Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Experimental Methods in Computer Science (and in Informatics Engineering)

DEI-FCTUC, 2023/2024

Assignment 2 – Design of an experiment

1. Introduction

This assignment is an exercise of designing an experiment. Although the main goal is pedagogic, the topic proposed for this assignment (see below) allows a highly realistic experiment design.

There are many types of experiments and the term "design of experiments" (often referred as experimental design as well) is used for the process of defining and planning experiments in such a way that the data obtained can lead (after some analysis that depend on the nature of the experiment) to technically and scientifically valid conclusions.

The topic of designing experiments of different types and for different purposes has been addressed in detail in the lectures (T classes). In any case, the next paragraphs present a brief overview before describing the assignment proposed.

A classic experiment includes the following elements (you can take them as basic steps for the purpose of this assignment):

1. Problem statement (or research question).
2. Identify variables.
3. Generate hypothesis.
4. Define the experimental setup/scenario.
5. Develop the tools and procedures (often called experiment protocol) required to run the experiment.
6. Run the experiments, collect the data/measurements, and perform basic statistical treatment to the measurements.
7. Perform data analysis, which may include hypothesis testing.
8. Draw conclusions (and often go back to the beginning and reformulate the problem statement or test a different hypothesis)

The formulation of sound problem statements (or research questions) is not easy, and this is particularly true for the computer science and informatics engineering field. A good (i.e., relevant problem statement) should be focused enough to allow the clear identification of the variables of the problem but, at the same time, should be sufficiently open to allow different hypothesis to answer the problem/question.

An example of formulation of a possible problem statement is:

How does the setting (noise, temperature, music, open space, etc.) of the room used by programmers affect the number of bugs found by test suits in the modules produced by such programmers?

The effect we want to measure (dependent variable) is the number of bugs found and the factors (independent variables) are the different room settings and situations. It is assumed that there is a set of conditions that remain stable (e.g., type of programming language, complexity of the modules, etc.). The basic type of experiments only changes one factor (i.e., independent variable) at the time. More complex experiments use a factorial approach in which several factors are changed together to make the experiment more efficient and allow the study of possible interactions among factors. For the time being, in the context of the present assignment, the recommendation is to consider only one factor at the time.

A possible hypothesis for the example of problem statement mentioned above is that listening to classic music at comfortable volume has no effect on the number of bugs programmers make while develop code. As we know, a hypothesis (called H_0) always includes an alternate hypothesis (H_1) that will be taken if the experiments allow us to reject the hypothesis H_0. In this case, H_1 could state that programmers make fewer bugs if they listen to classic music at a comfortable volume while programming. In summary, a possible hypothesis for the example of problem statement under analysis could be:

- H_0 - classic music at comfortable volume has no effect on the number of bugs programmers make.

- H_1 - classic music at comfortable volume reduces the number of bugs programmers make.

In this case, the hypothesis is directional (i.e., establish a direction for the dependency between the dependent variable and the independent variable) but the hypothesis could simply state that music influences the number of bugs (non-directional hypothesis).

Even in a simple example like this, it is easy to understand the difficulties associated to the correct validation of hypothesis. For example, it is necessary to perform experiments with a representative group of programmers, use a variety of software under development, assure that the condition of the different experiment runs remain stable, etc. The analysis of the results to check if their distribution fit the hypothesis requires adequate statistic testing techniques, as has been presented and discussed in the lectures. In most of the cases, the assumption that the observations (i.e., results directly gathered from the experiment) follow approximately the normal distribution (or a T student distribution) greatly simplifies the test statistics and allow us to use parametric techniques. But care is necessary to certify that this assumption is valid for the experiment at hand.

2. Assignment goal

The purpose of the assignment is to design and run an experiment and execute all the steps to evaluate the following problem statement:

Are software code inspections worthwhile?

This research question is very broad, especially because the notion of "being worthwhile" can have many interpretations. To make things more concrete, let us assume you are working for an international software development company and your manager told you that the company wants to be sure that software code inspections are really effective, since it is well-known that

software code inspections are expensive. In practical terms, your company define effectiveness in the following way:

Software code inspections must detect at least 70% of the bugs that may exist in the inspected code, with no more than 25% of false positives, with 95% of confidence.

Design and run an experiment to answer the question whether software code inspections are worthwhile or not, considering the way inspection effectiveness was defined by your company?

As happen most of the cases in experiment design, it is necessary to learn/know the concepts about the topic or subject of the experiment. In the present assignment, the topic selected is **software code inspections**. Thus, the students should research for information on software code inspections to learn and understand the basic aspects of software inspections.

It is important to consider that there are two types of code inspections:

- Classic Michel Fagan's code inspections proposed long time ago. This type of inspections requires a minimum of 3 inspectors and a meeting to inspect/review the code. Since this is a very expensive method it is not used very often.
- Modern code reviews have been started by the open-source initiative and later popularized by companies such as Google and Facebook. Modern code reviews are normally associated with pull requests in distributed version control systems like Git, as it is necessary to review the committed code before incorporating/merging changed or new code into the main branch. In modern code reviews the code is reviewed by only one programmer and there is no review meeting. Modern code reviews are much cheaper and faster than classic Fagan's code inspections, but obviously they are not as effective as Fagan's inspections.

In Assignment 2 you can use either classic Fagan's code inspections or modern code reviews. Naturally, the experiment is easier with modern code reviews. In any case, your need to learn the basic ideas about code inspections/reviews (it is very easy).

This assignment also requires all the concepts and techniques that have been learned and discussed in the lectures. In addition to experiment design concepts and hypothesis testing, key notions of **experiments with people** are also relevant (it is also a topic of the lectures). In fact, software development in general and software code inspections in particular are human intensive tasks, which means that any experiment designed to assess the effectiveness of software code inspections will involve people as key participants.

The experiment should be as realistic as possible, and students are encouraged to add all the elements they consider relevant to allow a meaningful evaluation of the proposed research question. The main research question provided can be refined and include other aspects that the students may feel relevant and interesting, taking into account the general goal. For example, if your results show that the effectiveness of code inspections is much lower that the target provided by your company (detection of at least 70% of the bugs and no more than 25% of false positives), you can do many interesting things. For example, to determine the percentage of detection and false positives that can be achieved with 95% of confidence. Or study why people involved in the code inspections cannot perform better than the results observed, investigating how complex and difficult they think the code inspection was (e.g. using NASA TLX tasks load index- https://humansystems.arc.nasa.gov/groups/TLX/ to get additional information).

The PL classes planned to provide support to the development of this assignment are the primary instance to help students in the development of this assignment. In fact, the broad way the assignment is proposed leaves many possibilities for the students to consider (and research) in their design of the experiment. The definition of the assignment does not indicate the approach to be used, the steps, the type of analysis etc., as well as it does not provide

background on key aspects of the experiment topic. This is part of the work that will be done by the students. PL classes will be used to help and guide students in this journey. Additionally, these classes will also be used to track the progress made by each group of students week after week.

3. Assignment outcome

The outcome of the assignment is a written report (PDF file). The report should describe all the steps of the design of the experiment with enough detail to allow others to reproduce the experiment and, of course, should provide clear conclusions about the problem statement.

Given that the sole outcome of the assignment is a report, the quality of the document (i.e., structure of the report, precision of the results reported, quality of writing) is of paramount importance. There is no suggested template for the report structure. The goal is to avoid the rather passive situation in which the students just fill in a given report structure. On the contrary, defining the best structure for the report is part of the goals of the assignment. The teacher is fully available to discuss the proposed structure with the students, as well as any other aspect of the report.

In addition to the report, the students must keep a folder with all the programs, scripts, tools, surveys, etc. used or developed for this assignment. This folder could be useful during the final assignment defence.

4. Resources

Students are supposed to use their own computers or virtual machines provided by the Department. Free software code repositories available on the Internet are the most obvious source of programs to be used in this experiment. But any other source of existing programs is acceptable. Concerning tools to support the data analysis and hypothesis testing, students can select the tool they want. Possible choices include DATAtab, R, Matlab, Excel, among others.

5. Calendar and miscellaneous

The students' groups already defined for Assignment #1 must be maintained for this assignment. The **deadline** was already defined in the very first class on September 15th, and is **December 15th, at 23:59**.

The written teste that replaces the traditional oral defences is scheduled to **December 15th** at **13h**. The duration of the test is 45 minutes. As planned, the grades will result from the evaluation of the written report (group evaluation) and the individual scores obtained bay each member of the group in the written test.

Plagiarism means mandatory fail in the course and internal (UC) disciplinary procedure. Please, refer adequately all text and material you take from the Internet. All parts of the report must be written by the students and not copied & pasted & changed from the Internet or from other collages' reports.

Students are free to use ChatGPT and similar tools (Careful! I'm not encouraging the use of such tools; I'm simply pragmatically saying that I cannot control that and, consequently, I do accept the use of such tools). In any case, do not forget that the written test will be done without any support from tools.