Effect of Requirements Analyst Experience on Elicitation Effectiveness: A Family of Quasi-Experiments

Alejandrina M. Aranda, Oscar Dieste, Jose Ignacio Panach, and Natalia Juristo

Abstract—*Context.* In software engineering there is a widespread assumption that experience improves requirements analyst effectiveness, although empirical studies demonstrate the opposite. *Aim.* Determine whether experience (interviews, eliciting, development, professional) influences requirements elicitation using interviews. *Method.* We ran 12 quasi-experiments recruiting 124 subjects in which we measured analyst effectiveness as the number of items (i.e., concepts, rules, processes) correctly elicited. The experimental task was to elicit requirements using the open interview technique followed by the consolidation of the elicited information in domains with which the analysts were and were not familiar. *Results.* In unfamiliar domains, interview experience, requirements experience, development experience, and professional experience does not have any relationship with analyst effectiveness. In familiar domains, effectiveness varies depending on the type of experience. Interview experience has a positive effect, whereas professional experience has a moderate negative effect. Requirements experience appears to have a moderately positive effect; however, the statistical power of the analysis is insufficient to be able to confirm this point. Development experience has no effect. *Conclusion.* Experience impacts analyst effectiveness differently depending on the problem domain type (familiar, unfamiliar). Generally, experience does not account for all the observed variability in effectiveness, so there are other influential factors.

Index Terms—Elicitation, requirements analyst, experience, effectiveness, problem domain, quasi-experiment

1 Introduction

Requirements elicitation is generally acknowledged as being one of the most important activities in software development to understand customer needs [1], and it has a direct impact on software system quality [2]. It is an activity that requires intense communication between stakeholders (e.g., clients and analysts). Human interaction plays a critical role in this context [3].

There are a number of personal characteristics that may influence the effectiveness of any requirements-related task: experience [4], [5], [6], academic education [7], [8], cognitive capabilities [9], domain knowledge [1], [5], [10], [11], [12], [13], etc.

The idea that experience [14], [15], [16] improves requirements analyst effectiveness is widespread in the software engineering (SE) community. However, empirical studies that experimentally research the effect of experience [4] [5], [6], [9], [17] have not been able to demonstrate that experience has a positive effect. Note that existing works lack replications, the number of subjects is not very large, and most of them do not recruit subjects with several levels of experience.

The aim of this article is to determine whether *the experience of subjects that play the role of requirements analysts influences effectiveness when using interviews*. According to IEEE [18], effectiveness is *the accuracy and completeness with which users achieve specified goals*. So, elicitation effectiveness can be considered as the number of requirements successfully elicited. Effectiveness is measured comparing the requirements extracted by the subjects with the requirements that appear in the yardstick elaborated by the experimenters. IEEE defines analyst as *systems engineer that develops the system requirements, who is skilled and trained to analyse problems*. We conducted a sequence of quasi-experiments analysing the effectiveness of requirements analysts depending on years of experience. The subjects that participated in the quasi-experiments were developers with different levels of experience recruited at the School of Computer Engineering, Universidad Politécnica de Madrid, and at an empirical fair as part of an international conference on software quality and requirements (REFSQ 2013). Independent variables used in the experimental design are defined to analyse how different aspects of experience may influence requirements elicitation effectiveness. These independent variables are: the experience with interviews; the experience in eliciting requirements, experience in software development, and professional experience. The metric for all these independent variables is the number of years of experience in industry.

- *Alejandrina M. Aranda, Oscar Dieste, and Natalia Juristo are with the Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid, 28660 Boadilla del Monte, Spain. E-mail: am.aranda@alumnos.upm.es, {odieste, natalia}@fi.upm.es.*
- *Jose Ignacio Panach is with the Departament d'Informàtica, Universitat de València, 46100 València, Spain. E-mail: joigpana@uv.es.*

Manuscript received 28 February 2022; revised 28 August 2022; accepted 19 September 2022. Date of publication 28 September 2022; date of current version 18 April 2023.
This work was supported in part by under Project PGC2018-097265-B-I00, in part by SREC under Grant PID2021-123824OB-I00, in part by the Spanish Ministry of Science and Innovation and co-financed by FEDER under Grant GV/2021/072 from Generalitat Valenciana. Alejandrina Aranda holds a PhD grant from Itaipú Binacional, Paraguay.
(Corresponding author: Jose Ignacio Panach.)
Recommended for acceptance by S. Nejati.
This article has supplementary downloadable material available at https://doi.org/10.1109/TSE.2022.3210076, provided by the authors.
Digital Object Identifier no. 10.1109/TSE.2022.3210076

At first sight, we can think that the effect of the experience should be more visible for unfamiliar domains. Novice subjects may have more problems within an unfamiliar domain, the contrary for experts. So, we opt for studying familiar and unfamiliar domains separately. This allows comparing experience results in both types of domains. Our results reveal that the effect of experience on analyst effectiveness varies as a function of the problem domain type in which the elicitation takes place. For the *familiar problem domain*, positive effects are observed for *interview experience* and, possibly, *requirements experience*. On the other hand, *professional experience* and *development experience* appear to have a negative effect. For the *unfamiliar problem domain*, none of the experience types has any effect. Our results also clearly indicate that, in both domains, familiar and unfamiliar, more experienced analysts are only slightly more effective than less experienced analysts.

The article is structured as follows. Section 2 reports the background and work related to this research. Section 3 describes the research methodology. Section 4 details the quasi-experiments conducted and reports the results, which are discussed in Section 5. Section 6 points out the main validity threats. Finally, Section 7 outlines the conclusions.

2 Background

The study of experience goes back a long way. In the early days, its aim was to ascertain which factors make expert subjects perform better. Studies by Groot [19] and by Chase and Simon [20] revealed two main characteristics that experts had in common: thorough knowledge of, and long years of service in, their field of expertise. Experience is not related to any inborn talent, like intelligence, and it is very specialized, that is, experience is not transferable from one person to another and, within a person, from one area to another [21].

Appendix A summarizes the major empirical studies addressing the influence experience related to engineering and requirements experience.

2.1 Studies of Software Engineering Experience

Experience has also been the subject of study in Software Engineering (SE) since its infancy. In the 1980s, the fields of interest were programming and low-level design [22]. Since then, experience has been studied in almost all SE areas: design [23], usability [24], testing [25], etc. Generally, the theory of experience proposed by Chase and Simon [20] has been repeatedly confirmed. For example, experts learn quickly test-driven and after a short training, they can become effective in performing small programming tasks [26], or experts find easily the right people for help and they can take corrective actions to address knowledge gaps [27]. Expert subjects have also been proven to perform better in RE. Two studies, namely Rosenthal et al. [28] and, Moreno-Montes et al. [29], confirm that expert analysts have specialized knowledge and use more complex reasoning than inexperienced analysts.

There are works that have analysed the elements that influence the programmer experience, such as programming environment, design documents, programming codes, and evaluation methods [30]. It is well-established that it takes about 10 years or 10,000 working hours to gain the required experience [31], [32] although this is a variable figure that depends on the area and type of instruction received [33]. In fact, shorter times are specified for SE. Campbell and Bello [34] claim that programmers need at least two years to become experts in Smalltalk. Sim et al. [35] consider that five years of experience is time enough for a software engineer to become an expert, which is the same figure as suggested by Atkins [36] and Shrikanth [37].

The passage of time is a necessary, albeit not sufficient, condition for gaining experience. Apparently, novice software engineers have quite often been found to outperform experts [31], [33], [38]. This has led to the time spent performing an activity (*experience*) being dissociated from proficiency at performing an activity (*expertise*). An expert is anyone who outperforms his or her colleagues, not a person with a long working history. Casual effectiveness of an activity over a long period of time is not sufficient to achieve an expert level of skill; in turn, *expertise requires a prolonged period of intensive training* (e.g., the abovementioned 10 years) [39]. This explains why experience and expertise do not match [32], [40], [41].

To avoid confusion, we will use the terms *experience* or *experienced person*, and *expertise* or *expert*, according to their **restricted meanings** (as specified above) in the remaining of the paper.

The most common way of defining a SE professional's experience is *by years of experience* [42], as applies, for example, in job postings. It is not usual in SE for professionals to participate in lifelong training activities (that is, intensive training) once they have completed their higher education [35]. This suggests that experience and effectiveness in SE should be at least partly correlated. Consequently, *we wonder how acceptable it is to use experience as a predictor of analyst effectiveness*. As analysts perform a variety of tasks, we focus on the requirements elicitation activity. This activity is not only important to assure the quality of the future software system, but also constitutes one of the biggest challenges for analysts [43]. It is reasonable to assume that experience, if it is clearly associated with better analyst effectiveness during requirements elicitation, should be straightforward to observe empirically.

2.2 Studies about Requirements Engineering Experience

As far as we know, there are seven empirical studies (Marakas and Elam [4], Niknafs and Berry [5], [17], Agarwal and Tanniru [44], Pitts and Browne [9], Hadar et al. [45], and Ferrari et al. [46]) focusing on the relationship between analyst experience and effectiveness during requirements elicitation.

2.2.1 Pitts and Browne

Pitts and Browne [9] designed an experiment in the information systems field examining the use of cognitive stop rules in order to measure sufficiency, or the point at which the acquired requirements are sufficient for system development to continue.

A total of 54 professional analysts with at least two years of experience in systems development participated in the experiment. The average number of years of experience was 11. It is reasonable to assume that the subjects are mature enough to perform requirements elicitation effectively, i.e.,

according to some authors such as Campbell and Bello [34], they achieved the expert level.

Pitts and Browne analysed the influence of experience, measured as number of years, in terms of number, breadth and depth of elicited requirements. As a result, in our view, they reported that analyst experience does not influence requirements determination; that is, the number, breadth and depth of requirements does not depend on the number of years of analyst experience.

2.2.2 Marakas et al.

Marakas and Elam [4] designed and ran a controlled experiment in the information systems field with the aim of evaluating the effectiveness of the semantic interview technique (a type of semi-structured interview) against the non-structured interviews.

A total of 20 inexperienced and experienced subjects participated in the experiment. Experience was measured as the number of years that the subject had worked on systems analysis and software development. Inexperienced subjects were final-year MSc in Software Engineering students, whereas experienced subjects were professional systems analysts and software developers.

The experimenters managed to identify differences in the effectiveness of the two interview types. Specifically, subjects were more effective using the semantic interview. However, irrespective of the interview type used, experienced subjects were only marginally better identifying requirements (around 3% more requirements) than novice subjects, where the differences are nowhere near statistically significant.

2.2.3 Agarwal and Tanniru

Agarwal and Tanniru [6] conducted a controlled experiment in the expert systems field in order to compare the effectiveness of the structured and non-structured interviews in terms of the number of extracted business rules, including other measures. These rules are used to specify relationships between concepts that influence the decision under consideration. An example of a rule is "positive balance is when the benefit is higher than 1000€".

A total of 30 subjects with different levels of experience –novices and experienced– participated in the experiment. The novice subjects were postgraduate students with similar backgrounds and job experience, whereas the experienced subjects were either:

- Knowledge engineering professionals with job experience working on at least one expert system, or
- Analysts with at least three years of systems analysis experience.

The researchers reported that:

- Experienced subjects, which used unstructured interviews, performed slightly better (by about 9%) than inexperienced subjects, which used unstructured interviews. However, the differences were not statistically significant.
- Novice subjects, which used structured interviews, achieved better results than other novice and experienced subjects, which used unstructured interviews. The differences were statistically significant.

2.2.4 Niknafs and Berry

Niknafs and Berry [5] conducted a controlled experiment to empirically study the impact of domain knowledge and requirements experience on requirements elicitation effectiveness measured in terms of the number of generated ideas. The main focus is on the analysis of whether the lack or presence of domain knowledge in analysts affects the effectiveness of their requirements elicitation, considering analysts experience as a secondary analysis. Unlike previous research, the experimental subjects used brainstorming as the elicitation technique.

A total of 19 groups participated in the experiment. Each group was composed of three undergraduate subjects with different levels of development and requirements specification experience. Experience was measured in number of years. The experimenters argue that although it would make sense for there to be a positive relationship between experience and requirements elicitation effectiveness, the trends that they observed suggest quite the contrary, that is, subjects with one and two years of experience were slightly less effective than inexperienced subjects, whereas the effectiveness in groups with more than two years of experience dropped sharply.

Niknafs and Berry [17] reported two controlled experiments (E1 and E2) with computer science and software engineering students, E2 being an exact internal replication of E1 published in 2012 [5] The aim of E2 is to increase the sample size recruited in E1. A total of 40 groups participated, each with three members and differing levels of professional and requirements experience of up to four years. E2 was analysed together with E1. The results show that the less Computer Science (CS) or Software Engineering (SE) education a group had, the more high experience helped the group to be more effective in brainstorming. Moreover, the more CS or SE education a group had, the more low experience helped the group to be more effective in brainstorming.

2.2.5 Hadar et al.

Hadar et al. [45] conducted an empirical study to analyse domain knowledge's positive and negative effects on the interview process. The experiment consists of one baseline with 27 subjects and one replication with 31 subjects. Subjects were assigned to high and low domain knowledge groups. The study also sought differences in the interviews conducted by analysts with and without domain knowledge. The research found that domain knowledge supports the communication between the analyst and the stakeholders. The completeness and correctness of the elicited requirements are positively affected by domain knowledge. However, some analysts with sound domain knowledge achieve suboptimal results because, being overconfident or time-constrained, they did not ask questions which seem apparent.

2.2.6 Ferrari et al.

Ferrari et al. [46] conducted a quasi-experiment with 43 subjects to evaluate an approach for doing requirements elicitation interviews combining role-playing, peer-review and self-assessment named SAPEER. The approach consists of letting students perform a role-playing interview and then

stimulate learning through reflection by asking students to identify mistakes in their own interview and in the interview of their peers. The acquired competence is then tested in a second interview. The study is under the context of a teaching activity where students must reflect on their mistakes and improve their interview skills.

Results yield that major reductions are observed for mistakes that can be corrected with well-defined actions. For example, providing a summary at the end of the interview or asking probing questions. Mistakes related to behavioural aspects are harder to correct, and some mistakes in question omission are not correctly addressed.

2.3 Studies of Software Engineering Experience

Although there is a very widespread belief in SE that experience improves analyst effectiveness [47], [15], [16], [48], it is a fact that existing empirical studies [4], [5], [6], [9], [17] have failed to confirm this belief, sometimes even reporting results to the contrary. The aim of this paper is to verify whether requirements experience influences analyst effectiveness during requirements elicitation using rigid question-answer-based interviews. To do this, we apply an empirical approach along the lines of the related work. Section 3 describes the study design.

3 Methodology

This section reports the experimental design, according to the SE experiment reporting guidelines proposed by Jedlitschka and Pfahl [49].

3.1 Hypothesis

We state the following research hypothesis based on its respective null (H_{0i}) and alternative (H_{1i}) hypotheses:

H_{0i}: There is no relationship between experience and elicitation analyst effectiveness.

H_{1i}: There is a relationship between experience and elicitation analyst effectiveness.

i being one of the experience types defined below in Section 3.2. As our study is exploratory and the literature has reported different trends with respect to the effect of experience, we cannot anticipate the direction of the effects. Therefore, the alternative hypothesis is two-tailed.

3.2 Independent Variables

The independent variable refers to requirements analyst *Experience*, as shown in Table 1. However, it is immediately clear that the concept of "experience" is ambiguous; there is more than one type of experience: interview, requirements, etc. It would not be very precise to consider just professional experience, subjects may have carried out many different activities throughout their career. On this ground, we decided to consider other types of experience apart from **professional experience**. (1) Interview experience is the number of years that the subject has participated in interviews to elicit requirements in industry; (2) Requirements experience is the number of years that the subject has participated in requirements specification activities or requirements validation activities in industry; (3) Development experience is the number of years that the subject has developed software in industry; (4) Professional experience is the number of years that the subject has been working in industry. Note that independent variables are not independent of each other (e.g., development experience could include requirements experience), so some of them could be confounded. This is not a problem for analysing them separately. The metrics of these variables are applied by the experimenters with no option to change it (they depend on the subject's background). Apart from experience, other factors may affect the elicitation process [50], such as creativity, personality, psychology, etc. These possible factors are out of the scope of the article, and their effect should disappear as soon as the sample size is larger.

TABLE 1
Independent Variables

Independent Variable	Metric
• Interview experience • Requirements experience • Development experience • Professional experience	Number of years of experience in industry of each type

Based on the related empirical literature, we specifically chose *requirements* ([6], [17]) and *development* ([9], [6]) *experience*.. We believe that it is important to also consider *interview experience*, because, as specified in Section 3.5, elicitation sessions were carried out by means of interviews, on which ground the subject's interview experience could influence their effectiveness.

3.3 Dependent Variables

The dependent variable is elicitation analyst *Effectiveness*. There is as yet no widely accepted metric for measuring requirements elicitation effectiveness. It is possible, however, to find several alternative measurements in the literature and existing empirical papers, where effectiveness is measured as the total number of identified rules that specify relationships between concepts that influence the decision under consideration [6]; according to the number of elicited requirements [9]; taking into account the elicited rules and clauses [51]; by dividing effectiveness into different categories [52]: total number of requirements, business processes and information necessary to inform tasks behaviours; and according to the number of ideas generated by the RE team [5].

Therefore, effectiveness has traditionally been operationalized as the number of items (irrespective of whether they are concepts, rules, processes, etc.) elicited by the analyst during elicitation, as it is reflected in a systematic review of requirements elicitation techniques [53]. In this case, we applied a similar procedure to the abovementioned researchers and measured requirements elicitation analyst *Effectiveness* as the *percentage of all problem domain items identified by the experimental subjects*. Experimenters have a gold standard used to measure such percentage (Appendix C, available online.

The problem domain is composed of different types of items. There is agreement in the literature on the essential items, such as objectives, processes and tasks [54],

requirements, functions and states [55], concepts, actions or rules [56], etc. Therefore, we used three key types of items to measure effectiveness: concepts, processes and requirements. We did not take into account domain details; for example, process inputs and outputs, or conceptual model attributes and relations. This is because, as explained later, the experimental task is a short interview in which analysts would find it difficult to appreciate such details.

We did not study more sophisticated aspects, like rules, objectives and stakeholders, either, as the selected domains are rather simple with few stakeholders and no business rules. On the same grounds, non-functional requirements were excluded from the study. This has the advantage of making the experiment easier to instrument.

The experiment focuses on eliciting requirements for information systems. Other types of systems, e.g., control systems, do not share the peculiarities of information systems. In these contexts, a better option would be the characterization by Gunter et al. [57], which regards requirements rather than specifications as operational declarations on a domain.

The problem domain items considered in this research are: concepts, processes and requirements. The list of concepts, processes, and requirements of the problems used in the experiment can be seen in Appendix C, available in the online supplemental material. The items were identified by one experimenter using written reports provided by the experimental subjects. The effectiveness measure is calculated according to the following formula:

$$Effectiveness = \frac{\#\ identified\ concepts + \#\ identified\ processes + \#\ identified\ requierements}{total\ number\ of\ items}$$

We opted not to calculate a weighted mean, as the fact that there is a greater percentage of a particular item in a domain does not mean that it is intrinsically more important. All the items play an important role in the construction of the future software system. This formula does not discriminate the items' granularity levels, it adds everything together. The analyses on a per item basis are available in Appendix H, available in the online supplemental material. The values of this formula come from a report that subjects prepare after the interviews. After checking this report and comparing it with the gold standard, the experimenter can calculate a metric of effectiveness for each subject.

3.4 Subject Selection

We used convenience sampling to select the experimental subjects. The subjects that participated in the quasi-experiments were recruited from among both Requirement Engineering students (53) enrolled in the Master in Software Engineering at the Universidad Politécnica de Madrid (UPM), and 21 participants in the 2013 edition of the Alive Empirical Study at the International Working Conference on Requirements Engineering (REFSQ). Most experimental subjects (117 out of 124) have a computing or similar background. Note that the metric of experience we are using is based on the years of experience that subjects have in industry. So, the fact of recruiting subjects from an academic context is not a key threat since subjects with no experience are classified as 0 years of experience in the analysis. Existing works, such as Pacheco et al. [53], highlights that most of the existing requirements elicitation methods (95% from a sample of 194 studies) have been validated using case studies. So, the recruitment of so many subjects in an empirical experiment (124) as this paper proposes is a step forward to cover this gap. There are also previous works that have reported the benefits of recruiting students rather than practitioners [58].

3.5 Experimental Task

Before starting the experimental task, experimenters acting as clients reached a consensus on the requirements gold standard. Two days before starting the experimental tasks, the experimenters learned the gold standard to conduct the interviews.

The experimental task was composed of three main phases: 1) elicitation session, 2) information reporting, 3) post-experimental questionnaire response. During the elicitation session, the subjects played the role of requirements analysts (interviewer), and three researchers acted as clients (interviewees).

The elicitation session was conducted by means of an open interview, which is an open conversation as is common in the early stages of the requirements process within a set time (30 or 60 minutes, depending on the case). We believe that the allocated times are sufficient to acquire a substantial amount of information about the problem domain. Clients answered questions without hiding any details, releasing as much information as possible, and they cannot lie. To avoid the fatigue effect, clients could not participate in more than 5 interviews per day, and each interview could not spend more than 30 minutes.

At the end of the elicitation session, the experimental subjects submitted a written report about all the information that they acquired during the interview. There was no set reporting format, and they were given up to 90 minutes to complete the report. We observed that the experimental subjects tended to report a list of items, often divided into sections (e.g., functional requirements, non-functional requirements, etc.) and even managing to use conceptual models. Therefore, we decided to allow subjects to use their preferred reporting format instead of requiring template use.

At the end of the experiment and after submitting their final report, the subjects completed the post-experimental questionnaire, which took fewer than five minutes. All the experimental package including questionnaires and problems can be seen in a Zenodo repository [59].

3.6 Assignment of Subjects to Treatments

We used a quasi-experimental design. Quasi-experiments are carried out when the subjects cannot be assigned at random to an experimental condition or, alternatively, a treatment cannot be assigned to a group. This is applicable here, since the experience is a built-in characteristic of the experimental subjects that cannot be randomized or blocked.

Therefore, subjects are not, strictly speaking, assigned to treatments in this study, because they all perform the same treatment, that is, all the subjects participating in the quasi-experiment perform requirements elicitation on the same

TABLE 2
Problem Domains Used in the Experiment

Problem	Brief Description	Type
UP1	Battery recycling machine control system.	Unfamiliar
FP1	Text messaging system.	Familiar

problem and with the same interviewee. Note that even though there are three interviewees, each subject interacts with only one of them.

3.7 Experimental Objects

In this research, we used two problem domains, as shown in Table 2. In the first case, we used a domain with which the analysts were unfamiliar, that is, the selected problem is so uncommon that most experimental subjects are unlikely to have any exposure to the issue. This rules out an uncontrolled *Experience x Knowledge* interaction.[1] The unfamiliar problem domain (UP1) is related to a battery recycling plant, where a series of domain-specific machines perform several peculiar processes practically impossible to infer unless one has first-hand knowledge of such domain. The problem is based on a simplified real system to assure that the subjects can address the problem in the limited time available for the experimental sessions.

The familiar problem domain (FP1) is an instant messaging system by means of which the user will be able to perform operations like flat text messaging, contact management, etc. In this case an *Experience x Knowledge* interaction is possible. To analyse this interaction we need to compare results of UP1 (unfamiliar) versus results of FP1 (familiar).

Both problem domains were described to subjects in full according to three types of items by which they are defined: requirements, concepts and processes as shown in Appendix C, available in the online supplemental material. These descriptions are a checklist of requirements that can be used to establish whether subjects identify the problem domain items during the elicitation session. We use the checklist to interpret which items the subject has identified and reported. So, problem description can be also used as a yardstick for measuring the effectiveness of the experimental subjects. Only items specified in that list are considered correct. Table 3 shows the total number of items defining the size of the problem domain. Note that we tried to assure that the total number of problem items was similar. This was intended to make sure that the *Problem* was not another variable possibly influencing the results, as, otherwise, we would have an unwanted *Knowledge x Size* interaction.

3.8 Measurement Procedure

The independent variables were gathered by means of a post-experimental questionnaire. It was implemented within the *Moodle* and *Google Forms* platforms. The aim of the questions was to gather information related to the experimental subject: years of experience, knowledge on problem domains and training, etc. Note that by years of experience we mean experience in real industry (not as student). The questionnaire used for all subjects is available in Appendix B, available in the online supplemental material.

The dependent variable, elicitation analyst effectiveness, was measured according to the reports submitted by the experimental subjects at the end of the reporting process. The items defining the problem domain (requirements, concepts and processes) were used as a checklist to measure effectiveness. Multiple repetitions of the same item in reports submitted by subjects were not taken into account to calculate effectiveness (they were counted once). Each quasi-experiment was measured by a single researcher. This could lead to some bias in the results and is considered as a possible validity threat in Section 6.

TABLE 3
Total Number of Items for Each Problem Domain

Problem	Items definning the Prblem (#)			
	Requirements	Concepts	Process	Total
UP1	15	24	12	51
FP1	28	10	16	54

3.9 Analysis Strategy

Typical inference tests (t-test, ANOVA, etc.) cannot be applied in the analysis of the quasi-experiments, as such tests cannot use scalar independent variables. One alternative option is an ANCOVA using the domain type as a between factor and experience as a covariable. However, as mentioned later in Section 4.4, groups are unbalanced, and one of the data sets is possibly heteroscedastic, which advises against the use of ANCOVA. We could use mixed models, but we would have to assume a particular covariance matrix structure, which is more complicated to interpret.

We have decided to use multiple linear regression (MLR). This statistical technique is appropriate because: a) it can account for scalar independent variables, and b) it is capable of jointly calculating the effects of several independent variables. For MLR to be reliable, we have to test for four conditions: collinearity, sampling size, normality and homoscedasticity.

- *Collinearity.* For the analysis to be reliable, it is necessary to assure that the model predictor variables are not collinear. Collinearity occurs in MLR when one or more independent variables are linearly correlated with other model variables. To check for collinearity between variables, we used the variance inflation factor (VIF), tolerance (T) and condition index (CI):
  - One recommendation often used by researchers [60] is to consider a large VIF, that is, VIF > 10, yielded if $R^2 > 0.9$ and $T < 0.1$, as evidence of collinearity. A second more rigorous option is to reduce the VIF limits to VIF > 5 with $R^2 > 0.8$ and $T < 0.2$ [61].
  - *Condition index (CI):* Belsley [62] suggests three degrees of collinearity: slight (CI < 10), moderate

1. Notice that the impact of problem domain knowledge is not one of the research goals of this paper; we have already reported about this issue in [66]. Our intention behind the separation of the *Experience* and *Knowledge* variables is to avoid the threat of previous problem domain knowledge having an influence on analyst effectiveness that is confounded with the experience effect.

TABLE 4
Total Number of Items for Each Problem Domain

Effect	Large	Moderate	Low	Zero
Coefficient (B)	$\pm$ (> = 3)	$\pm$ [2 - 3)	$\pm$ [1 - 2)	$\pm$ [0 - 1)

(10 < CI < 30) and severe (CI $\geq$ 30). When a model has a severe condition index, one or more variables have shared variance with the other variables. Usually, a variable with a high variance proportion (greater than 0.5) is considered to be involved in collinearity.

- *Normality*. The distribution of residuals must be normal with a mean of zero and random but constant variance. We used the Kolmogorov-Smirnov test with the Lillierfors correction and the Shapiro Wilks test to check for normality of residuals.
- *Homoscedasticity*. We checked for homogeneity of variance using scatter plots of the standardized predicted values against model residuals.

We interpret the effect of experience using the non-standardized coefficient (B). B indicates the mean change for the dependent variable against each unit of change of the independent variable, that is, the resulting Bs represent increases (or decreases) in effectiveness by year of experience. Bs are reported using the same unit as the response variable *Effectiveness*, that is, percentages. For example, five years of experience for a B = 1.543 is equivalent to an effectiveness increase of $1.543 \times 5 = 7.7\%$. Unlike effect sizes, we have no benchmarks for determining whether a particular B is equivalent to a large, medium or small effect. On this ground, we will set a criterion for interpreting the Bs as follows in Table 4.

3.10 Sample Size Estimation

To be able to rigorously apply a MLR, there should be a minimum sample size. Otherwise, the estimation of the effect of the independent variables will be less precise, the likelihood of detecting significant effects will be lower, and there may even be an overfitting phenomenon. Overfitting occurs when the model provides too exact a correspondence with the data set because there are too many independent variables for the number of cases. According to Miles and Shevlin [63] (cited in Field et al. [64]), 90 subjects are enough to test a MLR with four independent variables, assuming medium effect sizes. If the effects were large, 40 subjects would suffice. Note that in our case we have 124 subjects, considerable larger than the minimum number recommended.

4 Family of Quasi-Experiments

4.1 Quasi-Experiments

We conducted 12 quasi-experiments: eight on the unfamiliar problem domain (UP1) and four on the familiar problem domain (FP1). Consistency among the different replications is ensured since experimenters are the same in each one. The procedure was applied in the same way in all of them.

Table 5 shows the quasi-experiments conducted using the UP1 problem domain. For each quasi-experiment, it shows the type of replication used, the execution site and the number of subjects participating in each replication. Q-2007 was the baseline experiment, which we replicated seven times: six replications at UPM and one at REFSQ. We collected data about 88 experimental subjects. Note that even though the baseline was conducted with UP1, some replications were conducted using FP1 to analyse the effect of domain familiarity. These small changes can be considered in a family of experiments [65].

TABLE 5
Family of Empirical Studies About Requirements Elicitation – UP1

#	Quasi-Experiment	Replication Type	Site	Subjects
1	Q-2007-UP1	Baseline experiment	UPM	7
2	Q-2009-UP1	Internal replication	UPM	8
3	Q-2011-UP1	Internal replication	UPM	16
4	Q-2012-REFSQ-UP1	External replication of Q-2011	REFSQ	21
5	Q-2012-UP1	Internal replication	UPM	14
6	Q-2013-UP1	Internal replication	UPM	8
7	Q-2014-UP1	Internal replication	UPM	9
8	Q-2015-UP1	Internal replication	UPM	5
TOTAL		8 empirical studies	UPM / REFSQ	88

TABLE 6
Family of Empirical Studies About Requirements Elicitation – FP1

#	Quasi-Experiment	Replication Type	Site	Subjects
9	Q-2012-FP1	Internal replication	UPM	14
10	Q-2013-FP1	Internal replication	UPM	7
11	Q-2014-FP1	Internal replication	UPM	7
12	Q-2015-FP1	Internal replication	UPM	8
TOTAL		4 empirical studies	UPM / REFSQ	36

To study the effect of experience in the familiar domain, we executed four quasi-experiments with a total of 36 experimental subjects, shown in Table 6. Comparing Tables 5 and 6 it can be easily noticed that the research started in the unfamiliar domain (quasi-experiment #1 dates back to 2007), and only much later (2012) progressed to the familiar domain.

Our initial intention was to assess the experience effect independently of the problem domain knowledge effect (see Section 3 for details). However, after conducting quasi-experiments #1-4, we observed that experience did not seem to make any influence on analyst effectiveness (see Section 4.4).

Accordingly, we launched a series of quasi-experiments in a familiar domain, aiming to find out if the interaction *Experience x Knowledge* could explain the widespread belief in the positive effect of experience. The research finished when we approached the required sample sizes (see Section 3.10) for both domains, and the effects became apparent.

As shown in Table 6, quasi-experiments differed with respect to resource availability and contextual issues throughout the research. These differences have to be taken into account, as they can have a moderator effect on analyst effectiveness, that is, increase or decrease their effectiveness. For example, as Table 6 shows, the subjects in Q-2007 conducted the elicitation using the individual open interview as an elicitation technique, whereas the group interview (modelled on a

TABLE 7
Quasi-Experiment Characteristics

Experiment	Interview type	Language	Interviewee	Elicitation time	Execution	Warming-up
Q-2007	Individual	Spanish	OD	30 min	At the end of the course	16 weeks
Q-2009	Individual	English	AG	30 min	At the end of the course	16 weeks
Q-2011	Group	English	OD	60 min	At the end of the course	16 weeks
Q-2012 REFSQ	Group	English	OD	60 min	-	No warming up
Q-2012	Individual	English Spanish	OD / JWC	30 min	At the start of the course	No warming up
Q-2013	Individual	English Spanish	OD / JWC	30 min	At the start of the course	Warming up 1 week
Q-2014	Individual	English	OD	30 min	At the start of the course	Warming up 6 weeks
Q-2015	Individual	English Spanish	OD / JWC	30 min	At the start of the course	Warming up 2 weeks

requirements workshop) was used in Q-2011. It is evident that the interview type (individual, group) could influence the effectiveness achieved by analysts. Experimenters played the role of interviewees, so we can ensure that subjects applied the interview type they had assigned.

The contextual variables that could act as moderator variables between the analyst effectiveness and the years of experience are:

- *Interview Type:* individual open interview (1:1 interview) or group interview (1: N interview, where several analysts interview a client simultaneously).
- *Interviewee:* person who acts as the client during the requirements elicitation process. Three researchers played the role of client: OD (Oscar Dieste), AG (Anna Grimán) and JWC (John W. Castro).
- *Language:* some quasi-experiments are blocked by language (that is, one group held the interview in Spanish and another in English) and by interviewee. By blocking the subjects by language/interviewee, we increased the quality of the conversation. For example, a non-Spanish speaker could not communicate well enough in Spanish with a Spanish-speaking interviewee.
- *Elicitation Time:* interview duration. The interviews lasted at most 30 minutes and 60 minutes for individual and group interviews, respectively.

TABLE 8
Characterization of Subjects Depending on Experience

Family of Quasi-experiments			
Characte-ristic	Level	#UP1 subjects	#FP1 subjects
Academic Education	NCS	7	0
	CS	80	34
Interview Experience	Novices (0-1 year)	53	24
	Intermediate Level (2-4 years)	17	6
	Experts (> = 5 years)	13	1
Requirements Experience	Novices (0-1 year)	44	21
	Intermediate Level (2-4 years)	22	9
	Experts (> = 5 years)	17	2
Development Experience	Novices (0-1 year)	13	8
	Intermediate Level (2-4 years)	30	15
	Experts (> = 5 years)	24	6
Professional Experience	Novices (0-1 year)	13	9
	Intermediate Level (2-4 years)	20	9
	Experts (> = 5 years)	38	9

NCS: Non-computerscience, CS: Computer science.

- *Execution:* whether subjects participated in the quasi-experiment before or after the Requirements Engineering course.
- *Warming-Up:* short training course in requirements-related activities. Warming-up refers to the duration of the training, ranging, in this case, from 0 to 6 weeks, whereas the above *Execution* moderator variable is essentially a binary variable (with *before* and *after* values).

4.2 Data Collection

4.2.1 Demographic Data

Table 8 summarizes the key demographic data of the sample. We categorized the experience in industry of the subjects at three levels: novices (0-1 year), intermediate (2-4 years) and experts (5 years) [34], [35], [36].

We found that the distribution of experience for UP1 is reasonably balanced among novices, intermediate levels and experts. This does not apply for FP1, and unbalance may have an impact on the reliability of the results for the familiar domain, as specified in the validity threats. For a more thorough examination of the distribution of experience, Appendix D, available in the online supplemental material, provides bar charts by domain and experience type.

4.2.2 Descriptive Statistics

Table 9 shows the key descriptive statistics for effectiveness. There are no clear observable trends, as averages follow a sawtooth pattern. Exceptionally, there appears to be an increase in the effectiveness of the subjects depending on interview experience and a decrease depending on professional experience in both UP1 and FP1. The experimental data are available in http://grise.upm.es/sites/extras/15.

4.2.3 Aggregating Data Using Moderator Variables

We propose conducting the statistical analysis aggregating data of the different replications using moderator variables: interview type; interviewee; language; elicitation time; execution; warming-up. Usually, these variables would have been ignored in the analysis and would have appeared as threats to validity. We have identified these variables explicitly and we have estimated their effect since we explore different levels in different replications. This way we have avoided their effect on data. Language and elicitation time have been discarded because they are confounded with interviewee and interview type.

The possible effects of the moderator variables should be accounted for in order to conduct a more accurate joint

TABLE 9
Descriptive Statistics for Effectiveness

Characte-ristic	Level	UP1 #Sub	UP1 Mean	UP1 Std. Dev	FP1 #SuB	FP1 Mean	FP1 Std. Dev
Academic	NCS	7	35.29	14.32	0		
Education	CS	80	42.60	16.44	34	27.09	16.04
Interview	0-1 year	53	44.25	16.92	24	33.02	18.60
Experience	2-4 yrs	17	36.22	18.47	6	39.20	19.01
	>= 5 yrs	13	42.68	9.38	1	55.56	.
Requirements	0-1 year	44	44.30	15.40	21	36.11	21.75
Experience	2-4 yrs	22	38.68	19.57	9	37.65	20.47
	>= 5 yrs	17	41.64	14.33	2	30.56	18.32
Development	0-1 year	13	40.27	15.55	8	36.11	21.75
Experience	2-4 yrs	30	44.58	18.11	15	37.65	20.47
	>= 5 yrs	24	42.57	14.60	6	30.56	18.32
Professional	0-1 year	13	46.15	11.92	9	36.42	19.49
Experience	2-4 yrs	20	47.65	17.71	9	40.95	24.58
	>= 5 yrs	38	40.35	16.24	9	34.98	15.20

NCS: Non-computerscience, CS: Computer science, SUB: Subjects.

TABLE 10
Percentage Adjustment (Effect) for each Potential Moderator Variable

Moderator Variable	Levels	% Adjustment for UP1	% Adjustment for FP1
Interview type	Individual Group	11%	NA
Execution	Before After	12%	NA
Interviewee	OD AG	23%	NA
	OD JWC	18%	23%
Warming Up	0 week	0%	0%
	1 week	-1%	-6%
	2 week	7%	0%
	6 week	10%	5%

analysis of the data from the different quasi-experiments. For example, imagine that we have two subjects, one non-computer scientist (A) and one computer scientist (B). Apart from the job, the other difference between A and B is that B is familiar with the application domain. Likewise, this knowledge acts as a moderator variable and increases his or her effectiveness by 10 points. If we were to analyse differences in effectiveness between both subjects by omitting the knowledge effect, the relationship would be positive. This result is incorrect because the moderator variable (knowledge) is what really makes subject B more effective. The effect of this variable is confounded with the effect of the real independent variable (job), leading to the mistake in the estimation of the correlation coefficient.

The best way to account for the effects of moderator variables would be to add them as interaction terms to the MLR model discussed in Section 4.3. However, this strategy could be troublesome for two main reasons:

1) Each new variable entered in a MLR model increases the sample size required to achieve reasonable statistical power. We have six contextual variables presented in Section 4.1: Interview type, interviewee, language, elicitation time, execution and warming up, although two of them have a 1:1 relationship (Interviewee - Language and Interview type - Elicitation time). Added to the four independent variables (the four types of experience: interviews, requirements, development, professional), it makes a model containing 8 variables. The required sample size to detect medium effect sizes would be around 125 subjects, according to Miles and Shevlin [63]. So, even though we have 124 subjects, we have not enough sample to analyse unfamiliar domain and familiar domain separately.
2) From the viewpoint of the regression model, moderator variables are not different from independent variables. On this ground, orthogonality (all independent variables are uncorrelated) and a reasonable balance (similar number of observations) between combinations should exist in order the MLR be accurate. As the contextual variables in this research emerge on the grounds of contextual restrictions rather than design considerations, orthogonality and balance do not hold.

To solve both problems, we estimated, based on the available studies, different values for the moderator variables. We then averaged these values depending on the sample size of each study (as in meta-analysis, where sample size acts as a proxy of study reliability). We are reasonably sure that the applied adjustment procedure is reliable. Sometimes, the effect of a moderator variable can be estimated by means of a statistical model.

When this happens, the results of both procedures (statistical model, weighted averages) are very similar. For example, the interviewee effect was obtained using a mixed linear model in one particular study [66]; this value was 26%. The same interviewee effect after applying the weighted averages ranged from 18 to 23%. Therefore, the values are quite similar to each other.

In this manner, the influence of the moderator variables can be eliminated by subtracting the estimated effect of each variable on the affected subjects. Similarly, the effectiveness of subjects A and B in the above example (see Fig. 1b) is similar if we subtract the effect of the moderator variable for subject B (i.e., 10 points). This is what we would expect if we assume that there is no difference between the two. Since there were some differences among replications, such as period of time where the experiment was conducted or language, we adjusted the data for the aggregation. The data adjustment values for each moderator variable are specified in Table 10. The data adjustment procedure, as well as the details of the calculations are specified in Appendix E, available in the online supplemental material. The data adjustment does not change the patterns shown in Table 9. Results of the adjusted analysis and results of non-adjusted analysis agree. This means that the adjustment does not involve any non-significant result into significant or vice versa.

4.2.4 Data Set Reduction

For MLR, data must be available on all the experience types (interview, requirements, development, profession-nal) for all subjects. As Table 8 shows, a number of subjects did not

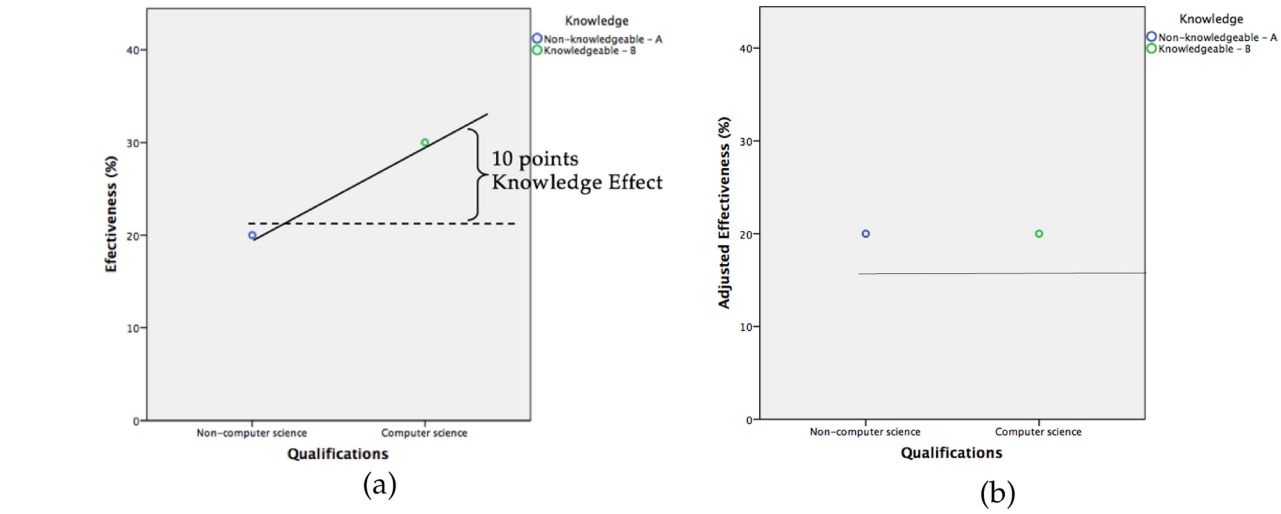

Fig. 1. Fictitious example of the effect of a moderator variable.

provide information on at least one of the types of experience. To apply MLR, these subjects must be removed. This means that:

- *Unfamiliar domain:* We recruited 88 experimental subjects, of which only 69 could be used.
- *Familiar domain:* Of 36 participant subjects, only 29 could be used.

Data loss can be reduced using fewer variables in the MLR. For example, if we take development experience out of the analysis, the usable data are greater for UP1 (76 subjects) and constant (29 subjects) for FP1. We did not adopt this approach as the models with three or four variables essentially yield the same results, as shown in Appendix F, available in the online supplemental material.

4.3 Testing of the Necessary Conditions for Applying MLR

As we studied the effect of experience on two different problem domains, the conditions of MLR were tested depending on each domain.

4.3.1 Unfamiliar Problem Domain

The MLR for studying the effect of different types of experience is composed of the four independent variables under study, namely, interview experience, requirements experience, development experience and professional experience.

$Effectiveness = \boldsymbol{\beta}_0 + \boldsymbol{\beta}_1\ InterExp + \boldsymbol{\beta}_2\ ReqExp + \boldsymbol{\beta}_3\ DevExp + \boldsymbol{\beta}_4\ ProfExp + \boldsymbol{\varepsilon}.$

The proposed model meets all the required assumptions (see Section 3.9) for applicability:

- *Non-Collinearity:* Collinearity statistics are within the specified ranges (VIF $<$ 10 and CI $<$ 10), as shown in Section 4.4.1, Table 11 and Appendix G.1, available in the online supplemental material, Table 9, respectively.
- *Normality:* The distribution of the model residuals is normal (p-value $= 0.200 > 0.05$ for the Kolmogorov-Smirnov test and p-value $= 0.243 > 0.05$ for the Shapiro-Wilk test). Data normality is confirmed by means of skewness (-0.538) and kurtosis (0.233) statistics, as they are within the usual ranges ± 1.
- *Homoscedasticity:* The observed variance is quite uniform across the range of typified residuals, as shown in Fig. 2. No bottleneck patterns are observed. The sample size is lower than the desired value. The model has four independent variables, on which ground at least 90 subjects are required to detect a medium-sized effect. In this case, the sample size ($n = 69$) is not large enough; however, it comes very close and it is sufficient to detect medium-to-large effect sizes (see Section 3.10). The analysis results (see Section 4.4.1) point out in the direction that the low sample size is not affecting negatively the conclusions of this research. In any case, the low sample size is listed as a potential threat to validity.

4.3.2 Familiar Problem Domain

The MLR for the familiar domain is calculated in the same way as above. The data compliance with the applicability conditions is reasonable:

- *Non-Collinearity:* The collinearity statistics are within the established ranges (VIF $<$ 10 and CI $<$ 10), as specified in Section 4.4.2, Table 12 and Appendix

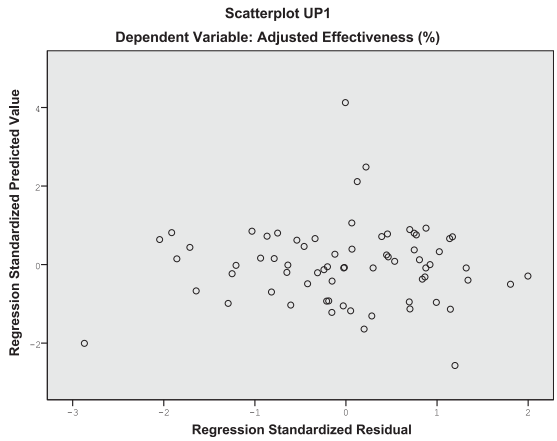

Fig. 2. Scatter plot of the residuals of the UP1 model.

TABLE 11
Effect of Experience for the UP1 – MRL

Model	Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Collinearity Statistics	
	B	Std. Error	Beta			Tolerance	VIF
1 (Constant)	28.730	2.543		11.297	.000		
Interview Experience (years)	.532	.785	.149	.678	.500	.318	3.145
Requirements Experience (years)	.271	.717	.102	.378	.707	.209	4.787
Development Experience (years)	.231	.524	.072	.441	.661	.574	1.742
Professional Experience (years)	-.540	.606	-.235	-.891	.376	.221	4.535

Dependent Variable: Adjusted Effectiveness (%) $R^2 = .020$; Real N = 69; Required N = 82

TABLE 12
Effect of Experience for the FP1 – MRL

Model	Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Collinearity Statistics	
	B	Std. Error	Beta			Tolerance	VIF
1 (Constant)	29.247	5.823		5.023	.000		
Interview Experience (years)	4.429	2.424	.376	1.827	.080	.773	1.294
Requirements Experience (years)	2.868	2.542	.244	1.128	.270	.698	1.432
Development Experience (years)	-.619	2.050	-.071	-.302	.765	.597	1.674
Professional Experience (years)	-2.556	1.383	-.421	-1.848	.077	.631	1.585

Dependent Variable: Adjusted Effectiveness (%) $R^2 = .214$; Real N = 29; Required N =82

G.2, available in the online supplemental material, Table 10, respectively.

- *Normality*. The model residuals have a normal distribution (p-value = .200 > 0.05 for Kolmogorov-Smirnov test and p-value = .453 > 0.05 for Shapiro-Wilks test). Data normality is confirmed by means of skewness statistics (.623) and kurtosis (.580), as they are within the usual ranges ±1.
- *Homoscedasticity*. As not many data are available, the patterns shown in Fig. 3 are not well-defined. On this ground, we cannot state or reject the homogeneity of variance. The missing information is discussed in the validity threats.

As in the unfamiliar domain, the sample size is not enough. The model has four independent variables, on which ground at least 40 cases are required to detect large effect sizes. In this case, we have n = 29 subjects, which is close to 40. The MLR reported here will only be able to detect large effects with a power of 80% and $\alpha = 0.10$ (calculations made using G*Power).

4.4 Results

The results of MLR used to determine the effect of subject experience on the elicitation analyst effectiveness are shown in Table 11 and Table 12 respectively.

4.4.1 Effect of unfamiliar problem domain experience

As Table 11 shows, interview experience ($B_1 = .532$), requirements experience ($B_2 = .271$) and development experience ($B_3 = .231$) tend to have a **positive effect** on effectiveness. On the other hand, professional experience ($B_4 = -.540$) has a **negative effect**. According to the interpretation of the non-standardized coefficients (B) established in Section 3.9, the effects in all cases are **zero**. Additionally, none of the effects are **significant** (p-value > 0.05).

Apart from the observed effects, two issues related to the MLR model should be stressed:

1. The coefficient of determination R^2 (degree of fit) is very low ($R^2 = .020 = 2\%$).
2. The model is not significant (p-value = .860).

This means that the **independent variables (the different types of experience) bear hardly any or no relationship at all to analyst effectiveness.**

4.4.2 Effect of familiar problem domain experience

The regression model results are shown in Table 12. We find that *interview experience* ($B_1 = 4.429$) has *a strong positive effect*, whereas professional experience ($B_4 = -2.556$), on the other hand, has a *moderate negative effect*. Both cases are very close to statistical significance (p-value = .080 and p-value = .077, respectively), which is notable in view of the limited number of data available (n = 29) for this domain.

Requirements experience ($B_2 = 2.868$) and *development experience* ($B_3 = -.619$) tend to have a positive and negative effect, respectively, although neither is statistically significant. We believe that requirements experience could have some influence on analyst effectiveness, as the p-value, .270, is rather low and the model is not statistically powerful. However, this is just a hypothesis.

The coefficient of determination is R2 = .214. The model is not significant (p-value = .199). This was only to be expected in view of the data limitations. We believe that these results should be construed as follows: 1) *interview*, *requirements* and *professional* experience is related to analyst effectiveness, but 2) *variables apart from experience are playing a role.*

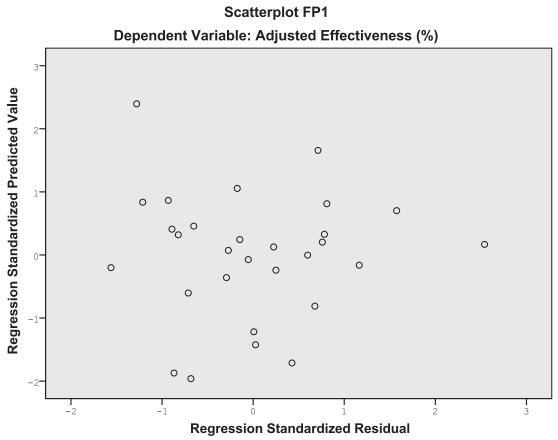

Fig. 3. Scatter plot composed of FP1 model residuals.

In conclusion, we cannot reject H_{0i}, which means that there is no relationship between experience and elicitation analyst effectiveness.

5 Discussion

5.1 General Observations

Our results show that for **unfamiliar domains** different experience types (interview, requirements, development and professional) do not have any effect on analyst effectiveness.

For **familiar domains**, the effect varies depending on experience type. **Interview experience clearly has a positive effect**, very close to statistical significance. Requirements experience might also have a positive effect, although our results are nowhere near significant. Note that the size of the sample that we managed to recruit for the familiar domain (29) is slightly lower than recommendations (40) as explained in Section 4.3.2. Therefore, non-significance could be due to a problem of statistical power. Development experience does not have any effect on analyst effectiveness.

On the other hand, **professional experience has a negative effect**, which is nearly significant, in familiar domains. We did not altogether expect this result. In view of the relative non-specificity of software engineer profiles, we thought there would be positive synergies between all (or many) activity types (e.g., years spent on programming or marketing could provide skills that improve requirements analysis). Our results show that this is not the case, illustrating that professional experience is indeed specific [21]. The same applies to development experience. The negative effect of professional experience may be interpreted as a justification of recruiting students, since experience yields no significant differences for effectiveness.

Note that the above results are **due exclusively to subject experience**, and domain knowledge is not having any impact. By separately analysing the two domains, we ensure that experience is the only variable influencing analyst effectiveness, as **domain knowledge is blocked**.

As regards knowledge effects, Table 11, reporting the results of the MRL for the unfamiliar domain, shows an intercept of 28.73%, whereas Table 12, for the familiar domain, has an intercept of 29.25%. The associated standard deviations (2.54% and 5.82%, respectively) suggest that the differences are not substantial, although analysts do, in any case, appear to be slightly more effective in the familiar domain (irrespective of experience). This suggests that domain knowledge has a positive (albeit small) effect on analyst effectiveness, as we reported in a previous paper [66]. In our view, the real effect is probably greater because: (1) experimental subjects have only very brief contact with FP1, that is, subjects are familiar with the domain concepts but did not work on them for a long time, and (2) subjects found some of the concepts, processes and requirements in FP1 to be particularly alien or hard to find. This means that, if such item types had not been present, subject effectiveness would have been much greater (higher intercepts and, probably, Bs).

Analysing the results we can summarize some implications for practice and for research. For practice, we can state that experience is not an important characteristic when analysts have to deal with unfamiliar domains. This means that non-experts, or even students, can have a similar effectiveness in this unfamiliar context. In a familiar domain, the most important experience is for "interviews". So analysts must focus on training with this technique to enhance their effectiveness. For research, more studies on the relationship between effectiveness and experience must be done. Note that the experiment was conducted in a controlled environment with a question-answer scenario. Maybe, in a less controlled environment, the experience can be affected by other aspects such as personality or context. Moreover, in this less controlled environment, requirements are not gathered but created in an interaction between the analyst and stakeholders. So, our results apply only to a strict question-answer scenario rather than more open exploratory interviews. To describe the sample, we consider subjects as experts when they have more than 5 years of experience, intermediate experience between 2 and 4 years, and non-experts between 0 and 1 year of experience. However, the statistical analyses are based on numerical values, not experience groups. Therefore such classification does not influence the results. Note that there are a few references in the related work section from recent years that have dealt with the topic of this paper. How our results contribute regarding other research works is analysed in next subsection.

5.2 Comparison with Related Empirical Work

Our findings are very much consistent with studies by Pitts & Browne [9], Marakas and Elam [4] Agarwal and Tanniru [6] and Hadar et al. [45], who, like us, used interviews as a requirements elicitation technique in their research. All four studies conclude that there are no significant differences between novice and experienced subjects, although experienced subjects turned out to be better (albeit very slightly) than novices. However, our results are contrary to findings reported by Niknafs and Berry [5], [17]. Note that the work of Niknafs and Berry considers the analysis of experience as a secondary study, that work focus mainly on how the domain knowledge may affect the effectiveness. So, this may affect how the experiment was designed and the obtained results.

The study by Pitts & Browne [9] is an experiment that finds the relationship between development experience and the number of requirements identified by analysts, where

the identified effect of r = 0.075. This value is, according to Cohen [67], equivalent to a very small effect. In our case, this correlation is r = -0.06[2], again a very small value, which is, for all practical purposes, negligible. The correlations that we and Pitts and Browne reported are not significant (p-value = 0.591 and 0.661, respectively).

The experimental study by Marakas and Elam [4] reported the effect of experience as a between-group difference (low vs high experience). As a percentage, experienced subjects are 3.09% more effective than the inexperienced subjects. The result is not statistically significant (p-value = 0.749). We understand that analysis experience is equivalent to requirements experience. Note that, in order to compare this effect with the findings of our study, we should divide 3.09% by the years of experience of the subjects in the high experience group. This data item is not available in Marakas and Elam [4]; it only indicates that "*highly experienced subjects were [. . .] employed at several major corporations*". Assuming that average experience is around 5 to 10 years, the effect per year would be from 3.09% / 5 = .618% to 3.09%/ 10 = .309%.

An additional problem with Marakas and Elam's study by that they do not clearly state whether their experiment addresses development or requirements experience. In our case, the effect of development experience is -0.619% (p-value = 0.765), whereas the effect of requirements experience is 2.9% (p-value = 0.270). The effect of experience that we calculated for Marakas and Elam's study falls in-between our values and is closer to the effect of development experience than requirements experience. This is, in our opinion, a predictable result, as we think that it is easier to recruit subjects with development experience than with requirements experience to perform an experiment.

The study by Agarwal and Tanniru [6] is also experimental. The experimental subjects were students and professionals with at least three years' experience in systems analysis. As far as we are concerned, it is equivalent to requirements experience. Experienced subjects were slightly more effective (by about 9%) than inexperienced subjects. This result is not significant; the authors state neither size nor direction. It is impossible to compare our results with theirs, as they do not define a gold standard for the rules (their dependent variable) within the experimental problem. We cannot transform their effect of experience into a value that is comparable with ours (percentage of elicited over total items).

Hadar et al. [45] focus on the study of domain knowledge using semi-structured interviews. We study experience, not domain knowledge. However, some connections with our work are apparent. They conclude that domain knowledge influences elicitation positively. When we compare the familiar and unfamiliar problem domains, we observe that analysts achieve the same effectiveness on average (26.94 and 26.74, respectively). However, the effectiveness increases with experience in the familiar domain. Experience does not play an effect by itself; it requires the concur of domain knowledge.

The work of Ferrari et al. [46] analyses whether, after teaching a requirements elicitation method, subjects with no previous experience are capable of avoiding mistakes in the elicitation process. Results show that the number of errors is significantly lower after the training. This result suggests that non-experts improve their elicitation effectiveness after just a short training. We fully agree. The warming-up moderator variable (see appendix D, available in the online supplemental material) indicates that after six weeks of an RE course, the students improve the elicitation effectiveness by 5%. Additionally, Ferrari et al.'s study was conducted on a domain that could be considered "familiar" to the experimental subjects (see https://zenodo.org/record/3765214#.YwNW3C8RrUI). In familiar domains, we also found that experience increases effectiveness.[45]

[68] For professional experience, Niknafs and Berry [17] detected a somewhat positive, albeit statistically non-significant, effect with respect to general, relevant and innovative idea generation, and a near significant positive effect for feasible idea generation. For requirements experience, we find that there is a statistically significant negative effect for one of the dependent variables studied (relevant idea generation). With regard to the other variables, the effect of requirements experience is sometimes positive and sometimes negative and is never significant.

Our results are not directly comparable with Niknafs and Berry's findings because the focus of both papers is not the same. While we aim to analyse the experience, Niknafs and Berry work focuses on analysing the different domains. The experience analysis is a side consideration of their main study on the effect of domain ignorance. So, some of the subjects that participated in the experiment were familiar and others were unfamiliar with the problem domain. Additionally, the range of requirements and professional experience is confined to from 0 to 4 years, whereas the range of experience is wider (from 0 to 30 years) in our case. As mentioned in previous studies [66], Niknafs and Berry are reporting an *einstellung* or functional fixation phenomenon [69] in the field of requirements elicitation.

5.3 Differential Effect of Experience by Item Type

Effectiveness could be affected by the manner in which analysts identify the items that define the problem domain (requirements, concepts and processes). For example, the types of items identified by experienced subjects and novices may differ. It seems reasonable to analyse whether experience effects differ by item type.

In order to determine whether the analysts capture one or other item type as a function of their experience, we used regression models to study the relationship between experience and analyst effectiveness for each of the items defining the problem domains. Appendix I, available in the online supplemental material, shows the calculations of the regression models equivalent to Tables 11 and 12 separated by concepts, processes and requirements. The results suggest that:

- For the unfamiliar domain, none of the experience types have an effect when analysed separately by processes, concepts and requirements.
- For the familiar domain, interview and requirements experience have positive effects across the board, that is, analysts with more years of interview and

2. Calculated using effect size calculators (http://www.uccs.edu/~lbecker) with df= N-4 and t taken from the regression model.

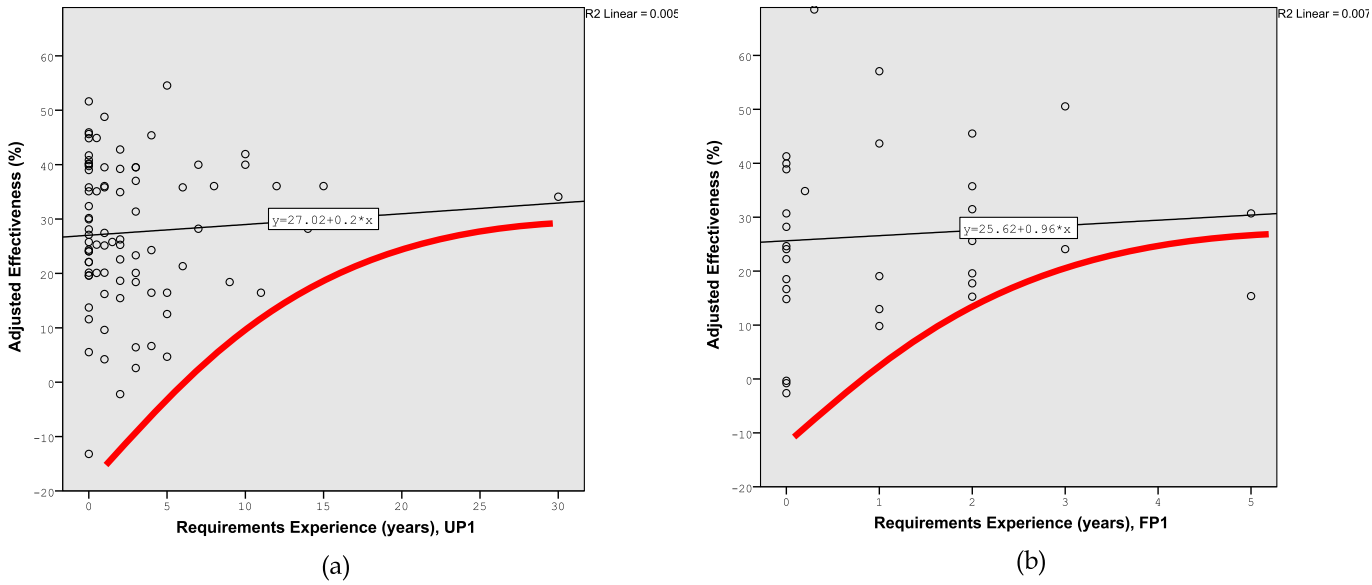

Fig. 4. Minimum effectiveness depending on years of requirements experience.

requirements experience are better able to identify more processes, concepts and requirements. This is especially true of processes and requirements for which some of the effects are even significant. Note, however, that the effect of development experience tends to be very low, having, for practical purposes, no effect whatsoever, whereas the effect of professional experience is usually negative.

Summarizing: the effects of the experience **do not change** regardless we use the Effectiveness variable, or the number of items of each type (concepts, processes, requirements).

5.4 A Possible Reinterpretation of the Experience Effect

The results suggest that, in unfamiliar domains, experience does not improve analyst effectiveness. However, this only applies on average. Fig. 4 shows the two scatter plots for the both unfamiliar domain (a) and the familiar domain (b) depending on years of requirements experience. The hand-drawn red line indicates the minimum analyst effectiveness as a function of their experience. Analysts with more years of experience achieve higher minimum effectiveness levels than analysts with less experience

The red line that we have plotted is similar to the *learning curves* often represented in the experience-related literature (e.g., Ericsson [33]). The learning curves are imaginary lines that denote how much an individual's effectiveness improves as his or her experience increases. In our case, this effectiveness improvement is related not to the individual effectiveness of each subject (effectiveness varies widely depending on years of experience) but to a minimum level of effectiveness that analysts almost always achieve (there is only one exception in FP1). In our opinion, Fig. 4 shows that requirements-related activities improve professional effectiveness more or less as specified by Sim et al. [35]. However, the lack of specialized training, explicit effectiveness evaluation, feedback, etc., prevents effectiveness from improving further.

Practice does not have a positive effect in the case of professional experience, as we discovered in Fig. 5. For the unfamiliar domain (a), the plotted learning curve is much flatter than the curves Fig. 4, as well as being a much worse fit for minimum effectiveness. For the familiar domain (b), this curve simply does not exist. We think that Fig. 5 illustrates that experience is specific, and skills acquired as a result of professional practice cannot be transferred to a specific field like requirements elicitation, as mentioned in Section 5.1.

6 Validity Threats

We grouped the validity threats from two viewpoints: a) threats specifically derived from quasi-experiment execution or contextual aspects, and b) general threats associated with statistical conclusion validity, internal validity, construct validity and external validity.

6.1 Specific Validity Threats

The results of our research could be affected by the threats specified below. After checking for each of the threats, we believe that they did not materialize:

- *Elicitation time.* The elicitation time available in the elicitation sessions is likely to be insufficient. To check out this possible threat, we studied the relationship between elicitation time and analyst effectiveness. Our results show that they are not correlated. For further details, see Appendix H.1, available in the online supplemental material.
- *Number of elicitation sessions.* The failure to detect significant effects for experience may be due to the fact that only one interview was conducted in each of the long series of quasi-experiments that we have conducted. Analyst effectiveness is likely to improve as the number of elicitation sessions increases. Our results show that an increase in the number of sessions (to two) does not result in any improvement in analyst effectiveness. For further details, see Appendix H.2, available in the online supplemental material.

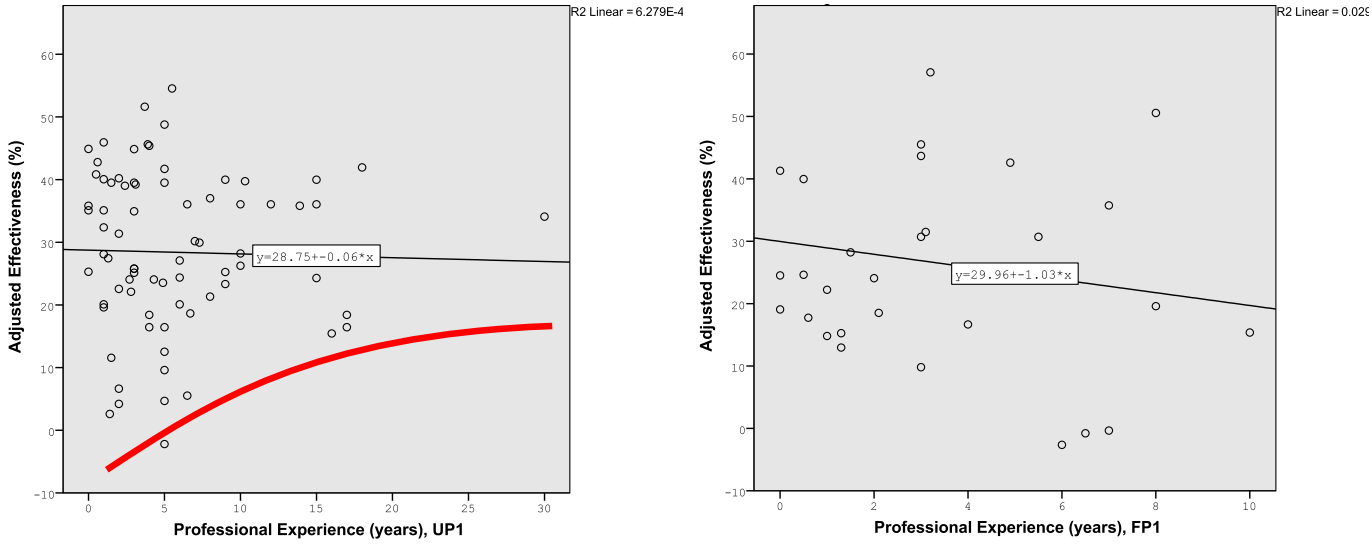

Fig. 5. Minimum effectiveness depending on years of professional experience.

Notice that we do not claim that 30 minutes or one elicitation session are enough to perform requirements elicitation *in all cases*. We simply point out that the complexity of the experimental objects was sufficiently small as to subjects become satisfied with their understanding after one ∼30-minute session. This impression was corroborated in the post-experimental questionnaire. In turn, most of the subjects manifested that they would perform more interviews if they could.

- *Comparison of individual vs. group interviews.* In group sessions, analyst effectiveness could possibly be affected by knowledge transfer from more experienced to less experienced subjects. As illustrated in Appendix H.3, available in the online supplemental material, no such knowledge transfer is observed.
- *Joint analysis of quasi-experiments.* In this study, we have analysed the data jointly without taking into account that they are taken from several experiments. The results of our analysis would not be reliable if the populations of the different quasi-experiments had different characteristics. We analysed the distribution of the model residuals depending on the quasi-experiment to which they belong. This analysis is available in Appendix J, available in the online supplemental material. The means and variances of the residuals can be observed to be consistent from one quasi-experiment to another. Additionally, the residuals are scattered around zero. We would expect this to be the case if the populations of the quasi-experiments do not have any influence on analyst effectiveness.

6.2 General Validity Threats

The possible threats [70] that we think may affect the quasi-experiments are as follows:

6.2.1 Statistical Conclusion Validity Threats

- *Low statistical power:* A total of 90 and 40 experimental subjects are required in both the unfamiliar and familiar problem domains, respectively, to study experience. We managed to gather information from 69 and 29 subjects, respectively. Note that although the sample size is less than the recommended, we are close to this limit. In both cases, we have at least the 75% of the subjects required for medium effects. We should note that:
  a) For the unfamiliar domain, the sample is greater than necessary (88 subjects) to achieve a power of 80% and detect medium-sized effects with $\alpha = 0.10$. Therefore, we believe that, although the results are not significant, they are genuine and not the product of type-II error [71].
  b) For the familiar domain, interview experience and professional experience have very low p-values (significant at $\alpha = 0.10$, if you prefer) with a sample of 36 subjects. Acting proactively, we considered the independent variables to be influential, anticipating a more than likely type-II error.
- *Violated assumptions of statistical tests.* In the familiar domain, we cannot assure homogeneity of variances. If the variances were not homogeneous, one of the conditions for applying MLR would be violated. Consequently, the results of the inference tests (p-values) could be mistaken. The non-standardized effects (B) would not be affected.
- *Unreliability of measures.* This threat to validity can operate in both the dependent and independent variables:
  c) The dependent variable effectiveness was measured in different settings (for example, individual or group interviews). We assume that some effects derived from the setting could be influencing effectiveness. These effects were eliminated by means of a data adjustment procedure.
  d) Measure for effectiveness depends on how subjects extract requirements using interviews. How these requirements are discovered may be subjective, and differently of how the process is done in

real life. We mitigated this threat using a context as similar as possible to a real context; clients gave as much details as they knew and analysts had to guide the interviews through questions.

e) The different types of experiences were reported by the experimental subjects using a post-experimental questionnaire. Self-reported values can be unreliable, e.g., inflated, as already observed in Aranda et al. [66]. Inflated experience values would flatten the regression lines, leading to low correlations. This threat cannot be counteracted easily, because there is not any reasonable alternative procedure to collect subject experience data. **In our opinion, this is the most serious threat that our research is experiencing.**

- *Measurement bias*. The measurements were made by a single researcher. We cannot rule out that there may have been some bias. For example, subjects studying the unfamiliar domain may have been unintentionally penalized. To prevent this bias, researchers did not access the demographic data until the effectiveness measurement was complete.
- *Restriction of range*. It is not easy to recruit experienced subjects in empirical studies. Our study is no exception. Most subjects had about 0 to 5 years of experience. Only a fraction of the subjects had longer experience. This could have a negative effect on the estimation of effect sizes. However, we believe that this is a secondary problem:

f) On one hand, most subjects (including the more experienced ones) were interviewed about unfamiliar domains. There is quite a wide range of experience in this domain, on which ground we believe that the result of the MLR is quite reliable. In fact, the MLR with bootstrapping (to overcome the range limitations) yields similar results to the ordinary MLR. In the worst case, if interview and requirements experience had a positive effect on analyst effectiveness, the size of the effect would be low.

g) In the familiar domain, this threat should tend to support the conclusions of this study, that is, the positive effects of interview experience (and probably requirements experience) and the negative effects of professional experience.

6.2.2 Internal Validity Threats

- *Participant selection*. It can operate in two different ways:

a) The subjects are not taken from a random sample in any of the quasi-experiments. Therefore, the results may be biased depending on particular characteristics of the populations used. We believe that this threat is unlikely to materialize as multiple quasi-experiments were conducted over several years in at least two settings: academia (UPM) and a professional congress (REFSQ). There is unlikely to be a common moderator variable in all cases.

b) A possible exception to the former is the *Familiarity* with the problem domain. We assume that the subjects are familiar with FP and unfamiliar with UP1. The former is likely, the latter not necessarily. In the post-experimental questionnaire, we asked subjects about their familiarity with the problem domain. The analysis of the relationship between familiarity and effectiveness yielded non-significant results.

c) Subjects may be familiar with the problem domains to different degrees. We also questioned the subjects about the problem domain *Difficulty*. Again, we did not identify any clear pattern.

d) Subjects are not practitioners. Even though we can ensure that subjects had enough knowledge to participate in the experiment, we cannot generalize the results to practitioners.

- *Ambiguous temporal precedence.* The conclusions of this research do not draw any causal relations because they are based on quasi-experiments. Even though we studied several experience-related independent variables with respect to experimental task effectiveness, there could be moderator variables that we have not accounted for that could explain the results. As a mitigation strategy, we measured all the observable moderator variables, like warming up, for example, which we considered explicitly in the analysis. We count on other variables, for example, any related to soft skills or analyst personality, offsetting each other in the overall analysis.

6.2.3 Construct Validity Threats

- *Construct confounding.* During the execution of the quasi-experiments, we purposely confounded several variables so as not to have to study their interaction:

a) *Confounding of elicitation time and interview type*. In the quasi-experiments Q-2011 and Q-2012, the increased effectiveness of the subjects could be due to a longer elicitation time (up to 60 minutes) or interview type (group interview) where all the participants have access to exactly the same information. In this case, we cannot separate the effects of the two variables. However, as they were confounded throughout, and neither the interview type nor the elicitation type are key variables in our research, their confounding does not pose a threat to our conclusions.

b) *Confounding of interviewee and language*. From E-2012 onwards, the interviewee and language variables are confounded. As above, neither the language nor the interviewee are key variables, on which ground their confounding is not a threat.

- *Mono-operation bias.* There are two threats of this type:

c) The independent variables (the different experiences) were measured in years. Year-based measurement may not accurately represent the level of subject expertise. Other measures (e.g., number of projects) may perhaps have been a better option. Although this may well be the case, research into experience (see Section 2) has

historically applied year-based measurement. Additionally, the use of other types of measures does not appear to make a big difference to the results of measurement in years [72]. In our particular case, we analysed a subset of high performing vs. low performing subjects (14 subjects in each group) on a number of variables (including the number of projects), as shown in Appendix H.4, available in the online supplemental material. None of the variables was able to explain the differences in subject effectiveness in the unfamiliar domain. In the familiar domain, the number of job positions yields low p-values ($< .10$), corroborating our results regarding the positive effects of interview experience. Also, in the familiar domain, low p-values ($< .10$) are associated to subject education; this suggests that personal characteristics may underlie the differences in analyst effectiveness.

d) The dependent variable (effectiveness) was operationalized in two different ways: 1) as the percentage of items acquired by subjects, and 2) as the total number of items separated by type (requirements, concepts and processes). The effect of experience was similar in both cases.

6.2.4 External Validity Threats

- *Interaction of the causal relationship with units.* The fact that experimental subjects are taken from a convenience sample rather than a random sample (that is, subjects are students enrolled in a particular course and not recruited from a larger population) poses a threat to the external validity of the experiment. Therefore, due care should be taken when generalizing our results to professional analysts. However, this threat was addressed since the students were taking a professional master's degree and most had professional experience in computer-related tasks. Therefore, subjects can be considered to be reasonably representative of the developer/analyst population. We believe that our results can at least be generalized to junior developers new to elicitation.
- *Interaction of the causal relationship with settings.* Quasi-experiments were conducted in the laboratory. This means that the setting is completely different to what analysts are used to in the professional world:
  a) The client participating in the interview is a simulation and will not, therefore, act exactly like a real client. To mitigate this threat, the subjects playing the role of interviewee carefully studied the problem domains in order to answer the questions as naturally and realistically as possible.
  b) The software system/s to be developd are not real. To mitigate this threat, the systems used in the quasi-experiments are based on real software systems. We checked that the complexity of the problems was as similar as possible to ensure that problem size is not another factor influencing analyst effectiveness.
  c) The time taken to complete the interviews is limited. We believe that elicitation time was not an obstacle to gathering information. Most subjects finished the elicitation session before the end of the specified time. Additionally, most of the complexity of the original system was eliminated from the experimental objects (that is, the description of the familiar and unfamiliar domains) to assure that they were easy to understand in a short period of time. Finally, there are not statistically significant differences in effectiveness between subjects that considered that the elicitation time was sufficient vs. those who did not.

7 Conclusion

This research studied the effects of different types of experience (interviews, requirements, development and professional) on the elicitation effectiveness of a set of 124 experimental subjects with different levels of experience. Due to the use of MLR with four independent variables, we had to discard subjects with some of these variables missing. This reduces the sample size to 98 subjects. Thanks to the experimental design used, we can study the effect of experience separately from the effect of domain knowledge. To do this, we used an unfamiliar and a familiar problem domain, respectively. Our results show that experience has different effects on analyst effectiveness, depending on the type of problem domain in which the elicitation takes place:

- In the **unfamiliar problem domain, the type of experience** (interviews, requirement, development or professional) **has no effect whatsoever** on analyst effectiveness. It means that the number of years of professional practice in RE does not substantially improve analyst effectiveness. Fortunately, this only applies on average, as we have found that the **baseline effectiveness increases as analysts acquire interview or requirements experience (**in both unfamiliar and familiar domains). Our interpretation of this result is that, although the RE discipline provides knowledge and strategies that analysts employ by during the elicitation process, it is not sufficient to achieve expertise. A large share of analyst effectiveness is explained by other issues, probably psychological or personal. *Identifying soft skills associated with highly effective analysts* would not only lead to a better understanding of what makes a highly effective analyst but also help to improve analyst training [73].
- In the *familiar problem domain, interview experience has a* (near significant) **positive effect**. **Professional** experience has a **moderate** (also near significant) **negative effect**. Requirements experience **could have a moderate positive effect**, although the results are nowhere near statistically significant. In our opinion, such results imply that analyst effectiveness is contingent on their previous exposure to the problem domain (and, probably, similar domains as well). The logical consequence is that **requirements analysts should move among development projects in the same (/related) domain(s) exclusively**. Likewise, special **attention should be paid**

to problem domains in requirements courses. It is even possible that a substantial share of the training should be spent in studying and carrying out course projects in relevant domains, e.g., banking, insurance, manufacture, health, etc.

Finally, our research shows that analyst effectiveness only improves with specialized training or practice. In other words, development experience and professional experience do not improve analyst effectiveness, but specialized experience (interview and, possibly, requirements experience) does lead to improvement. Therefore, experience behaves similarly in the field of RE to how it does in other branches of knowledge.

The confirmation of previous research in different contexts, and using different methodologies, increases the confidence in the results. However, empirical research is subjected to a large number of threats to validity. Our final word is a call for further research. The communication among stakeholders is a foundation process in requirements engineering, and it is not likely that it disappears or loses relevance in the near future. Elicitation is one the main sources of software quality problems. A better understanding of how elicitation works will contribute to increasing software quality.

REFERENCES

[1] D. Zowghi and C. Coulin, "Requirements elicitation: A survey of techniques, approaches, and tools," in *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin, Eds. Berlin, Germany: Springer, 2005, pp. 19–46.

[2] A. Aurum and C. Wohlin, *Engineering and Managing Software Requirements*. Berlin, Germany: Springer, 2005.

[3] F. Anwar, R. Razali, and K. Ahmad, "Achieving effective communication during requirements elicitation - A conceptual framework," in *Software Engineering and Computer Systems*, S. Zain, W. Wan Mohd, and E. El-Qawasmeh, Eds. Berlin, Germany: Springer, 2011, pp. 600–610.

[4] G. M. Marakas and J. J. Elam, "Semantic structuring in analyst and representation of facts in requirements analysis," *Inf. Syst. Res.*, vol. 9, no. 1, pp. 37–63, 1998.

[5] A. Niknafs and D. M. Berry, "The impact of domain knowledge on the effectiveness of requirements idea generation during requirements elicitation," in *Proc. IEEE 20th Int. Requirements Eng. Conf.*, 2012, pp. 181–190.

[6] R. Agarwal and M. R. Tanniru, "Knowledge acquisition using structured interviewing: An empirical investigation," *J. Manage. Inf. Syst.*, vol. 7, no. 1, pp. 123–141, 1990.

[7] D. Carrizo, O. Dieste, and N. Juristo, "Systematizing requirements elicitation technique selection," *Inf. Softw. Technol.*, vol. 56, no. 6, pp. 644–669, 2014.

[8] A. Albayrak and J. Carver, "Investigation of individual factors impacting the effectiveness of requirements inspections: A replicated experiment," *Empirical Softw. Eng.*, vol. 19, no. 1, pp. 241–266, Feb. 2014.

[9] M. G. Pitts and G. J. Browne, "Stopping behavior of systems analysts during information requirements elicitation," *J. Manage. Inf. Syst.*, vol. 21, no. 1, pp. 203–226, 2004.

[10] P. Loucopoulos and V. Karakostas, *Systems Requirements Engineering*. London, U.K: McGraw-Hill, 1995.

[11] M. G. Christel and K. C. Kang, "Issues in require-ments elicitation," 1992. [Online]. Available: http://www.Sei.Cmu.Edu/Publications/Documents/92.Reports/92.Tr.012.html

[12] R. R. Young, "Recommended requirements gathering practices," *Crosswalks*, vol. 15, no. 4, pp. 9–12, 2002.

[13] A. Niknafs and D. M. Berry, "An industrial case study of the impact of domain ignorance on the effectiveness of requirements idea generation during requirements elicitation," in *Proc. IEEE 21st Int. Requirements Eng. Conf.*, 2013, pp. 279–283.

[14] S. Helmstetter, R. Germann, M. Abbes, and S. Matthiesen, "Human factors for the evaluation of the user expertise in the usage of power tools," in *Proc. 31st Sympos. Des. X (DFX2020)*, 2020, pp. 21–30.

[15] N. P. Vitalari, "Structuring the requirements analysis process for information systems: A proposition viewpoint," in *Challenges and Strategies for Research in Systems Development*, W. W. Cotterman and J. A. Sen, Eds. New York, NY, USA: Wiley, 1992, pp. 163–179.

[16] D. B. Walz, J. J. Elam, and B. Curtis, "Inside a soft-ware design team: Knowledge acquisition, sharing, and integration," *Commun. ACM*, vol. 36, pp. 62–77, 1993.

[17] A. Niknafs and D. Berry, "The impact of domain knowledge on the effectiveness of requirements engineering activities," *Empirical Softw. Eng.*, vol. 22, no. 1, pp. 80–133, Feb. 2017.

[18] A. Geraci, "IEEE standard computer dictionary: Compilation of IEEE standard computer glossaries," IEEE Press, Piscataway, NJ, USA, 1991.

[19] A. D. De Groot, *Thought and Choice in Chess*. Berlin, Germany: Walter de Gruyter, 19784.

[20] G. Colvin, *Talent is Overrated: What Really Separates World-Class Performers From Everybody Else*. London, UK: Penguin Books, 1973.

[21] G. Colvin, *Talent is Overrated: What really Separates World-Class Performers from EverybodyElse*. New York, NY, USA: Penguin Publishing Group, 2008.

[22] B. Curtis, "Fifteen years of psychology in software engineering: Individual differences and cognitive science," in *Proc. 7th Int. Conf. Softw. Eng.*, 1984, pp. 97–106.

[23] S. Baltes and S. Diehl, "Towards a theory of software development expertise," in *Proc. 26th ACM Joint Meeting Eur. Softw. Eng. Conf. Symp. Foundations Softw. Eng.*, 2018, pp. 187–200.

[24] K. F. MacDorman et al., "An improved usability measure based on novice and expert performance," *Int. J. Hum. Comput. Interact.*, vol. 27, no. 3, pp. 280–302.

[25] G. Catolino et al., "How the experience of development teams relates to assertion density of test classes," in *Proc. IEEE Int. Conf. Softw. Maintenance Evol.*, 2019, pp. 223–234.

[26] R. Latorre, "Effects of developer experience on learning and applying unit test-driven development," *IEEE Trans. Softw. Eng.*, vol. 40, no. 4, pp. 381–395, Apr. 2014.

[27] P. Li, A. Ko, and J. Zhu, "What makes a great software engineer?," *IEEE/ACM 37th IEEE Int. Conf. Softw. Eng.*, vol. 1, pp. 700–710, 2015.

[28] K. Rosenthal, S. Strecker, and O. Pastor, "Modeling difficulties in data modeling," in *Proc. Int. Conf. Conceptual Model.*, 2020.

[29] I. M. D. Oca et al., "A systematic literature review of studies on business process modeling quality," *Inf. Softw. Technol.*, vol. 58, pp. 187–205, 2015.

[30] D. Chow et al., "The role of deliberate practice in the development of highly effective psychotherapists," *Psychotherapy*, vol. 52, pp. 337–345, 2015.

[31] M. T. Chi, P. J. Feltovich, and R. Glaser, "Categorization and representation of physics problems by experts and novices," *Cogn. Sci.*, vol. 5, no. 2, pp. 121–152, 1981.

[32] K. A. Ericsson, R. T. Krampe, and C. Tesch-Römer, "The role of deliberate practice in the acquisition of expert performance," *Psychol. Rev.*, vol. 100, no. 3, pp. 363–406, 1993.

[33] K. A. Ericsson, "The influence of experience and deliberate practice on the development of superior ex-pert performance," in *The Cambridge Handbook of Expertise and Expert Performance*, K. A. Ericsson et al., Ed. Cambridge, U.K.: Cambridge Univ. Press, 2006, pp. 683–703.

[34] R. L. Campbell and L. D. Bello, "Studying human expertise: Beyond the binary paradigm," *J. Exp. Theor. Artif. Intell.*, vol. 8, no. 3-4, pp. 277–291, Jul. 1996.

[35] S. E. Sim et al., "An initial study to develop an empirical test for software engineering expertise," Inst. Softw. Res., Univ. California, Irvine, CA, USA, Tech. Rep. UCI-ISR-06-6, 2006.

[36] C. Atkins, "An investigation of the impact of requirements engineering skills on project success," East Tennessee State Univ., Johnson City, TN, USA, Tech. Rep. 1522851, 2013.

[37] N. C. Shrikanth et al., "Assessing practitioner beliefs about software engineering," *Empirical Softw. Eng.*, vol. 26, no. 4, May 2021, Art. no. 73.

[38] C. F. Camerer and E. J. Johnson, "The process-performance paradox in expert judgment: How can experts know so much and predict so badly?," in *Research on Judgment and Decision Making: Currents, Connections, and Controversies*, W. Goldstein and R. Hogarth, Eds. Cambridge, U.K.: Cambridge Univ. Press, 1997, pp. 342–364.

[39] K. A. Ericsson and J. Smith, "Prospects and limits of the empirical study of expertise: An introduction," in *Toward General Theory Expertise: Prospects Limits*, Cambridge, U.K.: Cambridge Univ. Press, 1991, pp. 1–38.

[40] S. Sonnentag, C. Niessen, and J. Volmer, "Expertise in software design," *Cambridge Handbook of Expertise and Expert Performance*. Cambridge, UK: Cambridge University Press, 2006, pp. 373–387.

[41] M. A. McDaniel, F. L. Schmidt, and J. E. Hunter, "Job experience correlates of job performance," *J. Appl. Psychol.*, vol. 73, no. 2, 1988, Art. no. 327.
[42] J. Siegmund et al., "Measuring and modeling programming experience," *Empirical Softw. Eng.*, vol. 19, no. 5, pp. 1299–1334, 2014.
[43] Z. Zhang, "Effective requirements development-A comparison of requirements elicitation techniques," *Software Quality Management XV: Software Quality in the Knowledge Society*, E. Berki, J. Nummenmaa, I. Sunley, M. Ross, and G. Staples Ed., Swindon, U.K.: British Computer Society, 2007, pp. 225–240.
[44] R. Agarwal and M. R. Tanniru, "Knowledge extraction using content analysis," *Knowl. Acquisition*, vol. 3, pp. 421–441, 1991.
[45] I. Hadar, P. Soffer, and K. Kenzi, "The role of domain knowledge in requirements elicitation via interviews: An exploratory study," *Requirements Eng.*, vol. 19, pp. 1–17, Sep. 2012.
[46] A. Ferrari, P. Spoletini, M. Bano, and D. Zowghi, "Learning requirements elicitation interviews with role-playing, self-assessment and peer-review," in *Proc. IEEE 27th Int. Requirements Eng. Conf.*, 2019.
[47] K. D. Schenk, N. P. Vitalari, and K. S. Davis, "Differences between novice and expert systems analysts: What do we know and what do we do?," *J. Manage. Inf. Syst.*, vol. 15, no. 1, pp. 9–50, Jun. 1998.
[48] G. J. Browne, "Information requirements determination," in *Computing Handbook, Third Edition: Information Systems and Information Technology*, H. Topi and A. Tucker, Eds. Boca Raton, FL, USA: CRC Press, 2014, pp. 27:1-27:14.
[49] A. Jedlitschka and D. Pfahl, *Reporting Guidelines for Controlled Experiments in Software Engineering*. Kaiserslautern, Germany: Fraunhofer IESE, 2005.
[50] S. Debnath, P. Spoletini, and A. Ferrari, "From ideas to expressed needs: An empirical study on the evolution of requirements during elicitation," in *Proc. IEEE 29th Int. Requirements Eng. Conf.*, 2021.
[51] A. M. Burton et al., "Knowledge elicitation techniques in classification domains," in *Proc. 8th Europuean Conferene AI*, 1988, pp. 85–90.
[52] G. J. Browne and M. B. Rogich, "An empirical investigation of user requirements elicitation: Comparing the effectiveness of prompting techniques," *J. Manage. Inf. Syst.*, vol. 17, no. 4, pp. 223–249, 2001.
[53] C. Pacheco, I. García, and M. Reyes, "Requirements elicitation techniques: A systematic literature review based on the maturity of the techniques," *IET Softw.*, vol. 12, no. 4, pp. 365–378, 2018.
[54] S. B. Yadav et al., "Comparison of analysis techniques for information requirements determination," *Commun. ACM*, vol. 31, no. 9, pp. 1090–1097, 1988.
[55] A. Davis, *Software Requirements: Objects, Functions and States*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.
[56] E. D. Falkenberg et al., "FRISCO: A framework of information system concepts," in *IFIP WG*, vol. 8, 1997.
[57] C. A. Gunter et al., "A reference model for require-ments and specifications," *IEEE Softw.*, vol. 17, no. 3, pp. 37–43, May/Jun. 2000.
[58] D. Falessi et al., "Empirical software engineering experts on the use of students and professionals in experiments," *Empirical Softw. Eng.*, vol. 23, no. 1, pp. 452–489, Feb. 2018.
[59] A. M. Aranda et al., "Material of effect of requirements analyst experience on elicitation effectiveness: A family of quasi-experiments," *Zenodo*, 2022, doi: 10.5281/zenodo.6139653.
[60] R. O'brien, "A caution regarding rules of thumb for variance inflation factors," *Qual. Quantity*, vol. 41, no. 5, pp. 673–690, Oct. 2007.
[61] R. M. Heiberger and B. Holland, *Statistical Analysis and Data Display: An Intermediate Course With Examples in S-Plus, R, and SAS*. 1st ed. New York, NY, USA: Springer, 2013.
[62] D. A. Belsley, *Conditioning Diagnostics: Collineari-ty and Weak Data in Regression*. New York, NY, USA: Wiley, 1991.
[63] J. Miles and M. Shevlin, *Applying Regression and Correlation: A Guide for Students and Researchers*. Thousand Oaks, CA, USA: SAGE Publications, 2000.
[64] A. Field, J. Miles, and Z. Field, *Discovering Statistics using R*. Thousand Oaks, CA, USA: SAGE Publications, 2012.
[65] M. G. Mendonça et al., "A framework for software engineering experimental replications," in *Proc. IEEE 13th Int. Conf. Eng. Complex Comput. Syst.*, 2008, pp. 203–212.
[66] A. M. Aranda, O. Dieste, and N. Juristo, "Effect of domain knowledge on elicitation effectiveness: An internally replicated controlled experiment," *IEEE Trans. Softw. Eng.*, vol. 42, no. 5, pp. 427–451, May 2016.
[67] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. 2nd ed. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1988.
[68] A. Ferrari, P. Spoletini, and S. Gnesi, "Ambiguity and tacit knowledge in requirements elicitation interviews," *Requirements Eng.*, vol. 21, no. 3, pp. 333–355, Sep. 2016.
[69] J. Anderson, *Cognitive Psychology and its Implications*. 5th ed., New York, NY, USA: Worth Publishers, 2000.
[70] C. Wohlin et al., *Experimentation in Software Engineering*. 1st ed. Berlin, Germany: Springer, 2012.
[71] B. S. Everitt and A. Skrondal, "The cambridge dictionary of statistics," 2010.
[72] K. Ericsson and A. Lehmann, "Expert and exceptional performance: Evidence of maximal adaptation to task constraints," *Annu. Rev. Psychol.*, vol. 47, pp. 273–305, 1996.
[73] A. Ferrari et al., "SaPeer and ReverseSaPeer: Teaching requirements elicitation interviews with role-playing and role reversal," *Requirements Eng.*, vol. 25, no. 4, pp. 417–438, Dec. 2020.

Alejandrina M. Aranda received the MSc degree in information technology and the PhD degree from the Universidad Politécnica de Madrid. She is a researcher with the UPM's School of Computing. Her research interests include requirements engineering and empirical software engineering.

Oscar Dieste received the BS and MS degrees in computing from the University of La Coruña, and the PhD degree from the University of Castilla-La Mancha. He is a researcher with the UPM's School of Computer Engineering. He was previously with the University of Colorado at Colorado Springs (as a Fulbright scholar), Complutense University of Madrid, and the Alfonso X el Sabio University. His research interests include empirical software engineering, and requirements engineering.

Jose Ignacio Panach received the PhD degreee in computer science in 2010. He has been an assistant professor with the Universitat de València since 2011 and an assistant researcher with the Valencian Research Institute of Artificial Intelligence–VRAIN, Universidad Politécnica de Valencia since 2005. His research activities focus on MDD, usability, and interaction modelling.

Natalia Juristo is full professor of software engineering with the Computing School, Technical University of Madrid (UPM) since 1997, and holds a FiDiPro (Finland Distinguish professor) research grant since 2013. She was the director of the UPM MSc in Software Engineering from 1992 to 2002 and the coordinator of the Erasmus Mundus European Master on SE (whith the participation of the University of Bolzano, the University of Kaiserslautern and the University of Blekinge) from 2007 to 2012. Natalia has served in several *Program Committees* ICSE, RE, REFSQ, ESEM, ISESE and others. She has been *Program Chair* EASE13, ISESE04 and SEKE97 and General Chair for ESEM07, SNPD02 and SEKE01. She has been member of several Editorial Boards, including *IEEE Transactions Software Engineering*, *Journal of Empirial Software Engineering and Software magazine*. She has been gugest editor of special issues in several journals, including *Journal of Empirial Software Engineering*, *IEEE Software*, *Journal of Software and Systems*, *Data and Knowledge Engineering* and the *International Journal of Software Engineering and Knowledge Engineering*.