instructors who did not use OSS projects in SEE, 90.6% applied Exams/Tests to evaluate their students, 81.3% used Artifact Software, 65.6% used Exercises, 62.5% used Seminars, 45.3% used Reporting, 42.2% used Participation in Discussions, 31.3% used SE Topic Search, 20.3% Interview and only 1.6% used project

[10] In Portuguese: Objetivo, Contratempo, Catástrofe, Reação, Dilema e Decisão

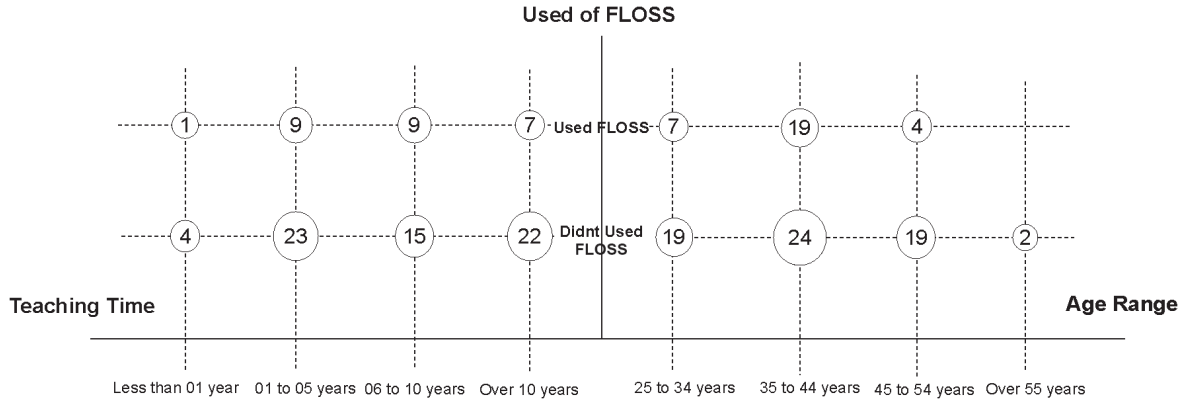

Figure 12. Use of OSS in SEE vs Teaching Time vs Age Group

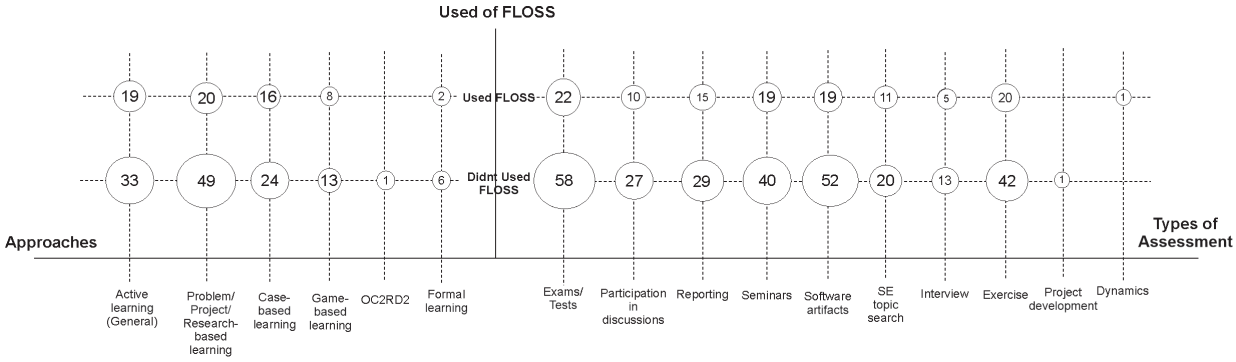

Figure 13. Use of OSS in SEE vs Approaches vs Age Group

development. No instructor who did not use OSS projects in SEE used Dynamics to assess their students in the classroom.

4.3.3 OSS and teaching strategy

Figure 14 shows the combination of using OSS projects in SEE with some classroom teaching strategies.

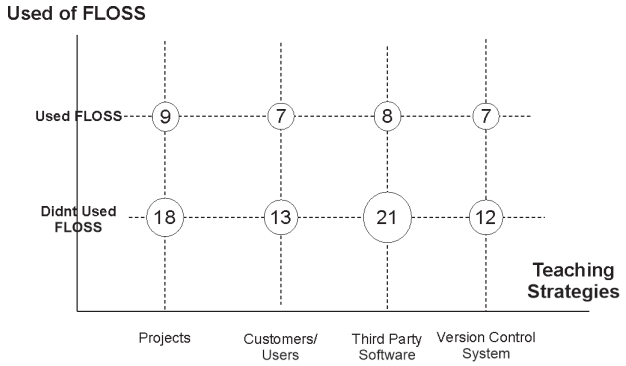

Figure 14. Use of OSS in SEE vs Teaching Time vs Age Range

Regarding teaching strategies, 34.6% of instructors who used OSS projects in SEE provided projects to be developed by their students in the practice of SE concepts, 30.8% of instructors affirm that their students deal with third-party software to practice SE content, 26.9% of instructors allowed their students' relationships with customers/users to practice SE content and 26.9% of instructors claim that their students use version control systems in the practice of SE content. Considering instructors who did not use OSS projects in SEE, 32.8% brought third party software to the classroom, 28.1% used project development, 20.3% provided contact with clients/users and 18.7% worked with version control systems in the classroom.

4.3.4 Use of OSS in SEE, familiarity and experience with OSS projects

Figure 15 presents a combination of the use of OSS projects in SEE with familiarity and experience with OSS projects.

Figure 15 shows that most instructors who have used OSS projects in SEE know the concepts and have already contributed to OSS projects. Most instructors who have not used OSS projects in the classroom have only read about it in the news or scientific articles.

In the analysis of familiarity with OSS projects, we noticed that half (50%) of instructors who used OSS projects in SEE know and contribute to communities of OSS projects, 38.5% have read about it and only 11.5% are not familiar with OSS concepts. Considering instructors who did not use OSS projects in the classroom, more than half (60.9%) read about the subject in the news or scientific articles, 37.5% know the concepts and have already contributed to OSS projects and only 1,6% are not familiar with OSS concepts.

Figure 16 presents a map that combines the frequency of using OSS projects in SEE with the choice of project and the level of control in the classroom. Regarding the choice of the project, the majority of instructors who used OSS 2–5 times and more than 5 times opted for a *choice list*, in which students could choose any project from the list provided by the faculty or staff support. However, most instructors who had only one episode using OSS projects in SEE used a predefined project, in which the faculty or support team chose the OSS project for all students.

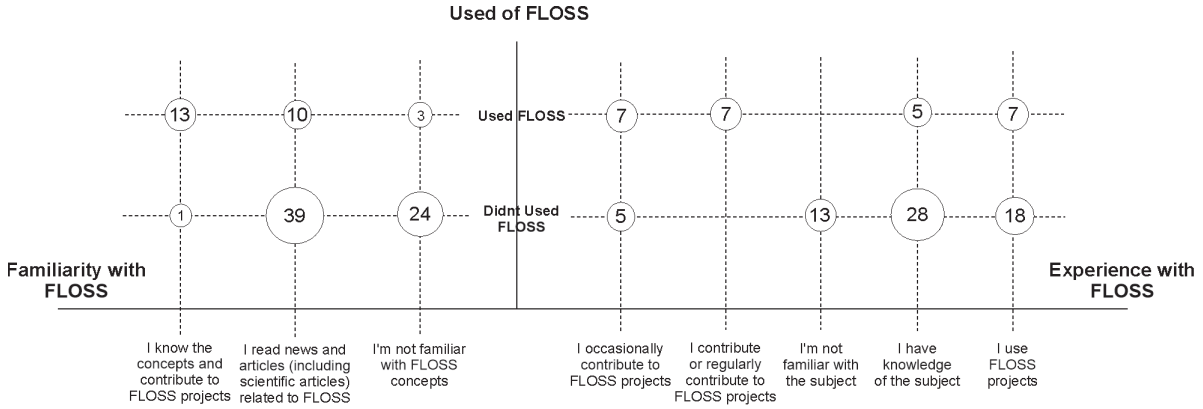

Figure 15. Use of OSS in SEE vs Familiarity with OSS vs Experience with OSS

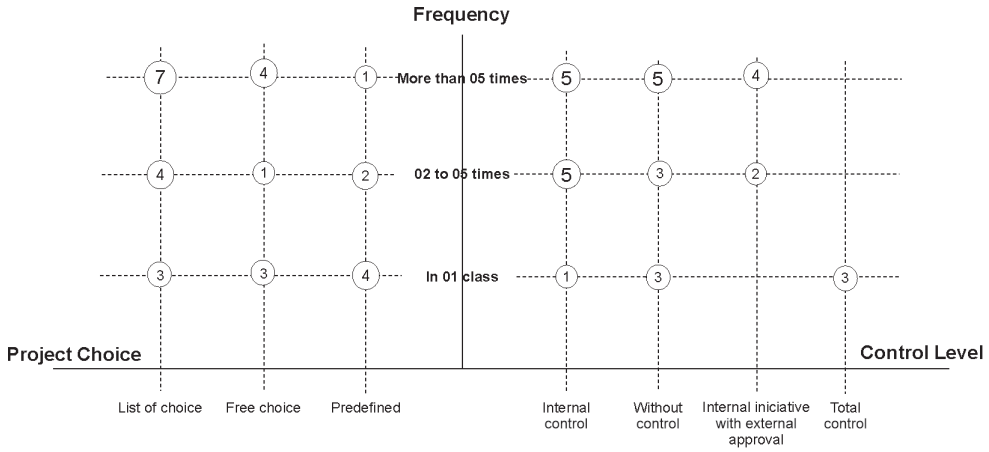

Figure 16. Frequency of Use of OSS in SEE vs Project Choice vs Control Level

Figure 16 shows the control level of the experiences of using OSS projects in SEE. The majority of instructors who used OSS more than 5 times opted for "internal control" or "without control". In "internal control", the faculty or support team forked the OSS project, prepared assignments and evaluated students' contributions. However, when adopted the "without control" level, the teaching staff or support team only monitored the student's activities within the OSS project: students work with requests from the community and the community approves the students' contribution. Most instructors that used OSS 2 to 5 times in the classroom opted for "internal control", while most instructors who had only one experience using OSS in SEE opted for "without control" or "full control". In "full control", the development and management of the OSS project core is carried out by the faculty or support team.

4.3.5 Interest in using OSS projects in SEE and experience

Figure 17 shows that most of the instructors concerned with training students to perform maintenance on real-world software systems, with different sizes, domains and complexity, showed interest in using OSS projects in SEE, while seven instructors remained neutral. Most instructors who were concerned with the opportunity to learn and develop social skills showed interest in using the approach.

4.3.6 Interest in using OSS, experience and familiarity with OSS

Figure 18 shows a combination of the interest in using OSS projects in SEE with the instructors' experience and familiarity with OSS projects.

With respect to experience with OSS projects, Figure 18 shows that most instructors who occasionally contribute to OSS projects and who use OSS projects show a total interest in using OSS projects in SEE. Most instructors who have knowledge about the subject show a partial interest in using OSS projects in the classroom. However, instructors who claim to be unfamiliar with the subject have shown no interest or disinterest in using OSS in SEE.

Regarding familiarity with OSS projects, the Figure 18 shows that the majority of instructors who know and still contribute to OSS projects show a total interest in using OSS projects in SEE. Most instructors who have read scientific news or articles on the subject show a partial interest in using OSS projects in the classroom. Most instructors who claim to be unfamiliar with OSS concepts have shown no interest or disinterest in using OSS in SEE.

4.4 Mapping Instructors' Characteristics

The survey data showed that 28.9% of instructors used OSS projects in the classroom, while 71.1% of instructors did not.

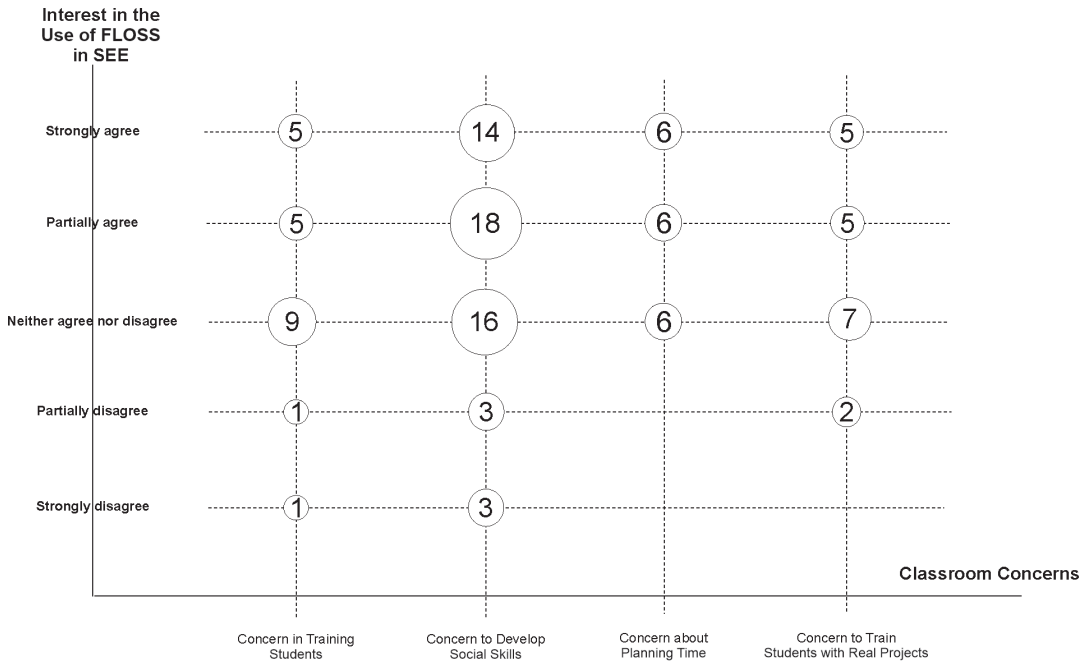

Figure 17. Interest in using OSS vs Concerns in the Classroom

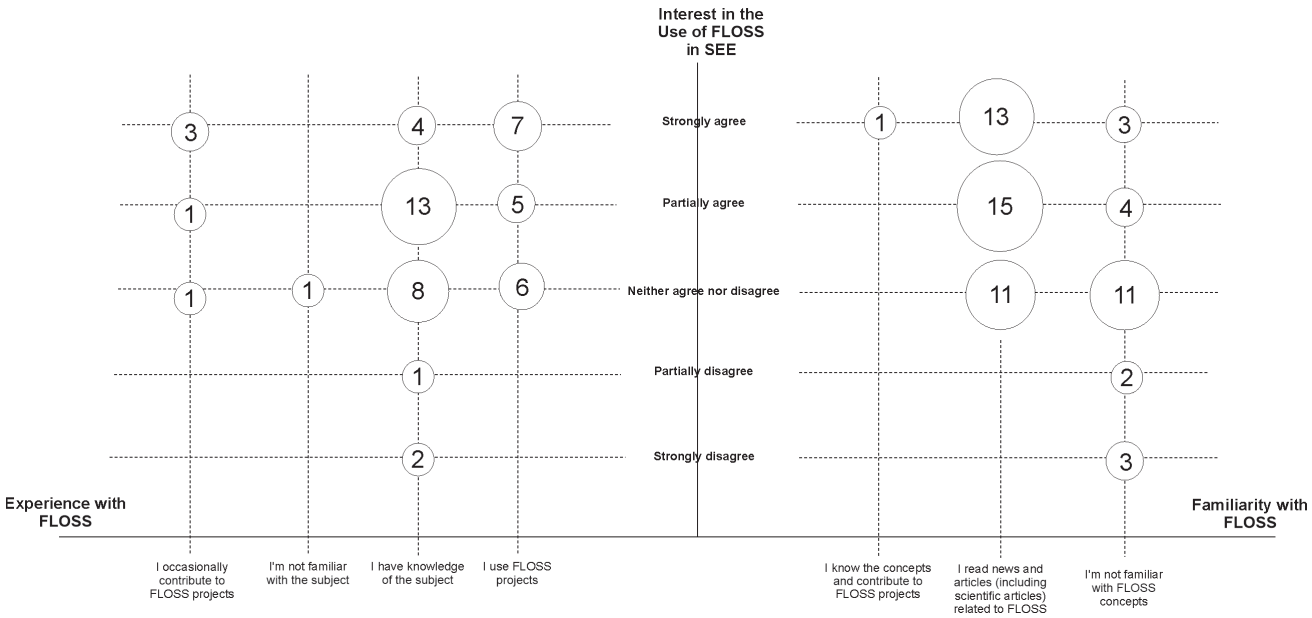

Figure 18. Interest in Using OSS vs Experience with OSS vs Familiarity with OSS

4.4.1 Application of the K-modes Algorithm

We disregard the information related to the use of OSS projects in SEE and the SE instructors were grouped in two clusters according to the similarity and dissimilarity of the responses in the execution of the K-modes algorithm. The representatives corresponding to each respondent are:

(0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0,
0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1,
1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0,
0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1,
0, 0, 0, 0, 0, 0, 1, 1, 0)

Representatives are arranged in the order in which the data set was made available for processing. Each "0" indicates that the instructor is grouped in `Cluster 0`, and each "1" characterizes the instructor in `Cluster 1`. Representatives were added to the data set processed by the K-modes algorithm. Following the application of the K-modes algorithm, 52 instructors (57.8% of the total respondents) were classified in `Cluster 0`, and 38 instructors (42.2% of the total respondents) were classified in `Cluster 1`.

Figure 19 shows the relationship between "the use of OSS projects in SEE" ("Yes"/"No") and the two clusters of instructors (`Cluster 0` and `Cluster 1`). In both clusters, there are instructors who used OSS projects in SEE, and accordingly, there are instructors who did not use OSS in the classroom.

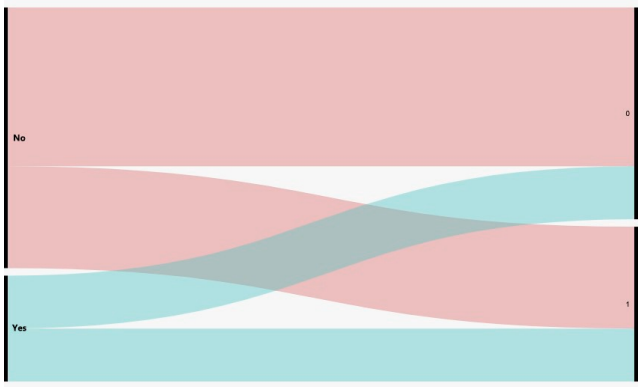

Figure 19. Relationship between the Use of OSS projects (left) and Clusters (right)

Instructors who used OSS projects are grouped in `Cluster 0` and `Cluster 1` (50%-50%). Instructors who did **not** use OSS projects in the classroom are grouped as follows: 60.9% in `Cluster 0` and 39.1% in `Cluster 1`.

After applying the K-modes algorithm, we identified the centroid of each cluster:

- `Cluster 0` **Centroid**:
  ["I totally agree" "I partially agree" "I totally agree" "Partially agree" "Partially agree" "Partially agree" "Partially agree" **"From 1 to 5 years"** **"Yes"** "Yes" "No" "No" "No" "No" "Yes" **"Yes"** "Yes" "No" "Yes" "Yes" "No" "No" "No" "No" "No" "Ever" **"Occasionally"** **"Often"** "Often" "I read news and/or articles (including scientific ones) related to OSS" **"I use OSS projects"** "Male" **"25 to 34 years-old"** "Computer Science / Information Systems / Computer Engineering / Software Engineering" "Doctorate degree" "Brazil"].
- `Cluster 1` **Centroid**
  ["Strongly agree" "Strongly agree" "Strongly agree" "Strongly agree" "Strongly agree" "Strongly agree" "Strongly agree" 'Over 10 years' "No" "Yes" "No" "No" "No" "No" "Yes" **"No"** "Yes" "No" "Yes" "Yes" "No" "No" "No" "No" "No" "Ever" **"Often"** **"Occasionally"** "Often" "I read news and/or articles (including scientific ones) related to OSS" **"I have knowledge on the subject"** "Male" **"35 to 44 years-old"** "Computer Science / Information Systems / Computer Engineering / Software Engineering" "Doctorate degree" "Brazil"].

Table 1 shows the divergent information between clusters. Based on the information from the centroid, we observed that out of the 36 columns in the processed data set, only seven presented divergent responses between the clusters.

Table 1. Divergent Answers Between Clusters

Columns	`Cluster 0`	`Cluster 1`
Teaching SE for:	1 to 5 years	Over 10 years
Uses active learning approach (general)	Yes	No
Uses report as type of evaluation	Yes	No
Students in contact with users to practice SE	Occasionally	Often
Students deal with third-party software to practice SE	Often	Occasionally
Experience with OSS projects	I use OSS projects	I have knowledge on the subject
Age range	25–34 years old	35–44 years-old

Data analysis showed that `Cluster 0` is characterized by younger instructors who have less experience in SEE, that use active learning and OSS projects, and with students that have contact with third-party software to practice SE content, while `Cluster 1` is characterized by instructors with the longest teaching experience in SE, with students that often relate to clients/users to practice SE content, that declared to be knowledgeable on OSS and that are older than the instructors assigned to `Cluster 1`.

4.4.2 Decision Tree Algorithm Application

To identify the characteristics that determine the use of OSS projects in SEE, we built a predictive model by means of the application of the Decision Tree algorithm. The algorithm used as a label the classification generated by the K-modes algorithm. In this study, the generated model has an accuracy of 86%, Recall of 46% and F1 score of 60%.

Following the application of the prediction model performed by the Decision Tree algorithm, 76 instructors (84.4%) were classified in `Group 0` and 14 instructors (15.6%) were classified in `Group 1`.

Figure 20 shows the relationship between the use of OSS projects in SEE and the grouping of instructors by the predictive model. We observed a subset of instructors who

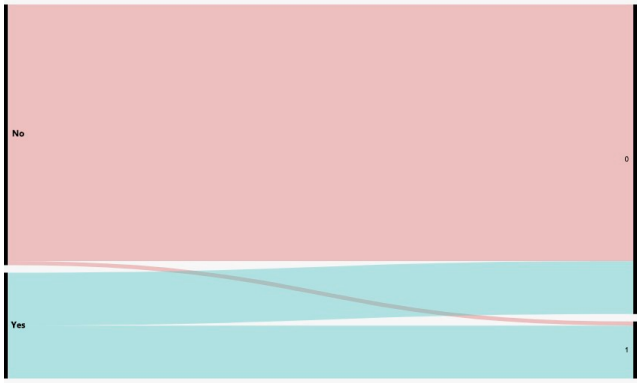

Figure 20. Relationship to Use of OSS Projects and Prediction

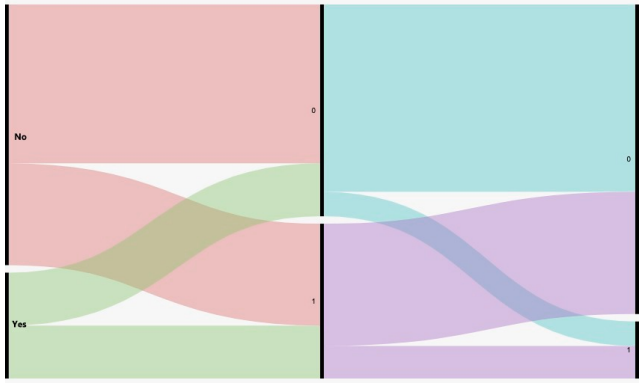

Figure 21. Relationship among the Use of OSS projects (left), Clusters (center) and Prediction (right).

used OSS projects in SEE ("Yes") and a subset of instructors who did not use this approach in the classroom ("No") in both groups. Concerning the instructors who used OSS projects ("Yes"), 50% were classified in `Group 0` and 50% in `Group 1`. We observed the same result in grouping instructors when applying the K-modes algorithm. Considering the instructors who did not use OSS projects in the classroom ("No"), 98.4% were classified in `Group 0` and only 1.6% in `Group 1`. Table 2 presents the key attributes (*Importances*) considered for the classification that were identified after applying the Decision Tree algorithm.

Table 2. Key-Attributes for Instructors Rating

Questions	Answers
Do you have knowledge about Software Requirements?	I totally agree
Do you have knowledge about Configuration Management?	I totally agree
Do you use active learning approach (general)?	No
Do students relate to customers/users to practice Software Engineering content?	Occasionally
How familiar are you with the concepts related to OSS?	I know the concepts and contribute to OSS projects

Figure 21 shows the relationship between the use of OSS projects in SEE (left), the clusters of instructors (center), and the classification groups of the predictive model (right). We observed instructors distributed in different combinations involving the use (or not) of OSS projects in the SEE, the two clusters, and the two classification groups. Instructors who use OSS projects in SEE were distributed in (1) `Cluster 1` and `Group 1`, or (2) `Cluster 1` and `Group 0`, or (iii) `Cluster 0` and `Group 1`, or (iv) `Cluster 0` and `Group 0`. Accordingly, we observed instructors who did not use OSS in the classroom distributed in such combinations.

5 Discussion of Results

With the results of this work, it was possible to present demographic data, information about familiarity with OSS projects and the use of this approach in the classroom by SE instructors in Brazil who answered the applied questionnaire. As far as we know, this information is new and brings a contribution by presenting information about instructors who teach the SE course in the country.

Availability of demographic data. This work presented demographic data about instructors in Brazil who answered the applied questionnaire.

Among the SE instructors in Brazil, the male gender predominates, aged between 35 and 44 years old, training in Computer Science and Information Systems courses and the doctorate level. Analyzing the crossing of data related to the demographic aspect, it was possible to observe that this predominance was also found in the group of instructors who used OSS projects in the SEE, of those who did not use it, of the instructors grouped in `Cluster 1` by the K mode algorithm and the two groups of instructors classified by the decision tree forecasting algorithm. There was divergence only in `Cluster 0`, where the predominance of the age group was 25 to 34 years.

Familiarity with Software Engineering and OSS. All the instructors declared to be knowledgeable in requirements, design/architecture, construction, evolution/maintenance, configuration management, and software quality. We observed that instructors who used OSS totally agreed that they knew all the contents mentioned above, while instructors that did not use OSS stated that they partially agreed to be familiar with configuration management, testing, and software evolution/maintenance.

Considering that 71.1% of the respondents have not yet had experience with the use of OSS projects in SEE, we believe that there is a possibility to increase the adoption of the use of OSS projects in SEE so that a more significant number of instructors have the first contact with this approach. Among the group of instructors who have not yet used OSS projects in the classroom, 60.9% belonged to `Cluster 0`. We could say that instructors who did not use OSS projects in SEE were more representative of `Cluster 0`. However, we cannot say that this cluster represents instructors who did not use OSS since it is necessary to analyze the centroid information generated by the K-modes algorithm. According to the information from the centroid, we observed that, out of the 36 columns present in the processed data set, only 7 showed divergent responses between the clusters. Therefore, we can

say that most answers are similar, even if instructors are participating in different clusters. This similarity can make it difficult to identify relevant characteristics that define the use or not of OSS in the SEE.

Instructors Characteristics. After the analysis of the instructors' characteristics in each cluster, we consider that `Cluster 1` grouped instructors who used OSS projects in SEE while `Cluster 0` grouped instructors who did not use it. The reason for such belief is that `Cluster 0` grouped more experienced instructors who: (i) often foster the students' contact with customers and users to practice SE content; (ii) occasionally encourage students to deal with third-party software to practice SE in the classroom; and (iii) stated to be only knowledgeable, with no experience with OSS projects.

Based on the clusters generated after the application of the Decision Tree algorithm, the representation of instructors who did not use OSS projects in SEE emerged in `Group 0`: 98.4% of the instructors did not use OSS projects in the classroom.

The key attributes in applying the prediction model were: *the use of an active learning approach* and *students' relationships with clients and users to practice SE content*. We also found them in analyzing the divergent responses between the clusters generated with the application of the K-modes algorithm. For the key attribute "the use of an active learning approach", we indicate a relationship with `Cluster 1`. For *students' relationships with clients and users to practice SE content*, we associated the response of the key attribute to `Cluster 0`. Based on the key attribute related to the *instructor's familiarity with the concepts related to OSS*, it is possible to state that the contribution to OSS projects is a characteristic present for 100% of the instructors classified in `Group 1`.

In the relation between the use of OSS projects in SEE, the clusters of instructors generated in the application of the K-modes algorithm and the classification of the prediction model was to observe that there are instructors distributed in all possibilities of grouping. Thus, we can affirm that the similarity or dissimilarity of the answers does not define the instructor's initiative regarding the use of OSS projects in the classroom. Regarding instructors who did not use OSS in the classroom, it is possible to notice that the result obtained by using the Decision Tree algorithm was more assertive than the use of the K-modes algorithm. We also emphasize the importance of the combined use of the two algorithms, because only in this way was it possible to identify the centroid of each cluster and the most important issues for the classification of the predicted groups. Even identifying a relationship between some divergent issues between clusters and some key attributes for instructor classification, it was still not possible to determine the characteristics that determine an instructor profile that uses OSS projects in SEE.

Classroom practices of instructors. The predominant classroom practices in the profile of instructors who used OSS in SEE are: the frequency of using OSS above 05 times; the uncontrolled level, where instructors only monitor the student's activities within the OSS project; the choice of the predefined project, in which the instructor defines the project that will be worked on by all groups; the benefit of offering an industry practice experience in the classroom; and the difficulty of understanding the code, artifacts and practices found in the OSS project.

With the exception of the concern to about developing the social skills of their students and the interest in using OSS projects in SEE, the other aspects are of interest to the minority of respondents who have not had the experience of using this approach in the classroom.

Mapping the use and interest in using projects OSS. Analyzing the information presented in the maps, we noticed the predominance in the teaching time of Software Engineering from 1 to 5 years old, in the age group of 35 to 44 years old, in the Problem/Project/Research-based learning approach and in the type of evaluation Exams/Tests, both for instructors who have used and for those who have not used OSS projects in SEE. Also considering the use of OSS projects in SEE, it was possible to notice divergences in the predominance of teaching strategies, familiarity with OSS projects and experiences with OSS projects. For instructors who used OSS projects in the classroom, there was a predominance, respectively, of project development for the practice of Software Engineering, knowledge of concepts, contribution to OSS projects and the use of OSS. For instructors who did not use OSS projects in the classroom, there was a predominance, respectively, of students' contact with third-party software, of familiarity with OSS content just by reading news and scientific articles.

Analyzing the relationship between the frequency of use of OSS projects in SEE with the choice of projects and the level of control, we believe that there is a preference for instructors who adopt this approach to start the experiment using the type predefined in the project choice and the total control of the activities that are performed by the students in the classroom.

In relation to the interest in using OSS projects in SEE by instructors who have not yet used such an approach, we identified a predominance of instructors who are concerned with developing social skills in the classroom, with the necessary planning time to apply the approach, who have knowledge on the subject and still contribute to OSS projects.

Threats to Validity. Our research presents threats to the validity of the conclusion, as it presents the results of a set of 90 instructors who teach SE in Brazil and, therefore, we cannot generalize the results in this context. We also have validity threats regarding the choice and exploratory use of three algorithms, K-means and K-modes (for clustering) and Decision Tree (for classification) and the need to explore other strategies for identifying instructor profiles.

6 Related Work

Related work presents findings of the use of OSS projects in teaching SE (Pinto et al., 2017; Silva et al., 2019; Nascimento et al., 2018) either focusing on the instructors' or stu-

dents' perspectives. However, to the best of our knowledge, our work is unique. We performed a survey study with SE instructors and received 90 responses. We resorted to data mining techniques to build a profile mapping for instructors who teach SE courses in Brazil with or without the support of OSS projects and the practices they used in the classroom.

Pinto et al. (Pinto et al., 2017) interviewed seven SE instructors who changed their pedagogical practices so that students perform comprehension and maintenance tasks on OSS projects as part of their course. Some of their findings are: (i) there are different ways to use OSS projects in SE courses in terms of project selection, assessment, and learning objectives, and (ii) there is evidence of clear benefits of this approach, including improving students' social and technical skills.

Silva and colleagues (Silva et al., 2019) presented an experience report on using OSS projects while teaching UML diagrams to undergraduate students. The instructor had no experience in using OSS projects. The research team helped in elaborating the course plan, including selecting OSS projects and creating examples to be used in the classroom. The effectiveness of adopting OSS in SE courses was investigated, based on the instructor's and students' perspectives, concerning the course's planning, execution, and evaluation.

Nascimento et al. (Nascimento et al., 2018) investigated higher education students' perceptions about their contact with open source projects as a real-world experience. They conducted three mixed-methods case studies with three different undergraduate classes. The first group of students focused on software maintenance and evolution, the second class on software testing, and the third group on software requirements reverse engineering. However, their survey studies focused only on students.

7 Final Considerations

This work presented the profiles of 90 instructors who teach Software Engineering in undergraduate Computing majors in Brazil. We performed an online survey and classified the collected data using data mining techniques. To the best of the authors' knowledge, no related work addresses the characterization of Software Engineering instructors concerning their use of OSS projects in the classroom. Furthermore, we did not find related studies that used data mining techniques to identify the profiles of SE instructors.

While designing the questionnaire, we considered only the facets presented in a systematic mapping of the use of OSS projects in SEE (Nascimento et al., 2013; Brito et al., 2018). The analysis of the data extracted from the application of the questionnaire allowed us to perceive the need to increase the adoption of OSS projects in SEE and that familiarity with OSS projects can increase instructors' interest in adopting this approach.

By applying the K-modes and Decision Tree algorithms, we found that the answers regarding SE knowledge, methods, and pedagogical resources used are similar in both groups of instructors. However, familiarity with OSS projects and using an active learning approach were characteristics present in applying the K-modes and Decision Tree algorithms, identifying instructors who used OSS in Software Engineering Education from those who did not.

In the profile of instructors who used OSS in SEE, the following practices used in the classroom predominate: using OSS projects more than five times, with "no control", with a predefined list of projects selected by the instructor, recognizing as a benefit the use of industry practices in the classroom, and understanding the OSS project as a difficult task. Based on the maps, we speculate that the instructor's familiarity and experience with OSS projects favor the interest and the possibility of adopting this approach in SEE.

The profile mapping of SE instructors in Brazil and the practices used in the classroom can imply significant improvements for SEE with the increased adoption of OSS projects and the proposal of new strategies to support SE instructors in adopting OSS projects in the classroom. In future work, we plan to carry out a detailed analysis of other aspects that may determine strategies for adopting OSS projects in SEE. Moreover, we plan to investigate platforms to support students' participation in OSS projects and other aspects related to their use in teaching specific SE topics.

References

Banimustafa, A. and Hardy, N. (2020). A scientific knowledge discovery and data mining process model for metabolomics. *IEEE Access*, 8:209964–210005.

Brito, M. S., Silva, F. G., G. Chavez, C. v. F., Nascimento, D. C., and Bittencourt, R. A. (2018). Floss in software engineering education: an update of a systematic mapping study. In *Proceedings of the XXXII Brazilian Symposium on Software Engineering*, pages 250–259.

Ferreira, T., Viana, D., Fernandes, J., and Santos, R. (2018). Identifying emerging topics and difficulties in software engineering education in brazil. In *Proceedings of the XXXII Brazilian Symposium on Software Engineering*, pages 230–239.

Flach, C. v. and Kon, F. (2021). Software livre: Pre-requisito para a ciencia aberta. *Computação Brasil*.

Gutica, M. (2018). Improving students' engagement with large-team software development projects. In *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, pages 356–357.

Han, J., Kamber, M., and Pei, J. (2012). *Data Mining: Concepts and Techniques, 3rd ed.* Kaufmann.

Leite, J. C. S. d. P. and Werner, C. M. L., editors (2008). *Anais do Fórum de Educação em Engenharia de Software, FEES 2008, Rio de Janeiro, RJ, Brasil, Outubro, 2008*.

Marques, A., Ferreira, B., Lopes, A., and Silva, W. (2020). Stimulating the development of soft skills in software engineering education through design thinking. In *Anais do XXXIV Simpósio Brasileiro de Engenharia de Software*, Porto Alegre, RS, Brasil. SBC.

Nascimento, D. M., Chavez, C. F., and Bittencourt, R. A. (2018). The adoption of open source projects in engineering education: a real software development experience. In *2018 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE.

Nascimento, D. M., Cox, K., Almeida, T., Sampaio, W., Bittencourt, R. A., Souza, R., and Chavez, C. (2013). Using open source projects in software engineering education: A systematic mapping study. In *Frontiers in Education Conference, 2013 IEEE*, pages 1837–1843. IEEE.

Nascimento, D. M. C., von Flach Garcia Chavez, C., and Bittencourt, R. A. (2019). Does FLOSS in software engineering education narrow the theory-practice gap? A study grounded on students' perception. In Bordeleau, F., Sillitti, A., Meirelles, P., and Lenarduzzi, V., editors, *Open Source Systems - 15th IFIP WG 2.13 International Conference, OSS 2019, Montreal, QC, Canada, May 26-27, 2019, Proceedings*, volume 556 of *IFIP Advances in Information and Communication Technology*, pages 153–164. Springer.

Park, H. and Jung, J.-Y. (2020). Sax-arm: Deviant event pattern discovery from multivariate time series using symbolic aggregate approximation and association rule mining. *Expert Systems with Applications*, 141:112950.

Pinto, G. H. L., Figueira Filho, F., Steinmacher, I., and Gerosa, M. A. (2017). Training software engineers using open-source software: the professors' perspective. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*, pages 117–121. IEEE.

Silva, F. G., Brito, M. S., Tavares, J. V. T., and Chavez, C. v. F. G. (2019). Floss in software engineering education: Supporting the instructor in the quest for providing real experience for students. In *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, pages 234–243.

Silva, F. G., Lessa, M. S. B., da Luz Lopes, N., and von Flach G. Chavez, C. (2020). Teaching uml models with floss projects: A study carried out during the period of social isolation imposed by the covid-19 pandemic. In *Proceedings of the 34th Brazilian Symposium on Software Engineering*, pages 483–492.

Smith, T. M., McCartney, R., Gokhale, S. S., and Kaczmarczyk, L. C. (2014). Selecting open source software projects to teach software engineering. In *Proceedings of the 45th ACM technical symposium on Computer science education*, pages 397–402. ACM.

Wen, M., Siqueira, R., Lago, N., Camarinha, D., Terceiro, A., Kon, F., and Meirelles, P. (2020). Leading successful government-academia collaborations using floss and agile values. *Journal of Systems and Software*, 164:110548.

Yin, Y., Long, L., and Deng, X. (2020). Dynamic data mining of sensor data. *IEEE Access*, 8:41637–41648.