Survival Analysis in Open Development Projects*

Felipe Ortega and Daniel Izquierdo-Cortazar
GSyC/LibreSoft, Universidad Rey Juan Carlos, Madrid (Spain)
{jfelipe, dizquierdo}@gsyc.urjc.es

Abstract

Open collaborative projects, like FLOSS development projects and open content creation projects (e.g. Wikipedia), heavily depend on contributions from their respective communities to improve. In this context, an important question for both researchers and practitioners is: what is the expected lifetime of contributors in a community? Answering this question, we will be able to characterize these communities as an appropriate model can show whether or not users maintain their interest to contribute, for how long we could expect them to collaborate and, as a result, improve the organization and management of the project. In this paper, we demonstrate that survival analysis, a well-known statistical methodology in other research areas such as epidemiology, biology or demographic studies, is a useful methodology to undertake a quantitative comparison of the lifetime of contributors in open collaborative initiatives, like the development of FLOSS projects and the Wikipedia, providing insightful answers to this challenging question.

1 Introduction

Open collaborative initiatives exhibit some superior features over the aged, old-fashioned centralized approach to create intangible products. Some authors even support the thesis that decentralized structures in collaborative projects actually unleash the full power of innovation and productivity in virtual communities [2], [19], [20], and that they have even shaped the transformation of global relationships in business models and human relationships at large scale [17].

Nevertheless, a significant proportion of contributors to these virtual communities are volunteers. This raises a natural question: given that most contributors may simply give up the project anytime, does it have any consequence for the project continuity? Clearly, if we detect that, starting at a certain point in time, the average lifetime of contributors in a project begins to decrease, we would immediately infer that i) the project is rapidly losing contributors and ii) if we can not retain users for longer periods of time, or if the number of users joining the project does not increase, then it will unavoidably die due to the lack of contributions.

In this paper, we explain how we can use survival analysis, a statistical methodology already applied with successful results in other research areas, to undertake a quantitative analysis of open virtual communities in collaborative projects. We will focus on the analysis of FLOSS projects, and their comparison with the Wikipedia as the epitome of open content development at present time. We are interested in evaluating if there are common survivorship patterns between these two types of projects. We also want to show potential implications of these results regarding organizational and management decision-making processes in these communities (for example, influencing decisions about project maintainability or allocation of efforts). As far as we know, this is the first attempt to apply this technique in this research area, proving that it is feasible and useful for the aforementioned purposes.

The structure of the paper is as follows: in the next section, we recapitulate previous research works on related topics in this area. Then, we describe the methodological principles of survival analysis, and how we can apply this technique to undertake a quantitative study of survivorship in open collaborative projects. After that, we present some preliminary results, demonstrating the feasibility of this new approach. We turn then to explore the consequences of these results to assess the present state and future evolution of FLOSS projects and the Wikipedia, and we introduce possible further applications of this methodology that we intend to explore. Finally, we summarize the most important conclusions that we can extract from this research work.

2 Previous research

The study of social dynamics in FLOSS development projects has been a matter of research for a long time. For

*This work has been funded in part by the European Commission, under the FLOSSMETRICS (FP6-IST-5-033547), QUALOSS (FP6-IST-5-033547) and QUALIPSO (FP6-IST-034763) projects, and by the Spanish CICyT, project SobreSalto (TIN2007-66172).

example, in [15] Robles *et al*. find that the lifetime of volunteers participating in the Debian community presents strong differences depending on the tasks assumed by these individuals within the project. In [6] the time for volunteers to become core contributors to FLOSS projects has been measured to be in the meean 30 months since their first contribution. In [10], Michlmayr *et al*. also present a quantitative analysis of the evolution over time of participation of volunteers in large FLOSS projects, introducing the measure of *half-life* for the human resources of a project. They study the population of developers participating in a project and identify the point in time when only half of them are still active. In the case of Debian, the expected half-life was 7.5 years.

Finally, risk management during the software development and maintenance process has been a matter of research for a long time, as well [3]. Some of the risks affecting software development are related to human resources. In Demarco and Lister's classic *Peopleware* [5], several chapters are devoted to the management of such issues. Developer turnover is one of the most important factors to be considered.

Regarding the analysis of social dynamics in Wikipedia, from a quantitative point of view, the first research study was conducted by Voss [21]. He shows that the number of different authors per article follows a power law, while the number of distinct articles per author follows Lotka's law. Kittur *et al*. find in [7] that Wikipedia is receiving a growing number of contributions from a significant number of casual editors, who provide less than 10 edits before leaving the project. Nevertheless, subsequent analyses demonstrate that we can also identify a core of very active editors in the Wikipedia, who are responsible for the main share of the content production process [12]. Likewise, the concentration of contributions towards this core of very active editors remains very stable over time [13]. Almeida *et al*. also present a quantitative analysis of critical parameters describing the community of editors in Wikipedia [1]. They find that the exponential growth of Wikipedia is mainly driven by its quickly increasing user base, showing the importance of its open editorial policy for its current success. In a recently published study, [16] Spinellis and Louridas find that Wikipedia is likely to continue growing, following a preferential attachment pattern.

As far as we know, this is the first attempt to apply survival analysis to study open collaborative projects like FLOSS development projects and Wikipedia. Therefore, our aim is to contribute to these previous research providing a novel approach suitable for obtaining complementing information that will give us additional insights about the social dynamics of open collaborative projects.

3 Methodology

In this section, we introduce basic aspects of the theoretical framework of survival analysis. Then, we explain the implementation details of our study on FLOSS projects and Wikipedia.

3.1 An introduction to survival analysis

Survival analysis is a powerful, yet conceptually simple statistical methodology that allows to build empirical models for data analysis in which the variable of interest can be formulated as *time until an event occurs*. We have chosen *GNU R*, a popular statistical software package [14] to implement our survival analysis of FLOSS projects and Wikipedia. More precisely, we have used the extensive set of tools and procedures included in the *R* package *survival* [11], written by Terry Therneau and ported to *R* by Thomas Lumley. [8], [4], [9] and [18] provide good introductions to the theory survival analysis and practical examples using *R*. We present here a very brief introduction to the basic theory behind survival analysis, just to provide a minimum background framework to understand the results and conclusions that we will draw in the following sections.

In our study, we are interested in modelling the lifetime of contributors to FLOSS projects and the Wikipedia. To achieve this, we measure the period of time in which they have been collaborating with the virtual community. Fortunately, both FLOSS projects and the Wikipedia log all activity information for contributions received from their respective communities. These log files include unique identifiers for each contributor and timestamps for each contribution. However, while FLOSS projects usually impose strict access restrictions to their source code management repositories, Wikipedia accepts contributions from anonymous users to any of its language editions. In the case of anonymous Wikipedia contributors, only the IP address is registered in the log file. Thus, we can not discriminate individual anonymous contributors (for instance, due to NAT services masquerading several users behind the same IP address). For this reason, we have filtered contributions from anonymous users in this analysis.

In our study, we will refer to the following definitions:

- We consider as a **committer** a developer contributing to a FLOSS project, with privileges to perform commits to the source code management repository, and identified by a unique numeric ID.

- We consider a Wikipedia **editor** as a logged user performing any kind of content contribution to a certain article. Wikipedia editors are uniquely identified by a numeric ID and a user name (login).

- We identify the **event of interest** in our survival analysis as the **last registered contribution** from a committer or developer to the collaborative project. Thus, we define the **lifetime** of a FLOSS committer or a Wikipedia editor as the time (measured in days) elapsed from her first registered contribution to the project to the last one logged in the audit files[1].

- Let T denote a random variable, describing the *lifetime* (in days) of a committer in FLOSS development projects or an editor of Wikipedia articles. Its values are contained in $(0, \infty)$, and its continuous distribution is specified by its cumulative distribution function $F(t)$, (expressing the probability of any developer or contributor of having a lifetime value $T \leq t$), with probability density function $f(t)$.

- We define the **survival function** $S(t)$ as:

$$S(t) = 1 - F(t) = P(T > t) \tag{1}$$

  Thus, expressing the probability for a certain user or developer to stay in the community longer than some specified time t.

One of the most important advantages of survival analysis is that we do not need to wait until all subjects included in the trial reach the event of interest to estimate $S(t)$. Instead, we simply define the limits of our observation period, and then assign a boolean value indicating, for each subject, whether she was "dead" or "alive" at the end of the study. In survival analysis, this is called **censoring** data (more precisely, *right censoring*). In other studies, we may have other types of censoring as well. However, both the log files of source code management systems and the Wikipedia include a complete list of all contributions performed within the period of analysis, so we do not have to deal with other types of censoring here.

As a result, when dealing with right censoring we have to slightly modify our definition of lifetime:

- To deal with the *right censoring* information, we define the **observed lifetime** for a certain committer or editor as the time period elapsed from her first logged contribution in the registry archive to either her last logged contribution (if $censoring == True$) or else until the end date of the study (if the individual is still *alive* at the end date of the study, so that $censoring == False$).

  Finally, once we have computed the observation time for each subject, and the censoring information, we calculate an estimation of $S(t)$, using the *survfit* function included in the *survival* library of *GNU R*. We define here the mathematical model that it applies to estimate the survival function as follows:

- Let $r(t)$ be the number of cases at risk before time t, i.e., those subjects that are still alive in the trial before time t. If we define a set of intervals $I_i = [t_i, t_{i+1})$, covering $[0, \infty])$, then the probability p_i of surviving an interval I_i is:

$$p_i = \frac{[r(t_i) - d_i]}{r(t_i)} \tag{2}$$

  Where d_i is the number of deaths counted in interval I_i. Hence, the probability of surviving until t_i is:

$$P(T > t_i) = S(t_i) \approx \prod_{0}^{i-1} p_j \approx \prod_{0}^{i-1} \frac{r(t_i) - d_i}{r(t_i)} \tag{3}$$

  We note that the fraction in the last productory will be non-unity for intervals in which deaths occur.

- We define the **Kaplan-Meier estimate** of a survival curve $S(t)$, denoted as $\hat{S}(t)$, as a maximum likelihood estimate obtained with:

$$\hat{S}(t) = \prod \frac{r(t_i) - d_i}{r(t_i)} \tag{4}$$

The *survival* package in *GNU R* automatically computes the estimate of $S(t)$ using the Kaplan-Meier method. It can also depicts several survival curves corresponding to different projects in the same graphic, for comparison purposes.

3.2 Data sources and implementation details

For this analysis, we have retrieved information from two distinct sources. For FLOSS projects, we have used the database provided by the FLOSSMETRICS project [2]. This database provides information about contributions from committers in each project. For each commit performed into the system, we have a unique ID for the responsible committer, and a timestamp. With the Wikipedia, we have decided to analyze its top ten language editions, according to their total number of articles. We have retrieved the complete database dumps for each edition from the official Wikimedia Foundation repository [3]. Then, we use WikiXRay[4], a Python tool that we have developed to automated the quantitative analysis of any language edition of Wikipedia.

[1] This definition of lifetime does not deal with intermediate idle intervals, in which a committer or editor ceases her contribution just to take it up again later on. Nevertheless, we do not consider that this limitation may affect the validity of our results.

[2] http://melquiades.flossmetrics.org

[3] http://download.wikimedia.org

[4] http://meta.wikimedia.org/wiki/WikiXRay

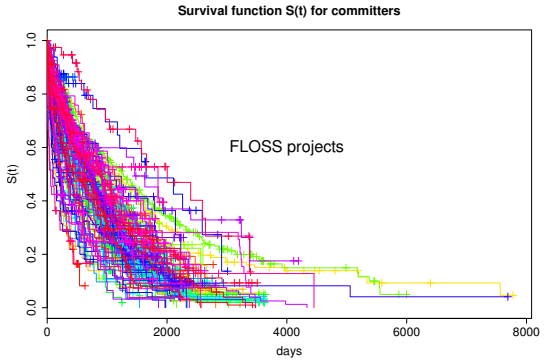

Figure 1. Estimated survival function of FLOSS projects

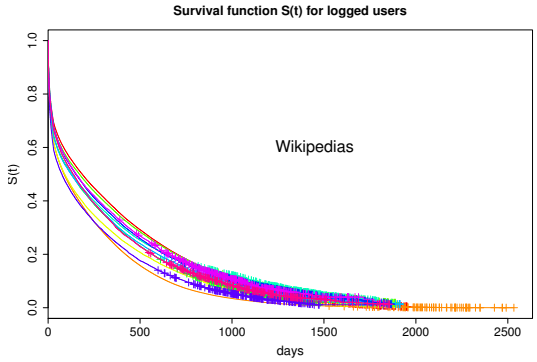

Figure 2. Estimated survival function of the top-ten language editions of Wikipedia

In order to select comparable projects from both fields, we have taken into account that the top-ten Wikipedias currently present complete data history for over five years. With this condition in mind, we have selected FLOSS projects from the FLOSSMETRIC database with more than 5 years of registered history in their source code management repositories.

Thus, we build, for each FLOSS project or Wikipedia language edition, a database table indicating: Committer or editor id, timestamp of the first contribution of the committer/editor, and timestamp of the last contribution of the committer/editor.

4 Preliminary results

In this section, we present some preliminary results of our comparison, based on survival analysis, of FLOSS projects and the Wikipedia. Figure 1 shows the estimated survival function for FLOSS projects from the FLOSSMETRICS database, with more than 5 years of history, while Figure 2 presents the estimated survival function for the top-ten language editions of Wikipedia.

As we can see, in FLOSS projects we have a lot more variability in survival functions than in the top-ten Wikipedias, where we obtain very similar values for all language editions. In order to have a quantitative comparison of a relevant survival parameter in both types of initiatives, we can compute two different parameters:

Median survival time: This is the median value of the survival time (in days) in a certain population, taken as the intersection of $S(t)$ with an horizontal line drawn at 0.5 on the plot of the survival curve. Alternatively, we can see this value as a *half-life* already introduced in [10].

Restricted mean survival time: This is the restricted mean survival time (in days) of the population of contributors in a certain initiative. We can also get the standard error of that mean value. It is called restricted because we have to consider situations in which the last observation(s) is not a death, so that the estimated survival curve does not goes down to zero. To solve that problem, the mean is computed as the mean survival value, restricted to the time before the last censoring.

Project	n	events	rmean	se(rmean)	median
abiword	68	59	792	103.1	448
alacarte	59	46	267	24.7	229
amsn	39	28	733	114.1	588
audacity	31	22	913	202.3	296
azureus	28	17	741	136.0	339
blender	67	38	1022	88.8	1132
boost	99	99	941	73.0	804
boost	113	77	1167	90.8	983
bug_buddy	173	123	925	71.9	708
bzflag	39	39	654	101.9	471

Table 1. Output of *survfit* call on FLOSS data with GNU R

We show in table 1 an example of the numerical output obtained with *GNU R* in FLOSS projects. For those projects with too many committers still alive (like `deskbar_applet`) the median value is ∞, though we still can parameterize the project with the restricted mean survival time. 2 shows the same information in the case of the top-ten Wikipedias.

Nevertheless, it is much more practical to get a visual summary of the obtained values. To achieve that, we draw in Figure 3 and Figure 4 the Kernel Density Estimation (KDE) summarizing the *median* and *rmean* survival times for FLOSS projects and the top ten Wikipedias, respectively. The KDE can be viewed as an interpolation of the histogram displaying the distribution of values of a certain

wiki	n	events	rmean	se(rmean)	median
de	113604	98688	364	1.383	185
en	728617	693645	228	0.380	87
es	47080	37571	282	2.332	90
fr	55723	43140	372	2.412	170
it	29138	23151	305	2.979	126
ja	42144	31563	355	2.973	131
nl	23873	19482	326	3.338	125
pl	23544	18509	351	3.545	150
pt	23106	19202	246	3.006	75
ru	15682	11233	356	5.324	153

Table 2. Output of *survfit* call on Wikipedia data with GNU R

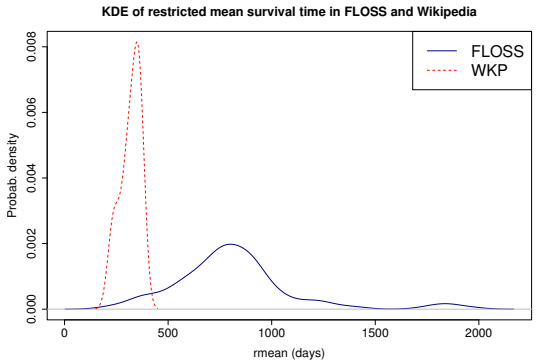

Figure 3. KDE of restricted survival means in FLOSS and the Wikipedia

variable. Hence, it lets us compare two or more probability density curves in the same graph.

As we can see, the majority of values of the restricted mean survival time in FLOSS projects varies in the interval $[500, 1000]$ (measured in days), while for the top-ten Wikipedias, the majority of values for the restricted mean survival time oscillate in the interval $[200, 400]$ days. For the median values, we have a very similar situation. We would like to remark that for the Wikipedias, the estimation of the restricted mean survival time is quite precise, with a value for the s.e. (rmean) between 0.38 and 5.34 days, showing that survival analysis improves its accuracy rapidly as the number of subjects considered in the trial increases.

5 Discussion and Conclusions

In this section, we discuss the preliminary results of our survival analysis presented in the previous section. We also show possible future applications for the analysis of the current state and evolution of open collaborative projects.

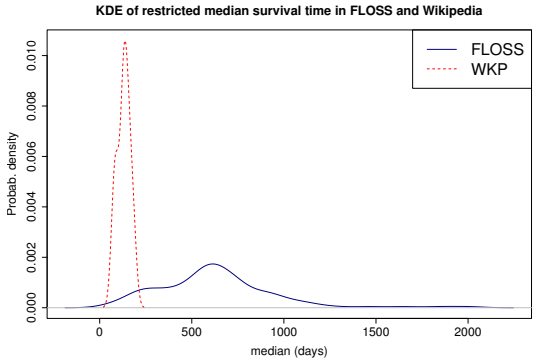

Figure 4. KDE of survival medians in the top-ten Wikipedias

5.1 Consequences of our preliminary results

Using basic survival analysis techniques, we have revealed some interesting patterns in FLOSS development projects and the Wikipedia. In the first place, we can promptly identify two common range of values for descriptive measures of survival time of contributors in both areas: the *restricted mean* and the *median* survival time. These descriptive parameters can be of great help to assess managers and coordinators of FLOSS projects or open content development projects for adopting wiser decisions about maintainability and planning of the expected evolution of the project in the next future.

On the one hand, software maintainability is one of the most critical tasks affecting the distribution of human resources in FLOSS projects, according to previous research [3]. In the same way, previous analyses also show that Wikipedia is a self-regulated production environment [13] which constantly needs new users to take the relay of the content creation effort. Survival analysis help us to identify the overall ratio of persistence of users in our production environments, therefore giving a simple, but informative, picture of the current hustle and bustle of users in our project.

On the other hand, survival analysis also lets us establish precise expectations about the future evolution of the project, under the assumption that the situation will not vary significantly from the current trend (something that we will have to explore in future research works). For instance, project coordinators may have an estimation of the number of new users that the project must attract in the near future to countermeasure the mortality index of the current population, if they are interested in maintaining the same human resources as in the recent past. We can also identify if certain users have surpassed the "usual" survivorship thresh-

old of the project. This result indicates a higher level of compromise with the daily work and goals of the initiative (which translates in a longer relationship with the project).

Finally, we can also compare distinct projects (like in the graphics and numerical comparisons presented in the previous section), searching for explanatory differences among them, perhaps indicating that the project will still probably have problems to start up (low survival values in its early history). On the contrary, we may also identify that a certain project as high survival values in its early and middle history, showing that its contributors, once they have joined, can be retained for longer periods (and we take that into account to plan future tasks in the project roadmap).

In the comparison undertaken in this paper, we have seen that FLOSS development projects present higher survival rates than the top-ten Wikipedias, showing very different sustainability patterns in each type of initiative.

5.2 Further applications

Despite obtaining interesting results concerning the analysis of communities of contributors in FLOSS projects and Wikipedia, we have just "scratched the surface" of the real potential applications of this statistical methodology in the context of networking collaborative projects.

In the first place, a natural follow-up of this research work will be the analysis of hazard function estimators for each project. This will provide complementary information about the instantaneous risk of contributors abandoning the initiative, possibly revealing additional behavioral patterns of great interest for project management and coordination. Secondly, survival analysis provides additional tools that would be of interest in this study, as well. For example, the *Cox proportional hazards models* can be applied to perform "multivariate regression like" studies on survival data, analyzing the influence of different covariates of interest on the survival function. This study can reveal important factors determining the behavior of contributors in the project, offering precise assessment about the motivations that moves contributors to stay in.

References

[1] R. Almeida, B. Mozafari, and J. Cho. On the evolution of wikipedia. In *International Conference on Weblogs and Social Media*, Boulder, Colorado, USA, March 2007.

[2] Y. Benkler. *The Wealth of Networks : How Social Production Transforms Markets and Freedom*. Yale University Press, May 2006.

[3] B. W. Boehm, editor. *Software risk management*. IEEE Press, Piscataway, NJ, USA, 1989.

[4] P. Dalgaard. *Introductory Statistics with R (Statistics and Computing)*, chapter 14. Springer, 2nd edition, August 2008.

[5] T. Demarco and T. Lister. *Peopleware : Productive Projects and Teams, 2nd Ed.* Dorset House Publishing Company, Incorporated, 1999.

[6] I. Herraiz, G. Robles, J. J. Amor, T. Romera, and J. M. González-Barahona. The processes of joining in global distributed software projects. In *GSD '06: Proceedings of the 2006 international workshop on Global software development for the practitioner*, pages 27–33, New York, NY, USA, 2006. ACM Press.

[7] A. Kittur, B. A. Pendleton, B. Suh, and T. Mytkowicz. Power of the few vs. wisdom of the crowd: Wikipedia and the rise of the bourgeoisie. In *Proceedings of the 25th Annual ACM Conference on Human Factors in Computing Systems (CHI 2007)*. ACM, April 2007.

[8] D. G. Kleinbaum and M. Klein. *Survival Analysis: A Self-Learning Text (Statistics for Biology and Health)*. Springer, 2nd edition, August 2005.

[9] J. Maindonald and J. Braun. *Data Analysis and Graphics Using R: An Example-based Approach (Cambridge Series in Statistical and Probabilistic Mathematics)*, chapter 8, pages 275–284. Cambridge University Press, December 2006.

[10] M. Michlmayr, G. Robles, and J. M. Gonzalez-Barahona. Volunteers in large libre software projects: A quantitative analysis over time. In S. K. Sowe, I. G. Stamelos, and I. Samoladas, editors, *Emerging Free and Open Source Software Practices*, pages 1–24. Idea Group Publishing, Hershey, Pennsylvania, USA, 2007.

[11] S. original by Terry Therneau and ported by Thomas Lumley. *survival: Survival analysis, including penalised likelihood.*, 2008. R package version 2.34-1.

[12] F. Ortega and J. M. Gonzalez-Barahona. Quantitative analysis of the wikipedia community of users. In *WikiSym '07: Proceedings of the 2007 international symposium on Wikis*, pages 75–86, New York, NY, USA, 2007. ACM.

[13] F. Ortega and J. M. Gonzalez-Barahona. On the inequality of contributions to wikipedia. In *Proceedings of the 41st Hawaiian International Conference on System Sciences (HICSS 2008)*, January 2008.

[14] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.

[15] G. Robles, J. M. Gonzalez-Barahona, and M. Michlmayr. Evolution of volunteer participation in libre software projects: evidence from Debian. In *Proceedings of the 1st International Conference on Open Source Systems*, pages 100–107, Genova, Italy, July 2005.

[16] D. Spinellis and P. Louridas. The collaborative organization of knowledge. *Commun. ACM*, 51(8):68–73, August 2008.

[17] D. Tapscott and A. D. Williams. *Wikinomics: How Mass Collaboration Changes Everything*. Portfolio Hardcover, April 2008.

[18] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*, chapter 13. Springer, September 2003.

[19] E. von Hippel. *The Sources of Innovation*. Oxford University Press, USA, September 1994.

[20] E. v. von Hippel. *Democratizing Innovation*. The MIT Press, April 2005.

[21] J. Voss. Measuring wikipedia. In *International Conference of the International Society for Scientometrics and Informetrics : 10th*. ISSI, July 2005.