Investigating the Use of Chronological Splitting to Compare Software Cross-company and Single-company Effort Predictions

Chris Lokan¹, Emilia Mendes²

¹ School of IT&EE, UNSW@ADFA, Canberra ACT 2600, Australia
c.lokan@adfa.edu.au

² Computer Science Department, The University of Auckland, Private Bag 92019, Auckland, New Zealand
emilia@cs.auckland.ac.nz

CONTEXT: Numerous studies have investigated the use of cross-company datasets to estimate effort for single-company projects; however to date only one has compared the effect of using a chronological split instead of a random split to assign projects to a training set and a validation set, finding no significant differences.

OBJECTIVE: The aim of this study is to extend [15] using a project-by-project chronological split, and also to investigate how this type of split affects the results when comparing within- to cross-company effort estimation.

METHOD: Chronological splitting was compared with two forms of cross-validation. Here a more realistic form of chronological splitting than the one used in [15] is investigated, in which a validation set contains a single project, and a regression model is built from scratch using as training set the set of projects completed before the validation project’s start date. We used 228 single-company projects and 678 cross-company projects from the ISBSG Release 10 repository.

RESULTS: We obtained contradictory results when comparing cross- to single-company predictions for single-company projects. First, when results were compared using absolute residuals there were no differences between cross- and single-company predictions, or between techniques. However, when using z values, chronological splitting favoured cross-company models, and cross-validation (both types) favoured single-company models.

CONCLUSIONS: Results were promising when using project-by-project splitting because: i) they favoured cross-company models; and ii) this type of splitting mimics an effort estimation scenario in a real environment. However, these results were obtained using z values only. Therefore we urge future studies comparing prediction models to document results obtained using both z values and absolute residuals, such that a full picture can be provided.

Keywords: chronological split, effort estimation, software projects, cross-company estimation models, single-company estimation models, regression-based estimation models.

1. INTRODUCTION

Numerous software companies find it difficult to accumulate data on their past finished projects, and yet, to remain competitive, they must provide project effort estimates that are accurate. This situation has motivated researchers to investigate the use of cross-company project data to estimate effort for projects that belong to a single-company, where this company’s projects are omitted from model construction [10]. Numerous studies have been conducted to date; however without converging results.

Some have found that cross-company models did not present significantly worse prediction accuracy than single-company models [1],[3],[20],[23]. Some have found that cross-company models did present significantly worse prediction accuracy than single-company models [7],[8],[10],[19],[23]. Others were inconclusive for various reasons.

The normal approach in these studies – and indeed in almost all work in software engineering that builds effort estimation models from historical data – involves separating the data into a training set (from which the model is built) and a testing set (used to assess a model’s accuracy). When using leave-one-out cross-validation, each of the N projects in the data set is estimated in turn using a model built from the other N-1 points. Alternatively, the data can be allocated randomly or in a stratified way into a single training set (usually of about two thirds of the data) and a single testing set. Bootstrap and k-fold validation fall somewhere in between, involving multiple repetitions of model building and validation on different training and testing sets.

Almost invariably, the assignment of projects to training and testing sets is done without regard to the completion date of the projects. This makes it very likely (in leave-one-out cross-validation it is certain) that the data used to build a model to estimate effort for a given project p include projects that were completed after p was finished. However, in a real scenario, a developer building a model to estimate effort for a new project could only consider a set of projects already completed. In other words, future projects could not be considered. To reflect this reality
when building estimation models from historical data, the allocation of projects to training and testing sets should be based on a chronological split. Given this has not been the practice used in previous studies comparing cross- to single-company predictions, we conjectured as to whether the application of a more realistic approach to the estimation process using a chronological splitting could influence the results.

Looking at this issue, Lokan and Mendes' [15] recently investigated the use of a dataset-based chronological splitting (detailed later) to compare cross- and single-company predictions. They found that the use of a chronological splitting, and the use of two different dates to split the data chronologically, did not seem to influence the results when comparing single- to cross-company estimates. This paper (S2) extends [15] (S1) and investigates further whether using a chronological split influences the accuracy of the estimation models produced. S2 uses a different chronological splitting and cross-validation approach from those used in S1, as detailed below:

The differences between S1 and this study (S2) are as follows:
1. S1 used the same approach to chronological splitting employed in [20], where a splitting date \( d \) was chosen and used such that both cross- and single-company data sets only contained projects completed prior to \( d \), and the validation set only contained projects started at or after \( d \). The validation set used was an independent single-company holdout sample. S1 used two different splitting dates. S2 used a Project-by-project chronological splitting. Here, predictions for the SC projects are obtained by selecting one SC project at a time, and using as its training set the set of all SC projects completed up to that project’s starting date. Predictions for the SC projects using the CC data are obtained in a similar way; but this time the training set comprises CC projects completed up to the SC project’s starting date. No independent holdout samples are used in S2.
2. S1 used as benchmark cross-company & single-company data sets, and a single-company holdout sample containing randomly allocated projects. S2 used as benchmark leave-one-out and leave-two-out cross-validations, applied to both cross- and single-company datasets.
3. S1 built each of the cross-company regression models without taking into account the availability of data for the projects in the holdout samples. Therefore if a cross-company model provided a term for which there was no corresponding data available in the hold-out sample’s project, that term was replaced by ‘1’. S2 used the same approach used in S1, and, in addition, also built models considering the availability of data in the project for which effort was to be estimated. Only these latter results are presented since overall they presented superior predictions.
4. S1 considered both Web and non-Web projects in either their single- or cross-company data sets. S2 only considered non-Web projects since they wanted to isolate any effects the use of Web projects would have upon the results.
5. S1 removed most but not all cross- and single-company projects whose normalised effort differed from recorded effort. Their exception to this rule was one of the randomly allocated single-company training sets, where projects in which normalised effort differed from reported effort by no more than 8% were also kept. This led to the use of data on 1191 projects, drawn from Release 10 of the ISBSG data repository, where 450 came from a single company, and 741 came from other companies. Conversely, S2 removed all cross- and single-company projects whose normalised effort differed from recorded effort (meaning that the recorded effort was the actual effort across the whole life cycle), and where projects were Web-based. This led to the use of data on 909 projects, also drawn from Release 10 of the ISBSG data repository, where 231 came from a single company, and 678 came from other companies.
6. S1 used the manual stepwise procedure proposed by Kitchenham [9] to select, for each of the regression models, the independent variables to be used by that model. Once the independent variables were selected, they were used as input to fit a linear regression model. S2 used an automated stepwise regression procedure. In both cases the choice of multivariate regression was motivated by it being the single technique employed in all previous studies of cross-company and single-company models, where it either provided the best accuracy or was amongst the best [11].
7. S1 carried out all the data analysis using SPSS v15. S2 used the R statistical language, where all the model building and cross-validation analyses, and statistical significance comparisons were automated.

The research questions addressed by this study are as follows:
1. Using a project-by-project chronological splitting for both cross- and single-company projects:
   a) How successful is a cross-company model at estimating effort for projects from a single company, when the model is built from a data set that does not include projects from that company?
   b) How successful is a cross-company model, compared to a single-company model?
2. Using a leave-one-out cross-validation approach:
   a) How successful is a cross-company model at estimating effort for projects from a single company, when the model is built from a data set that does not include projects from that company?
   b) How successful is a cross-company model, compared to a single-company model?
3. Using a leave-two-out cross-validation approach:
   a) How successful is a cross-company model at estimating effort for projects from a single company, when the model is built from a data set that does not include projects from that company?
   b) How successful is a cross-company model, compared to a single-company model?

---

\(^1\)Also available from http://www.cs.auckland.ac.nz/~emilia/EASE/
4. Does the use of a project-by-project chronological split, instead of a leave-one-out cross-validation, affect the accuracy of the models?
5. Does the use of a project-by-project chronological split, instead of a leave-two-out cross-validation, affect the accuracy of the models?
6. Does the use of a leave-two-out cross-validation, instead of a leave-one-out cross-validation, affect the accuracy of the models?

The main contribution of this paper is therefore to extend S1 by considering further the impact of chronological splitting on the accuracy of the cross- and single-company models built. This paper uses a more realistic approach to chronological splitting than S1 because a project-by-project chronological split mimics much more closely what would happen in a real scenario in industry than simply using a single date as training and validation set splitting criterion.

The remainder of the paper is organised as follows: Section 2 briefly summarizes related work. Section 3 describes the research method employed in this study. Results are presented in Section 4, and research questions are addressed in Section 5. Finally, threats to the validity of our results are discussed in Section 6, and conclusions and directions for future work are presented in Section 7.

2. RELATED WORK

2.1 Cross- versus Single-company Models
One systematic review has been published on how the accuracy of cross-company models compares with that of single-company models [11]. They found ten studies. Four further papers have since addressed the question [14], [20],[23],[25]. The conclusions of the studies vary, and it is not clear what characteristics of the data sets and analysis methods affect the outcome. Kitchenham et al. [11] note that there are no consistent patterns concerning quality controls on data collection, the quality of the overall study, the size metrics used, the procedure used to build the single-company model, or the strength of the cross-company relationship.

If there is a trend emerging, it is to do with the homogeneity of the data. Kitchenham et al. [11] also note that all studies in which single-company models were significantly better than cross-company models used small single-company data sets; they conjecture that large data sets from a single company may be an indication of the size of the company and the homogeneity of the data set. The range of effort values in the single-company data set, compared to the range in the cross-company data set, also seems important: the greater the difference, the less likely it is that the cross-company model will be accurate for single-company projects.

Premraj and Zimmerman [25] studied the issue of homogeneity by grouping projects by business sector before building cross-company models. While their results varied for different companies, they concluded that it is better to train models using only homogeneous data rather than all data available.

Lokan and Mendes [14] applied Mendes et al.’s (S2a) [21] experimental procedure to the ISBSG database version used in Jeffery et al. (S1a) [8], to assess if differences in experimental procedure would have contributed towards the different results of [8] and [21]. This work was later extended [20] by applying the experimental procedure of S1a to the data set used in S2a. By investigating the effect of all the variations between S1a and S2a, they concluded that differences in data preparation and analysis procedures did not affect the outcome of the analysis. Thus, the different results of S1a and S2a were due to fundamental differences in the data sets.

Mendes et al. [23] replicated Mendes and Kitchenham [19] study using data on Web projects from the Tukutuku database, volunteered after [19] was conducted. Their results corroborated most of those in [19], except for one of their regression-based cross-company models, which showed significantly similar predictions to the single-company model, thus contradicting the findings from [19].

2.2 Use of Chronological Data
Chronological splitting has been applied as a data-splitting approach in several other domains. Examples include forecasting stock earnings [4], predicting coronary disease [17], and birdsong recognition [26].

To our knowledge, only three papers used chronological splitting to compare cross- and single-company predictions and different estimation techniques. Two studies - Lefley and Shepperd [13], and Sentas et al. [27], used a chronological splitting where a set date was chosen and used to split their data; however they did not investigate chronological splitting as a research question in its own right. Lokan and Mendes [15] investigated a research question in its own right the use of chronological splitting to compare cross- and single-company predictions. They found that the use of chronological split, and the use of two different dates to split the data chronologically, did not seem to improve the accuracy of either single- or cross-company estimates.

Another two studies have addressed a slightly different problem, using information from previous phases within a project to estimate effort in later tasks in the same project. MacDonell and Shepperd [16] studied effort distribution in major waterfall phases in 16 projects. They found the patterns of effort distribution between phases varied too much for this approach to work by itself, though it was helpful in conjunction with expert estimation. Abrahamsson et al. [1] were more successful using two projects developed using an extreme programming environment, finding that data from early iterations produced increasingly accurate effort estimates for later iterations.
The chronology of projects was also important in [24], which investigated changes in software development productivity over time. The focus in that paper was on characterising productivity in different years, rather than using past data to estimate future projects.

3. RESEARCH METHOD

3.1 Data set Description

The analysis presented in this paper was based on software projects from Release 10 of the ISBSG database. Release 10 contains data on 4106 projects, including 566 from the single company that we studied. ISBSG’s rules about confidentiality mean that we do not know the identity of the single company.

To form a data set suitable for our analysis, we removed projects according to the following criteria:

- Remove projects if they were not assigned a high data quality rating (A or B) by ISBSG.
- Remove projects for which the implementation date and/or overall project elapsed time is unknown.
- Remove projects if their size is measured in lines of code, or in a version of function points other than IFPUG, or in an outdated version of function points (size measured with an older version is not directly comparable with size measured with IFPUG version 4.0 or later). Also remove projects which measured size with an unspecified version of function points, and whose completion pre-dated IFPUG version 4.0.
- Remove projects for which the size in unadjusted function points is unknown.
- Remove projects with for which the development team effort (resource level 1) is unknown. Projects that reported the effort for other participants as well were retained, but only if they identified the development team effort separately, and only the development team’s effort was used in our analysis.
- Remove projects whose normalised effort differs from recorded effort. This should mean that the reported effort is the actual effort across the whole life cycle.
- Remove Web projects.

We finished with a set of 909 software projects, where 231 projects came from a single company, and 678 came from other companies. All have high data quality, and comparable definitions for size and effort. The project selection criteria we applied were carried out carefully in order to maximise comparability between projects. Preliminary analysis revealed that three of the single-company projects were extremely influential (according to the principles described in Section 3.3). They were removed permanently, leaving 228 single-company projects.

Release 10 of the ISBSG database provides data on 80 variables. We reduced the number of variables to a small set that we believed could potentially have an impact on effort. In addition, we also removed variables presenting a large number of missing points in the single-company data set. The remaining variables are presented in Table 1.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Scale</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effort</td>
<td>Ratio</td>
<td>Project effort in person hours</td>
</tr>
<tr>
<td>Ufp</td>
<td>Ratio</td>
<td>Application size in unadjusted function points</td>
</tr>
<tr>
<td>LangType</td>
<td>Nominal</td>
<td>Language type (e.g. 3GL, 4GL)</td>
</tr>
<tr>
<td>DevType</td>
<td>Nominal</td>
<td>New development, enhancement or re-development</td>
</tr>
<tr>
<td>Platform</td>
<td>Nominal</td>
<td>The type of hardware the system was developed for (mainframe, midrange, PC, multi-platform)</td>
</tr>
</tbody>
</table>

Summary statistics for the ratio-scale variables are presented in Table 2. We also include the project delivery rate (PDR), calculated as Effort/Ufp, to provide readers an additional way to compare cross- to single-company projects. This measure is often used to measure productivity, where high values indicate low productivity. Table 2 suggests differences in size and effort between the single- and cross-company data; however their productivity, measured as PDR, remains fairly similar.

<table>
<thead>
<tr>
<th>Variable</th>
<th>SC data - 228 projects</th>
<th>CC data - 678 projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>N</td>
<td>Mean</td>
</tr>
<tr>
<td>Ufp</td>
<td>228</td>
<td>496</td>
</tr>
<tr>
<td>Effort</td>
<td>228</td>
<td>4553</td>
</tr>
<tr>
<td>PDR</td>
<td>228</td>
<td>16.47</td>
</tr>
</tbody>
</table>

We do not present group statistics for the nominal variables since they varied between datasets, and their description would use a large amount of space.

---

2 www.isbsg.org
3.2 Project-by-project Chronological Splitting

Project-by-project chronological splitting was used both with the single- and cross-company data sets.

When employed with the single-company data set to obtain predictions for the single-company projects, the project-by-project chronological splitting was used as follows:

a. A project $p$ in the single-company data set was selected as the target project, for which estimated effort was to be obtained.

b. The starting date ($sd$) for $p$ was used to split the remaining single-company projects into two groups:
   i. **Completed**: projects that had finished prior to $sd$
   ii. **Active/future**: projects that were active or had not yet started at $sd$

c. The set of completed projects was used as the training set in order to build a regression model $R$.

d. Cook's distance [6] was used to determine whether any highly influential completed projects should be removed; if any were removed, $R$ was then refitted using the reduced data set.

e. $R$ was applied to $p$'s data in order to obtain an effort estimate for $p$.

f. Project $p$ was returned to the single-company data set.

g. Steps a to f were repeated until effort estimates were obtained for all the projects in the single-company data set.

When employed with the cross-company data set to obtain predictions for the single-company projects, the procedure was the same, except that cross-company projects replaced single-company projects as the training set.

3.3 Modelling Techniques

One model in this investigation was built manually (described in Section 4.1). All other models used in this investigation were built using an automated process programmed in the statistical programming language R.

The same basic procedure was followed in both the manual and automated processes:

- The first step in building every regression model was to ensure numerical variables were normally distributed. We used the Shapiro-Wilk test on both the training and testing sets to check if Effort and Ufp were normally distributed. Statistical significance was set at $\alpha = 0.05$. In nearly every case, Ufp and Effort were not normally distributed (the exception was early in the project-by-project chronological sequence, when the training sets were small). Therefore, Ufp and Effort were transformed to a natural logarithmic scale. The transformed variables' names were preceded by 'L'; so Effort became LEffort, and Ufp became LUfp.

- Models were built using multivariate stepwise regression. The manual process used the approach described in [9]. The automated process used backward stepwise regression.

- The size of the training set was always considered: no model was investigated involving more than $N/10$ independent variables, for a training set of $N$ projects.

- To verify the stability of the effort model, the manual process used the following steps [10]:
  - Use a residual plot showing residuals vs. fitted values to investigate if the residuals are random and normally distributed.
  - Calculate Cook’s distance values for all projects to identify influential data points. Any projects with distances higher than $3 \times (4/N)$, where $N$ represents the total number of projects, are immediately removed from the data analysis. Those with distances higher than $4/N$ but smaller than $3 \times (4/N)$ are removed temporarily in order to test the model stability, by observing the effect of their removal on the model. If the model coefficients remain stable and the goodness of fit improves, the highly influential projects are retained in the data analysis [14].

- The automated process used Cook’s distance in the same way, but did not use a residual plot as that is impractical when building large numbers of models automatically.

3.3 Prediction Accuracy Measures

To date the four measures most commonly used in software engineering to compare different effort estimation techniques have been [4]:

- Magnitude of Relative Error (MRE).
- Mean Magnitude of Relative Error (MMRE).
- Median Magnitude of Relative Error (MdMRE).
- Prediction at level $l$ (Pred($l$)).

MRE is defined as:

$$MRE = \frac{\vert e - \hat{e} \vert}{e}$$

where $e$ represents actual effort and $\hat{e}$ estimated effort. MMRE is the mean of all MREs. An alternative to the mean is the median, which also represents a measure of central tendency; however it is less sensitive to extreme values. The median of MRE values is called the MdMRE.

The Prediction at level $l$, also known as Pred($l$), measures the fraction of estimates that are within $l$% of the actual values. Suggestions have been made [4] that $l$ should be set at 25% and that a good prediction system should offer this accuracy level 75% of the time.
Although MMRE, MdMRE and Pred(l) are often used as evaluation criteria to compare different effort estimation techniques, Kitchenham et al. [11] showed that MMRE and Pred(l) are respectively measures of the spread and kurtosis of z, where \( z = \hat{e} / e \). They suggested the use of boxplots of z and boxplots of the residuals \( e - \hat{e} \) as useful alternatives to simple summary measures since they can give a good indication of the distribution of residuals and z and can help explain summary statistics such as MMRE and Pred(25).

Kitchenham et al. [11] also suggest the use of the Magnitude of Relative Error relative to the Estimate (EMRE) as a comparative measure. EMRE uses the estimate, not the actual value, as the divisor:

\[
\text{EMRE} = \frac{|e - \hat{e}|}{\hat{e}}
\]  

(2)

We can also calculate the mean EMRE (MEMRE) and Median EMRE (MdEMRE).

We use MMRE, MdMRE, Pred(25), MEMRE and MdEMRE to compare the effort models used in this study, and test for significant differences in absolute residuals and z.

To compare the statistical significance of predictions we used a non-parametric test – the Wilcoxon paired-samples test – and set the statistical significance at 5%. All calculations were carried out using the statistical language R.

4. RESULTS

The single-company data set contained data on 228 projects. However, with project-by-project chronological splitting we only built single-company regression models whenever there were at least 12 single-company projects that had been completed prior to start of the single-company project for which effort was to be estimated. Therefore we were only able to obtain predictions for 206 of the 228 single-company projects. For comparability, all other models were built and evaluated using the same 206 projects.

4.1 Cross-company Model

The cross-company models used as part of the project-by-project chronological splitting procedure will not be described in this Section as there were 206 different models that were automatically fit using the statistical language R.

Therefore, the only cross-company model that will be detailed here is the one to be compared with predictions obtained from applying leave-one-out and leave-two-out cross-validations in the single-company data set. In the cross-company data set, the categorical variables Platform, DevType, and LangType had four, three and three levels respectively. They were each replaced by three, two and two dummy variables respectively. In addition, Effort and Ufp were transformed to a natural logarithmic scale to approximate a normal distribution. Table 3 presents the final set of variables used to build the cross-company model.

<table>
<thead>
<tr>
<th>TABLE 3: Variables used to build the cross-company model</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEffort</td>
</tr>
<tr>
<td>LUfp</td>
</tr>
<tr>
<td>DevTypeNew</td>
</tr>
<tr>
<td>DevTypeRedev</td>
</tr>
<tr>
<td>PlatformMulti</td>
</tr>
<tr>
<td>PlatformPC</td>
</tr>
<tr>
<td>PlatformMR</td>
</tr>
<tr>
<td>LangTypeOther</td>
</tr>
<tr>
<td>LangType4GL</td>
</tr>
</tbody>
</table>

The best cross-company model, based on the full set of 678 projects, selected three significant independent variables: LUfp, LangType4GL and PlatformPC. Its adjusted \( R^2 \) was 0.439. The residual plot showed several projects that seemed to have very large residuals. This was also confirmed using Cook’s distance. Eight projects had their Cook’s distance above three times the cut-off point (4/678), so they were permanently removed from further analysis and the model refitted. This new model presented an adjusted \( R^2 \) of 0.463. This time there were no projects with Cook’s distance >= 3*(4/670); however 44 projects presented Cook’s D >= (4/670). To check the model’s stability, a new model was generated without the 44 projects that presented high Cook’s distance, giving an adjusted \( R^2 \) of 0.623. In the new model the independent variables remained significant however their coefficient values were dissimilar to those in the previous model. Therefore, the 44 high influence data points were removed. The final equation for the cross-company data set (Equation 3), based on 626 projects, is described in Table 4.

<table>
<thead>
<tr>
<th>TABLE 4: Best Cross-company model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unstandardized Coefficients</td>
</tr>
<tr>
<td>B</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>(Constant)</td>
</tr>
<tr>
<td>LUfp</td>
</tr>
<tr>
<td>LangType4GL</td>
</tr>
<tr>
<td>PlatformPC</td>
</tr>
</tbody>
</table>

The best cross-company model, based on the full set of 678 projects, selected three significant independent variables: LUfp, LangType4GL and PlatformPC. Its adjusted \( R^2 \) was 0.439. The residual plot showed several projects that seemed to have very large residuals. This was also confirmed using Cook’s distance. Eight projects had their Cook’s distance above three times the cut-off point (4/678), so they were permanently removed from further analysis and the model refitted. This new model presented an adjusted \( R^2 \) of 0.463. This time there were no projects with Cook’s distance >= 3*(4/670); however 44 projects presented Cook’s D >= (4/670). To check the model’s stability, a new model was generated without the 44 projects that presented high Cook’s distance, giving an adjusted \( R^2 \) of 0.623. In the new model the independent variables remained significant however their coefficient values were dissimilar to those in the previous model. Therefore, the 44 high influence data points were removed. The final equation for the cross-company data set (Equation 3), based on 626 projects, is described in Table 4.
LEffort = 3.713 + 0.790LUfp – 0.391 LangType4GL – 0.369 PlatformPC \hspace{1cm} (3)

which, when transformed back to the raw data scale, gives the Equation:

\[ \text{Effort} = 40.976 \ Ufp^{0.790} e^{-0.391 \text{LangType4GL}} e^{-0.369 \text{PlatformPC}} \hspace{1cm} (4) \]

The residual plot and the P-P plot for the final model are presented in Figure 1. P-P Plots (Probability plots) are normally employed to verify whether the distribution of a variable matches a given distribution, in which case data points gather around a straight line. The distribution that has been checked here is the normal distribution, and Figure 1 suggests that the residuals are normally distributed.

![Normal P-P Plot of Regression Standardized Residual](image)

**FIGURE 1:** Residuals and PP Plot for Cross-company model

### 4.2 Cross-company Models applied to Single-company projects

Two different sets of prediction accuracy statistics are presented in Table 5:

1. Predictions based on the application of project-by-project chronological split to each of the 206 automatically fit cross-company models (CC1), applied to each of the 206 single-company projects. Each of the cross-company models was built from scratch for each single-company project, using a procedure automated in the statistical language R.

2. Predictions based on the application of the cross-company model represented by Equation 4 (CC2) to the same 206 single-company projects. (The absolute residuals and z values resulting from CC2 will later be compared to those obtained from leave-one-out and leave-two-out cross-validations.)

<table>
<thead>
<tr>
<th>Cross-company Effort prediction models</th>
<th>CC1</th>
<th>CC2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Accuracy statistics</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean absolute residuals</td>
<td>2,802</td>
<td>3,013</td>
</tr>
<tr>
<td>Mean MRE (MMRE)</td>
<td>0.982</td>
<td>1.477</td>
</tr>
<tr>
<td>Mean EMRE (MEMRE)</td>
<td>1.640</td>
<td>0.872</td>
</tr>
<tr>
<td>Mean z</td>
<td>1.391</td>
<td>2.146</td>
</tr>
<tr>
<td>Pred (0.25)</td>
<td>0.184</td>
<td>0.180</td>
</tr>
<tr>
<td><strong>Median absolute residuals</strong></td>
<td>1,399</td>
<td>1,819</td>
</tr>
<tr>
<td>Median MRE (MdMRE)</td>
<td>0.650</td>
<td>0.649</td>
</tr>
<tr>
<td>Median EMRE (MdEMRE)</td>
<td>0.678</td>
<td>0.626</td>
</tr>
<tr>
<td>Median z</td>
<td>0.911</td>
<td>1.395</td>
</tr>
</tbody>
</table>

Overall both CC1 and CC2 presented poor prediction accuracy (high mean and median MREs and EMREs, and low Pred(.25) values). However, their median MREs and EMREs, and Pred(.25) were very similar, suggesting that their prediction accuracy was overall similar. This trend was confirmed by the Wilcoxon paired-samples test when applied to absolute residuals. However, when the Wilcoxon test was applied to z values the trends changed, and CC1 was found to present significantly superior predictions than CC2.

### 4.3 Single-company Models

Three different sets of prediction accuracy statistics are presented in Table 6:

1. Predictions based on project-by-project chronological splitting, using single-company data, for each of the same 206 single-company projects. Each of the single-company models was built from scratch, using a procedure automated in the statistical language R.
2. Predictions based on the use of leave-one-out cross-validation applied to the 206 single-company projects, where each of the single-company regression models (SC2) was built from scratch, using a procedure automated in the statistical language R.

3. Predictions based on the use of leave-two-out cross-validation applied to the 206 single-company projects, where each of the single-company regression models (SC3) was built from scratch, using a procedure automated in the statistical language R.

The training set was small for early projects in the chronological sequence, but by the end of the sequence it was almost the complete data set.

**TABLE 6:** Prediction accuracy Statistics for single-company models applied to single-company projects

<table>
<thead>
<tr>
<th>Accuracy statistics</th>
<th>Single-company effort prediction models</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SC1</td>
</tr>
<tr>
<td>Mean absolute residuals</td>
<td>3.015</td>
</tr>
<tr>
<td>Mean MRE (MMRE)</td>
<td>1.528</td>
</tr>
<tr>
<td>Mean EMRE (MEMRE)</td>
<td>0.940</td>
</tr>
<tr>
<td>Mean z</td>
<td>2.168</td>
</tr>
<tr>
<td>Pred(25)</td>
<td>0.209</td>
</tr>
<tr>
<td>Median absolute residuals</td>
<td>1.663</td>
</tr>
<tr>
<td>Median MRE (MdMRE)</td>
<td>0.632</td>
</tr>
<tr>
<td>Median EMRE (MdEMRE)</td>
<td>0.620</td>
</tr>
<tr>
<td>Median z</td>
<td>1.323</td>
</tr>
</tbody>
</table>

Overall all single-company models presented poor prediction accuracy (high mean and median MREs and EMREs, and low Pred(.25) values). However, similarly to the two cross-company models, their median MREs and EMREs were very similar, and Pred(25) was also very similar for SC2 and SC3. These trends suggest that their prediction accuracy was overall similar, trend confirmed by the Wilcoxon paired-samples test when applied to absolute residuals. However, when the Wilcoxon test was applied to z values the trends changed: SC1 was found to present significantly worse accuracy than either SC2 or SC3, and SC2 and SC3 were found to present similar accuracy.

4.4 Comparing Single- to Cross-company Predictions

Table 7 shows the results for the statistical significance tests comparing predictions between single- and cross-company predictions. All results were obtained using the Wilcoxon paired-samples test. Absolute residuals indicate no differences in prediction accuracy between cross- and single-company models, when used to estimate effort for single-company projects. Conversely, z values show that cross- and single-company models provide significantly different predictions when used to estimate effort for single-company projects. If project-by-project chronological split is used, cross-company models present superior accuracy; otherwise single-company models are those that present superior accuracy.

These are very interesting results because they show that not only the measure used in the statistical significance test – z or absolute residuals – can affect the results, but also the mechanism used to obtain the predictions (project-by-project chronological split, leave-one-out or leave-two-out cross-validation). To date [15] has been the only study that used z, in addition to absolute residuals, when carrying out statistical significance tests to compare predictions between cross- and single-company models when estimating effort for single-company projects. This means that the findings obtained to date could have been quite different if z had also been used. Therefore we urge other researchers to use both z values and absolute residuals when comparing models, so that a full picture can be provided.

The two cross-validation approaches did not present any significant differences.

**TABLE 7:** Comparing prediction accuracy between single- and cross-company models

<table>
<thead>
<tr>
<th>Cross-company model(s)</th>
<th>Single-company model(s)</th>
<th>Statistically significantly different based on z?</th>
<th>Statistically significantly different based on absolute residuals?</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC1</td>
<td>SC1</td>
<td>Yes, favours CC1</td>
<td>No</td>
</tr>
<tr>
<td>CC2</td>
<td>SC2</td>
<td>Yes, favours SC2</td>
<td>No</td>
</tr>
<tr>
<td>CC2</td>
<td>SC3</td>
<td>Yes, favours SC3</td>
<td>No</td>
</tr>
</tbody>
</table>

5. ANSWERING OUR RESEARCH QUESTIONS

Research questions 1a), 2a) and 3a) from Section 1 ask how accurate cross-company models are for single-company data. These questions are addressed in Table 5. None of the estimates obtained for the single-company projects using cross-company models indicate good prediction accuracy. MMRE values range between 0.982 and 1.477, which is considered poor (0.25 is considered “good” [4]). The same applies to Pred(25), which ranges from 0.180 to 0.184 (0.75 indicates a good prediction model).

Research questions 1b), 2b) and 3b) compare cross- and single-company models applied to single-company data. These questions are addressed in Table 7. Absolute residuals show no significant differences between any of the cross- and single-company predictions, when used to estimate effort for single-company projects. However,
based on z values, trends have changed drastically: i) when project-by-project chronological split is used, cross-
company models presented significantly superior predictions than single-company models when estimating effort
for single-company projects; ii) when cross-validation is used, single-company models present significantly
superior predictions than cross-company models when estimating effort for single-company projects.

To answer the fourth research question, comparing project-by-project chronological splitting with leave-one-out
cross-validation, we used the Wilcoxon paired-samples test to compare the z values and absolute residuals
between models CC1 and CC2 and models SC1 and SC2. For the cross- and single-company models, no
significant differences were found when based on absolute residuals. However z values presented contradicting
results: CC1 to present significantly superior prediction than CC2, and SC2 to present significantly superior
prediction than SC1. So the answer to question four is ‘yes’ if based on z values and ‘no’ otherwise.

Question five (comparing leave-two-out cross-validation, rather than leave-one-out cross-validation, with project-
by-project chronological splitting) had the same answer as question four: the answer to question five is ‘yes’ if
based on z values and ‘no’ otherwise.

To answer question six, comparing the two cross-validation approaches, we used the Wilcoxon paired-samples
test to compare the z values and absolute residuals between models SC2 and SC3. No significant differences
were found, either using z values or absolute residuals. Therefore the answer to this question is ‘no’.

It is noteworthy that estimates with cross-validation are more accurate than estimates based on project-by-
project chronological sequence. The reason for this could be that both types of cross-validation used in this paper
make use of a larger number of training set projects than when applying chronological splitting. The latter grows
over time and only approaches the full data set at the end. The significance of this result is that project-by-project
chronological estimation is much closer to how estimation is done in a real situation. If this result can be
generalised, it means that the accuracy obtained in studies based on cross-validation, or other validation
techniques that do not consider the sequence in which projects were completed, is probably too optimistic.

6. THREATS TO THE VALIDITY

This study has some limitations and threats to validity. First, the ISBSG repository is a convenience sample, and
does not represent a random sample of projects. Therefore these results are only applicable to those companies
that volunteered data to the ISBSG dataset, and companies that manage software projects similar to those used in
this study.

Second, most of the models were built automatically. Automating the process necessarily involved making some
assumptions, and the validity of our results depends on those assumptions being reasonable. For example, log
transformation is assumed to be adequate to transform numeric data to an approximately normal distribution;
residuals are assumed to be random and normally distributed without that being actually checked; when choosing
between two models in which all independent variables were significant, the one with higher adjusted $R^2$ is
assumed to be preferred; multicollinearity between independent variables is handled automatically in the stepwise
procedure used to build the manual model, but is not checked in the automated process. Based on our past
experience with manual model-building from ISBSG data, including the model described in Section 4.1 above, we
believe that these assumptions are acceptable. One would not want to base significant decisions on a single model
built automatically, without at least doing some serious manual checking. But for calculations such as leave-one-
out/leave-two-out cross-validation, or project-by-project chronological estimation across a substantial data set, we
believe that the process here is reasonable.

7. CONCLUSIONS

This paper investigated the use of project-by-project chronological splitting when comparing cross- and single-
company prediction models, and compared it with two cross-validation approaches. Data from Release 10 of the
ISBSG repository was used to answer our research questions.

When comparing cross- and single-company predictions, we found no significant differences when absolute
residuals were used as the accuracy measure. With z as the accuracy measure, significant differences were found,
and they were contradictory. Using cross-validation, estimates for single-company projects were significantly more
accurate when based on models developed from the single company’s own data; using project-by-project
chronological splitting, estimates for the single-company projects were significantly more accurate when the
models were based on cross-company data rather than single-company data. This last result is surprising. Some
other studies have found single-company models to be no better than cross-company models, but this is the first
time that single-company models have been found to be worse than cross-company models. This highlights that
had previous studies comparing cross- to single-company predictions used z values, instead of absolute residuals
or MREs, the evidence gathered to date might have been quite different. Therefore we urge other researchers to
use both z values and absolute residuals when comparing models, so that a full picture can be provided.

We found no difference between the results from leave-one-out and leave-two-out cross-validation. Leave-two-
out cross-validation has the advantage of providing a distribution of estimates for each project, rather than a single
estimate; it has the disadvantage that it involves a great deal more computation, proportional to the square of the
size of the data set instead of linear in the size of the data set. Our results suggest that the computational cost of
leave-two-out cross-validation is not worthwhile.

Future work includes repeating this experimental approach using other data sets; particularly, more
homogeneous data sets, including Web projects. We are also interested in tracing the evolution of prediction
models and their accuracy as a training set grows. Another research question to be investigated is whether it is
REFERENCES.


