Analyzing Software Engineering Experiments: Everything You Always Wanted to Know but Were Afraid to Ask

Sira Vegas
Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid
Madrid, Spain
svegas@fi.upm.es

ABSTRACT

Experimentation is a key issue in science and engineering. But it is one of software engineering's stumbling blocks. Quite a lot of experiments are run nowadays, but it is a risky business. Software engineering has some special features, leading to some experimentation issues being conceived of differently than in other disciplines. The aim of this technical briefing is to help participants to avoid common pitfalls when analyzing the results of software engineering experiments. The technical briefing is not intended as a data analysis course, because there is already plenty of literature on this subject. It reviews several key issues that we have identified in published software engineering experiments, and addresses them based on the knowledge acquired after 19 years running experiments.

CCS CONCEPTS

• **Software and its engineering → Software creation and management**

KEYWORDS

Software Engineering experimentation, controlled experiments, analysis of experiments

ACM Reference format:

S. Vegas. 2018. Analyzing Software Engineering Experiments: Everything You Always Wanted to Know but Were Afraid to Ask. In *Proceedings of 40th International Conference on Software Engineering, Gothenburg, Sweden, May 27-June 3, 2018 (ICSE'18 Companion), 2 pages.*

DOI: 10.1145/3183440.3183466

1 DESCRIPTION OF THE TOPIC

The goal of this technical briefing is to help participants to improve how they analyze the data obtained when running software engineering (SE) controlled experiments.

ICSE'18 Companion, Gothenburg, Sweden

DOI: 10.1145/3183440.3183466

It starts by recalling what a SE experiment is, and its distinctive features: control and causality. This introduction will last *5 minutes*.

Then, five topics related to data analysis will be explored. Advice will be given to participants based on our own experiences after 19 years running controlled experiments. Several real experiments will be used during the whole technical briefing, to illustrate the issues covered:

1. **One-tailed vs. two-tailed tests**. SE experimenters very often opt for a one-tailed hypothesis, but this can be a shortcoming in many experiments. We will discuss when each type should be used (*10 minutes*).
2. **Choosing the right data analysis technique**. Data analysis is driven by experimental design. The selected data analysis technique should match the experimental design. However, the choice is not straightforward, and several issues have to be taken into consideration. We will discuss these issues (*10 minutes*).
3. **Analyzing tricky designs**. Special care has to be taken when analyzing complex designs. We will discuss two designs which are commonly not properly analyzed: blocked and crossover designs *(15 minutes)*.
4. **Parametric vs. non-parametric tests**. Parametric tests are more powerful than non-parametric tests, and can be used to analyze factorial experiments. However, data do not always meet their requirements. We will discuss the different options that can be used when data do not meet the parametric tests assumptions: transformations and non-parametric tests *(20 minutes)*.
5. **The 3 musketeers: statistical significance, effect size and power**. We will discuss the meaning and implications of the three parameters, how they relate to each other, and how they should be used to properly interpret the results of an experiment. Non-significant results might be due to lack of power. Statistical significant results might be not relevant due to small effect sizes, etc. *(20 minutes)*.

The technical briefing will end with participants' **questions** *(10 minutes)*.

2 INTEREST OF THE TOPIC FOR THE SOFTWARE ENGINEERING COMMUNITY

It is now very common practice to conduct laboratory experiments in SE. However, this is a challenging error-prone activity. Shepperd, Bowes and Hall [2] analyzed the results of 42

papers reporting studies comparing methods for predicting fault-proneness. They found that the explanatory factor that accounted for the largest percentage of the differences between studies (30%) was research group [2]. In contrast, the main topic of research, accounted for only 1.3% of the variation between the studies. Sjøberg *et al.* [3] reported that replications performed by researchers who undertook the initial study were more likely to find the same result than replications undertaken by independent researchers. In the field of psychology, Pashler and Wagenmakers [1]report "a crisis of confidence in psychological science reflecting an unprecedented level of doubt among practitioners about the reliability of research findings in the field".

Experimentation is quite a recent practice in SE (compared with other much more mature experimental disciplines). We still have a long way to go, and much more effort and research is needed to adapt the experimental paradigm to SE. Experimentalism is a paradigm that needs to be instantiated, translated and adapted to the idiosyncrasy of each experimental discipline. Copy and paste, that is, copy from physics what an experiment is, copy from medicine the threats to the validity of experiments, or copy from psychology how to deal with experimental subjects, will not do. We can borrow from other experimental disciplines, but our field needs to adopt its own form of experimentalism.

The technical briefing focuses on aspects of SE experiment analysis that often are error prone.

3 PRESENTER'S BACKGROUND

3.1 Bio

Sira Vegas is associate professor of software engineering at Universidad Politécnica de Madrid. Her main research interests are experimental software engineering and software testing. She is regular reviewer of highly ranked journals such as IEEE Transactions on Software Engineering, Empirical Software Engineering Journal and ACM Transactions on Software Engineering. Dr. Vegas was program chair for the International Symposium on Empirical Software Engineering and Measurement (ESEM) in 2007.

3.2 Empirical Software Engineering Tutorials Taught

- **S. Vegas**. *Analyzing Software Engineering Experiments: Everything You Always Wanted to Know but Were Afraid to Ask.* 39th International Conference on Software Engineering (ICSE'17). Buenos Aires, Argentina, 2017.
- N. Juristo, **S. Vegas**. Analyzing Software Engineering Experiments: Everything You Always Wanted to Know but Were Afraid to Ask. 38th International Conference on Software Engineering (ICSE'16). Austing, Texas, 2016.
- N. Juristo, **S. Vegas**. *Challenges of Conducting Software Engineering Experiments: Everything You Always Wanted to Know but Were Afraid to Ask.* Half day. 10th joint meeting of the European Software Engineering Conference and the Symposium on the Foundations on Software Engineering (ESEC/FSE'15). Bergamo, Italy, 2015.
- O. Dieste, N. Juristo, **S. Vegas**. *Replication and Aggregation of Software Engineering Experiments.* Full day. 6th International Advanced School on Empirical Software Engineering (IASESE'08), Kaiserslautern, Germany, 2008.
- A. Moreno, **S. Vegas**. *Experimentation in Software Engineering.* Half day. I Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento (JIISIC'01). Buenos Aires, Argentina, 2001.

3.3 Empirical Software Engineering Keynotes and Invited Talks

- ***S. Vegas**. What Makes a Good Empirical Software Engineering Thesis?: Some Advice.* 13th International Doctoral Symposium on Empirical Software Engineering (IDoESE'15). Beijing, China, 2015.
- **S. Vegas.** *Using Differences among Replications of Software Engineering Experiments to Gain Knowledge.* 7th Working conference on Mining Software Repositories (MSR'10). Cape Town, South Africa, 2010.

3 TECHNICAL BRIEFING HISTORY

This technical briefing has its roots in the one-hour talk given by Natalia Juristo at the ICSE 2014 Doctoral Symposium, entitled "Basics on Design and Analysis of SE Experiments: Widespread Shortcomings". It covers some of the topics (the ones related to the analysis of experimental data) taught on the more general four-hour tutorial given by Natalia Juristo and Sira Vegas at the ESEC/FSE 2015, entitled "Challenges of Conducting Software Engineering Experiments: Everything You Always Wanted to Know but Were Afraid to Ask" (13 attendees).

After its success at ICSE 2016 (it had 14 participants; many more than the other tutorials given at the same time) and ICSE 2017 (more than 20 participants, a higher number than the previous ICSE) and given the continuous increasing interest of the SE community in Experimental SE, we have decided to propose it this year again.

ACKNOWLEDGMENTS

Research funded by the Spanish Ministry of Economy and Competitiveness research grant TIN2014-60490-P.

REFERENCES

[1] H. Pashler, E.J. Wagenmakers. Editors' Introduction to the Special Section on Replicability in Psychological Science: A Crisis of Confidence? *Perspectives on Psychological Science*, 7(6):528–530, 2012.

[2] M. Shepperd, D. Bowes, T. Hall. Researcher Bias: The Use of Machine Learning in Software Defect Prediction. *IEEE Transactions on Software Engineering*, 40(6):603–616, 2014.

[3] D.I.K. Sjøberg, J.E. Hannay, O. Hansen, V.B. Kampenes, A. Karahasanovic, N.K. Liborg, A.C. Rekdal. A survey of controlled experiments in software engineering. *IEEE Transactions on Software Engineering*, 31(9):733–753, 2005