Information and Software Technology 56 (2014) 1063–1075

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Investigating the use of duration-based moving windows to improve software effort prediction: A replicated study

Chris Lokan[a,*], Emilia Mendes[b]

[a] School of Engineering & Information Technology, UNSW Canberra, Canberra, Australia
[b] Software Engineering Research Laboratory, Blekinge Institute of Technology, Karlskrona, Sweden

ARTICLE INFO

Article history:
Available online 12 March 2014

Keywords:
Chronological splitting
Effort estimation
Regression-based estimation models
Moving window

ABSTRACT

Context: Most research in software effort estimation has not considered chronology when selecting projects for training and testing sets. A chronological split represents the use of a projects starting and completion dates, such that any model that estimates effort for a new project *p* only uses as training data projects that were completed prior to *p*'s start. Four recent studies investigated the use of chronological splits, using moving windows wherein only the most recent projects completed prior to a projects starting date were used as training data. The first three studies (S1–S3) found some evidence in favor of using windows; they all defined window sizes as being fixed numbers of recent projects. In practice, we suggest that estimators think in terms of elapsed time rather than the size of the data set, when deciding which projects to include in a training set. In the fourth study (S4) we showed that the use of windows based on duration can also improve estimation accuracy.

Objective: This papers contribution is to extend S4 using an additional dataset, and to also investigate the effect on accuracy when using moving windows of various durations.

Method: Stepwise multivariate regression was used to build prediction models, using all available training data, and also using windows of various durations to select training data. Accuracy was compared based on absolute residuals and MREs; the Wilcoxon test was used to check statistical significances between results. Accuracy was also compared against estimates derived from windows containing fixed numbers of projects.

Results: Neither fixed size nor fixed duration windows provided superior estimation accuracy in the new data set.

Conclusions: Contrary to intuition, our results suggest that it is not always beneficial to exclude old data when estimating effort for new projects. When windows are helpful, windows based on duration are effective.

1. Introduction

Models for estimating software development effort are commonly built and evaluated using a set of historical projects [1]. The usual approach involves separating the data into a training set (from which a model is built) and a testing set (with which the model's accuracy is assessed). An important question is which projects to use as training data to build the model: should it be all of them, or a subset that seems particularly relevant?

Four recent studies—S1 [2], S2 [3], S3 [4], and S4 [5]—examined this issue by investigating the use of a chronological split taking into account a project's age. A chronological split represents the use of a project's starting and completion dates, such that any model that estimates effort for a new project *p* only uses as their training set projects that have been completed prior to *p*'s starting date.

These studies' research question was whether the use of a training set containing only the most recent past projects, i.e. a window of recent projects, would lead to more accurate predictions when compared to using the entire history of past projects completed prior to the starting date of a new project.

The intuition behind this question is that removing "noise", by discarding older projects that do not reflect current practice, is more beneficial than retaining a larger data set from which to learn.

The first three studies defined window size as being a fixed number of projects, i.e. the window contained the *N* most recently completed projects in the training set. S1 investigated the issue

[*] Corresponding author. Tel.: +61 262688060; fax: +61 262688581.
E-mail addresses: c.lokan@adfa.edu.au (C. Lokan), emilia.mendes@bth.se (E. Mendes).

http://dx.doi.org/10.1016/j.infsof.2014.02.008
0950-5849/© 2014 Elsevier B.V. All rights reserved.

using estimates based on models built by applying stepwise regression, and a dataset of single-company projects from the International Software Benchmarking Standards Group ("ISBSG") data repository.[1] S2 investigated the same issue, but with a different estimation technique (estimation by analogy) and different data (two single-company datasets from the PROMISE repository). S1 found that using a window could improve accuracy significantly, while S2 did not. S3 then compared stepwise regression and estimation by analogy directly on the same data set (the one used in S1), finding again that using a window could improve accuracy significantly, and that the effect of the window was stronger with regression.

An alternative is to define the window size in terms of duration: the training set contains projects whose development span occurred during the last *N* years or months. We suggest that this is a better reflection of how people think in practice: we have heard statements like "I would never consider data more than 10 years old", but not "I only consider the last X projects".

Duration-based windows were investigated in S4 [5], using the same data set as in S1 and S3. S4 found that windows could be helpful, and that differences between windows based on fixed numbers of projects and fixed duration were not statistically significant. This paper's research contribution is to extend S4, by investigating an additional data set and drawing comparisons between the two data sets and the different results obtained with the two data sets.

Similar to S4, we address the following research questions:

1. Assuming a project-by-project approach to effort estimation (meaning a separate training set is formed for each project to be estimated), is there a difference between the accuracy of estimates using prediction models that are built using all available data as the training set, and the accuracy of estimates using prediction models that are built considering only those projects whose development occurred during the last *N* months? The null hypothesis is that there is no difference, for all values of *N*.
2. Can insights be gained by observing trends in estimation accuracy as *N* varies?
3. How do these results compare with results based on fixed-size windows (windows containing a fixed number of projects)?

The remainder of the paper is organized as follows. Section 2 briefly summarizes related work. Section 3 describes the research method employed herein. Results are presented in Section 4, and discussed in Section 5. Section 6 details threats to validity, and finally our conclusions and directions for future work are presented in Section 7.

2. Related work

Research in software effort estimation has a long history [1]. However, consideration of chronology is rare.

Lefley and Shepperd [6], and Sentas et. al. [7], used chronological splitting as the basis for splitting their data into training and testing sets, when comparing effort and productivity models. Lokan and Mendes [8] compared a chronological split against a random split of data into training and testing sets, finding no significant impact on estimation accuracy.

More relevant to the present paper are studies in which training data was viewed as a portfolio that grew over time. Auer and Biffl [9] and Auer et al. [10] considered the effect of a growing portfolio in their research into estimation by analogy. They tracked changes in accuracy as the portfolio of completed projects grew. However, they did not consider the use of a window of projects. Lokan and Mendes [11,12] compared estimates based on a growing portfolio with estimates based on leave-one-out cross-validation, using two different data sets. In both cases, cross-validation estimates showed significantly superior accuracy.

To the best of our knowledge, moving windows were first mentioned in 2002, by Kitchenham et al. [13]. As one aspect of a broader study, they considered a growing data set and whether a moving window should be used. They found that when they divided their data into four subsets by start date, the regression models relating size to effort changed between the subsets. As a result they argued that old projects should be removed from the data set as new ones were added, so that the size of the data set remained constant. They recommended that the estimate for a given project should be based on the most recent 30 projects.

In S1 [2], Lokan and Mendes studied the use of moving windows with a data set of 228 projects from a single organization, sourced from the ISBSG repository. Training sets were defined to be the *N* most recently completed projects. They found that for small window sizes (small values of *N*), it was significantly worse to use a window than to retain all training data; for large window sizes it was significantly better to use a window (in terms of magnitude of relative error, though not in terms of absolute residuals). For this particular data set, the best window size seemed to be around $N = 75$.

In S2 [3], Amasaki et al. also investigated different-sized moving windows. They used a different estimation technique than in S1 (estimation by analogy), and studied different data sets (sourced from the PROMISE repository [14]: one of the datasets used was the same one employed by Kitchenham et al. [13]; the other was made available by Maxwell [15]). They found that using windows improved the average values of accuracy statistics, although the improvements were not statistically significant.

In S3 [4], Amasaki and Lokan investigated moving windows using both regression and estimation by analogy, on the data set used in S1. They found ranges of window sizes for which it was significantly better to use a window, with both regression and estimation by analogy. The effect of using a window was stronger with regression. Some differences in research method meant that the results could not be compared directly with S1 (because an extra independent variable was considered in S3) or S2 (because more neighbors and more combinations of potential independent variables were considered in S3). Later, in S4 Lokan and Mendes [5] employed the same dataset used in S1 and S3 to investigate the effect on accuracy when using moving windows of various durations to form training sets on which to base effort estimates. Their results showed that the use of windows based on duration can affect the accuracy of estimates (a window of about three years of duration appears the best choice); however to a lesser extent than windows based on a fixed number of projects.

MacDonell and Shepperd [16] investigated moving windows as part of a study into how well data from prior phases in a project could be used to estimate later phases. They found that accuracy was better when a moving window of the 5 most recent projects was used as training data, rather than using all completed projects as training data.

Turhan [17] describes several forms of "dataset shift", whereby training data differs from testing data. Changing data characteristics over time is one type of dataset shift.

As previously stated, this paper extends S4 via the analysis of another single-company dataset. Given that we aim to compare the results from this study with those in S4, we include herein the analysis previously carried out in S4 (using the ISBSG database) to facilitate the comparison.

[1] http://www.isbsg.org.