Predicting mobile apps spread: An epidemiological random network modeling approach

J. Alegre-Sanahuja, J.-C. Cortés, Rafael-J. Villanueva,*
Instituto de Matemática Multidisciplinar, Edificio 8G piso 2
Universitat Politècnica de València, 46022 Valencia (Spain)
email: {juaalsa@doctor, jccortes@imm, rjvillan@imm}.upv.es
Telephone: +34 963 877 007 Ext. 88288

Francisco-J. Santonja,
Departamento de Estadística e Investigación Operativa
Universitat de València, 46100 Valencia (Spain)
email: francisco.santonja@uv.es
Telephone: +34 963 544 313

Abstract

The mobile applications business is a really big market growing constantly. In app marketing, a key issue is to predict future app installations. The influence of the peers seems to be very relevant when downloading apps. Therefore, the study of the evolution of mobile apps spread may be approached using a proper network model that considers the influence of peers. Influence of peers and other social contagions have been successfully described using models of epidemiological type. Hence, in this paper we propose an epidemiological random network model with realistic parameters to predict the evolution of downloads of apps. With this model, we are able to predict the behavior of an app in the market in the short term looking at its evolution in the early days of its launch. The numerical results provided by the proposed network are compared with data from real apps. This comparison shows that predictions improve as the model is feedback. Marketing researchers and strategy business managers can benefit from the proposed model since it can be helpful to predict app behavior over the time anticipating the spread of an app.

Keywords: Epidemiological Random Network, Mobile Apps Spread, Prediction, Behavior over the time.

---

*Author for correspondence

1 Introduction

In 2014, more than one billion smartphones were shipped [1] and the sales of smartphones grew 20% in the third quarter of 2014 [2] being millions the number of applications (apps) available for downloading by millions of users at app markets [3].

The app business is a really big market growing constantly and, in app marketing, one key issue is to predict future app installations. In the literature, there are studies that examine how the information spreads in implicit networks [4] or related with the network effect on information dissemination on social network sites as explained in [5]. Specifically about mobile apps, there are contributions that examine how the adoption (downloads) of the apps is influenced by others in their social network [6] and several approaches to model the proliferation growth of apps over the users [7, 8], where the network effects in users' app downloads have been studied.

In [8], the authors use a composite network model, comprised by a call-log network, a Bluetooth proximity network, a friendship network, an affiliation network plus a network that takes into account the exogenous factors, like app popularity. The data used to validate the model came from a sample of 55 students.

In [6], a sample of 180 students was surveyed about their usage of apps and the results were analyzed to examine the influence of social contacts on the use of apps. The results show this influence, being the most significant app advisors friends and family members.

In [7], a sample of 200 participants was considered. The data was collected via a passive data collection software platform that registered Bluetooth proximity hits by closeness and via surveys. One of the conclusions of this work is that one should be cautious in using declared friendship networks to infer the spreading of smartphone apps and for applying viral marketing strategies, since the face-to-face interaction seems to have stronger correlation with app diffusion.

Thus, all these previous contributions claim for:

- Social networks play an important role in consumers' decisions to download and use mobile apps [6].

- The adoption of mobile apps appears to spread via social contagion [6].

- People who spend more time in face-to-face interaction are more likely to share common apps [7].

- Face-to-face interaction has a strong correlation with app dissemination [7].

- There are strong network effects in app installation patterns [8].

However, in [7] and [8], the experiments are based on a small group of participants. In [6], it is recognized that the sample size for the study is relatively

small and the generalizability of the results is limited. To the best of our knowledge, there is a lack of experiments carried out with big network simulations and with multiple repetitions, comparing the results with data from real app markets.

Taking into account the above comments and how social contagion has been successfully studied using models of epidemiological type [9, 10], in this paper we propose an epidemiological random network model to estimate the evolution of downloads of the apps over a theoretical random network, analysing the potential spread of the apps and comparing the theoretical results with real data coming from real downloaded apps. One of the main contributions of this paper is that the network simulation was ran over a big theoretical random network of 1,000,000 members and was repeated 100,000 times with different sets of realistic parameters via computational methods. The resulting set of simulations from running the model multiple times provided us a bank of possible behaviors. This bank of potential behaviors allows us to predict the future behavior of an app looking at its evolution in the early days of its launch.

To conduct our study and compare our results with real data, we have followed the evolution of apps in a real Android app marketplace, where the exact number of downloads was available [11]. The monitorized apps have been randomly chosen, among free apps, and the results have been scaled for comparison with available real data. The scaling has to be done because the potential public for each app is very different. For instance, game apps have usually more downloads than specific apps.

Marketing researchers and strategy business managers can benefit from the model proposed in this paper since it can be helpful to predict the app behavior over the time and then anticipating the spread of an app.

This paper is organized as follows. In Section 2 we present the model building, model parameters, data used for modeling, simulations and comparison methods. Section 3 is devoted to present results and their discussion. Conclusions are drawn in Section 4.

2 Material and Methods

2.1 Model

Taking into account the network effects suggested by [6, 7, 8], we will build our model as a SIR-type epidemiological random network. The nodes will be the users and the edges will be the face-to-face relations between users. A user gets infected if he/she downloads the app; susceptible or not infected when he/she has never downloaded the app and, he/she will become recovered (and hence immune) when removes the app from his/her device. The number of initial infected nodes will be random; the infection rate will be defined based on the face-to-face relation between nodes and, the recovery rate will be based on the user app retention. A flow chart diagram for the SIR model applied in the paper is shown in Figure 1.
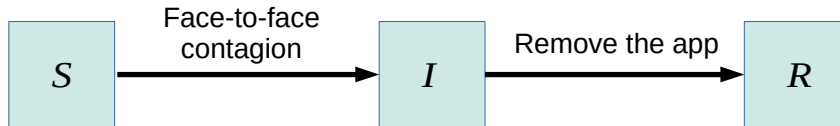
Figure 1: Flow chart of the SIR model applied in our study.

Regarding our model, we will assume that other mechanisms for app adoption different from face-to-face relation as described in [8], i.e., exogenous factors due to app popularity and spontaneous app installation after browsing an app market by the user, are weaker and, hence, less significant than face-to-face relations. Comparing our results with real data, we will be able either to validate or reject this assumption.

2.2 Parameters

To build our theoretical random network, we consider a population of 1,000,000 users. For the number of edges (or users' friends), we will be based upon the results obtained in [12], regarding the face-to-face friends relations in Spanish population between 15 and 20 years old. According to this latter paper, we will consider a mean network degree of $k = 13.25$ friends with standard deviation of 8.27. We focus on this 15–20 aged people because this group is considered as Mobile Addicts, i.e., a consumer that launches apps more than 60 times per day, and then, they are the most susceptible to app infection by others than any other age group [13]. The number of friends for every user will be assigned randomly, generating for every user a random number of friends from the normal (Gaussian) distribution with mean $\mu = 13.25$ and standard deviation $\sigma = 8.27$, $N(\mu = 13.25; \sigma = 8.27)$.

To do this, we sample k from $N(\mu = 13.25; \sigma = 8.27)$. Then, the number of edges in the network is $e = [k \cdot 1,000,000)/2]$, where $[\,]$ denotes the integer part function. In order to assign the e edges, we select two nodes randomly. If there is not a previous edge between them, then we assign the edge to these two nodes. Otherwise, we select another couple of nodes. We repeat this process until the e edges have been assigned. Thus, our random network is an Erdös-Rényi random network, where all edges are independent [14].

To simulate the network evolution we will need to set the simulation time, the infection rate and the user retention rate parameters.

For the simulation time, we consider $t_s = 100$ days timeline. This decision is made because, as shown in [15], the Android OS app half-life is 3 months. In [15], the half-life of an app is defined as the time instant at which the number of users has declined 50% with respect to its maximum value throughout its lifetime. After this point, the *virality* or infectiousness of the app is weaker.

Retention rates of apps by users at 30, 60 and 90 days, are determined

in reference [16]. We can express these rates as the probability that a user retains the app more than 30, 60 or 90 days, i.e., $P[X \geq 30]$, $P[X \geq 60]$ and $P[X \geq 90]$, respectively. Assuming that the retention time X has an exponential distribution of parameter $\lambda > 0$, and, since $P[X \leq x] = 1 - P[X \geq x]$, then we can calculate $P[X \leq 30]$, $P[X \leq 60]$ and $P[X \leq 90]$ as follows

$$f(x) = P[X \leq x] = 1 - e^{-\lambda x}, \qquad \lambda > 0, \tag{1}$$

being x the time the user has the app downloaded in his/her device, and $\lambda > 0$ the parameter needed to estimate the user retention days. Taking into account the values of x for different types of apps given in [16], we obtain the λ values satisfying the function $f(x)$ in Eq. (1) for 30, 60 and 90. Hence, we obtain an interval for λ values that will be between 0.008273 (for apps with high user retention rate) and 0.03539 (for apps with low user retention rate).

We assume that the infection rate parameter, β, will be a function of k/t_s,

$$\beta = \delta \frac{k}{t_s}, \tag{2}$$

where, as it has been previously defined, k and t_s are the network mean degree and the simulation time, respectively, and $\delta > 0$ is a tuning parameter. We will consider values for δ in the interval $[0, 0.65]$ in order to cover as many scenarios as possible.

2.3 Data

In order to compare our model simulations with real data, we have monitored several apps. They have been randomly chosen from [11]. These apps and their number of accumulated downloads in some dates are collected in Tables 1 and 2.

2.4 Simulations and selections

For modeling simulations, we use 1,000,000 of nodes and the variable parameters will be:

- The number of initial infected nodes: A random integer number generated uniformly in the interval $[1, 50]$. From an epidemiological point of view, a natural candidate for the number of initial infected nodes would be very small $(1-5)$, however considering a real context to our problem this number can be greater because the companies can use promotion campaigns where the app is offered for free use among some selected customers. Here, we will assume this number lying in the interval $[1, 50]$.

- The user retention rate: A random number, $\lambda > 0$, uniformly generated in the interval $[0.008273, 0.03539]$ that appears in Eq. 1).

- The infection rate: A random number generated uniformly in the interval $[0, 0.65]$ being δ the parameter that appears in Eq. 2.

In order to compute reliable estimations based on 95% confidence intervals (CI 95%), the technique referred to as Latin Hypercube Sampling (LHS) will be used, [17]. This technique will be applied to select sets of the variable parameters to be substituted into the model. LHS (a type of stratified Monte Carlo sampling) is an efficient method for achieving equitable samples of all input parameters simultaneously. In our problem, by LHS we obtain an equitable sample of 100,000 input parameters simultaneously. We substitute each set of the 100,000 parameters into the model and then we run a simulation. The set of results from the obtained simulations represent all the possible behavior of an app according to the considered parameters. After performing these simulations, a set of scenarios will be generated.

Based on the number of downloads over the time for a real app, then we should be able to select the behavior from our set of scenarios that best fit the behavior of the real app. In this manner, the evolution curve of that app will be estimated. This curve will be built taking into account two issues: On the one hand, in a real scenario, we want to be able to know the expected behavior of an app based just on the early days of its launch. This means that in practice the number of downloads will be available only at some early dates. On the other hand, a set of 100,000 results from our simulations are available. Based on the two previous facts, we will select the simulations that best fit the real data. For that, we introduce the following notation:

- $d(i)$ denotes the total number of accumulated downloads at the i-th day. In practice, the values of $d(i)$ are only known for some specific days, say, $d(i_1), d(i_2), \ldots, d(i_p)$, $1 \leq i_1 < i_2 < \cdots < i_p \leq t_s = 100$.

- $s(i,j)$ denotes the total number of accumulated downloads at the i-th day ($1 \leq i \leq 100$) for simulation j, $1 \leq j \leq 10^5$. To compare simulations with the available real data, $d(i_1), d(i_2), \ldots, d(i_p)$, just simulations $s(i_1, j), s(i_2, j), \ldots, s(i_p, j)$, $1 \leq j \leq 10^5$, will be required.

Taking into account that our network is comprised by 1,000,000 users and that the number of users in the real network is unknown, for each simulation j, a factor, $\alpha_j > 0$, will also be determined to scale the available real data $d(i_k)$, in such a way that the scaled real data $\alpha_j d(i_k)$ and the simulation $s(i_k, j)$ be close for all the days, $1 \leq k \leq p$. This approximation will be built using the Mean Square Error (MSE) as error measure. Thus, we calculate ϵ_j, the MSE of simulation $s(i,j)$, as

$$\epsilon_j = \sum_{k=1}^{p} \left(\alpha_j d(i_k) - s(i_k, j)\right)^2, \quad \alpha_j = \frac{\sum_{k=1}^{p} d(i_k) s(i_k, j)}{\sum_{k=1}^{p} (d(i_k))^2}, \quad 1 \leq j \leq 10^5. \quad (3)$$

This defines a set of mean square errors $\{\epsilon_j > 0 : 1 \leq j \leq 10^5\}$ associated to each simulation j. Notice that the best simulation $s(i, j^*)$, in the mean square sense, is given by the one where $\epsilon_{j^*} = \min\{\epsilon_j > 0 : 1 \leq j \leq 10^5\}$.

Now, we sort the simulations $s(i,j)$ by MSE in ascendant order. Thus, we search for the subset of simulations with lowest MSE such that, once it is calculated the 95% CI in each point (day), the available real data $d(i_1), d(i_2), \ldots, d(i_p)$ lie inside their corresponding 95% CI. Then, with this obtained subset of simulations we expect to estimate the behavior of the app downloads in the near future.

3 Results and discussion

This section shows the results obtained according to the method described in Section 2.4. We have monitorized fifteen apps from the real app market [11]. Hereinafter, we will show the results of our technique for two apps.

In Table 1, the total number of accumulated downloads during different days $d(i_k)$, for the first app (App1), is shown.

i_k-th day ($1 \leq k \leq 8 = p$)	4	11	14	21	28	34	84	95
# of accumulated downloads ($d(i_k)$)	281	707	873	1123	1284	1392	1886	1992

Table 1: App1. Total number of accumulated downloads in eight different days.

Considering a real scenario, we would only have the total number of downloads until the present, say 11-th day, given by $d(11)$. Therefore, the only available data is the total number of accumulated downloads corresponding to days, $d(4)$ and $d(11)$. By selecting the behaviors from our set that best fit the these two values and the 95% confidence interval, as explained in Section 2.4, we would obtain the results shown in Figure 2.

With our simulations, we are able to capture the real behavior of the App1 (red points in Figure 2) until the 28-th day after the app launch, since data $d(14)$, $d(21)$ and $d(28)$ is inside the 95% confidence interval generated by the proposed method. Notice that, although the value corresponding to 34-th day, $d(34)$, lies outside the confidence interval, it is not far from the 95% CI.

Following with the real scenario, if we reach the 14-th day, and we would dispose of the number of accumulated downloads $d(14)$, then we can feedback the proposed method with this new data. Then, using $d(4)$, $d(11)$ and $d(14)$, we would obtain the results shown in Figure 3. Now the real value $d(34)$ lies inside the new generated confidence interval. Our prediction also gives more downloads than the real future values.

As showed in Figure 2 and Figure 3, the behavior far from the app launch, for example in 84-th and 95-th days, whose number of total downloads are given by $d(84)$ and $d(95)$, respectively, are not captured. However, if we again feedback the selection, from $d(11)$, $d(14)$ and $d(21)$, the predicted behavior by the mean fits the data in that days, as shown in Figure 4, but generating wide confidence intervals.
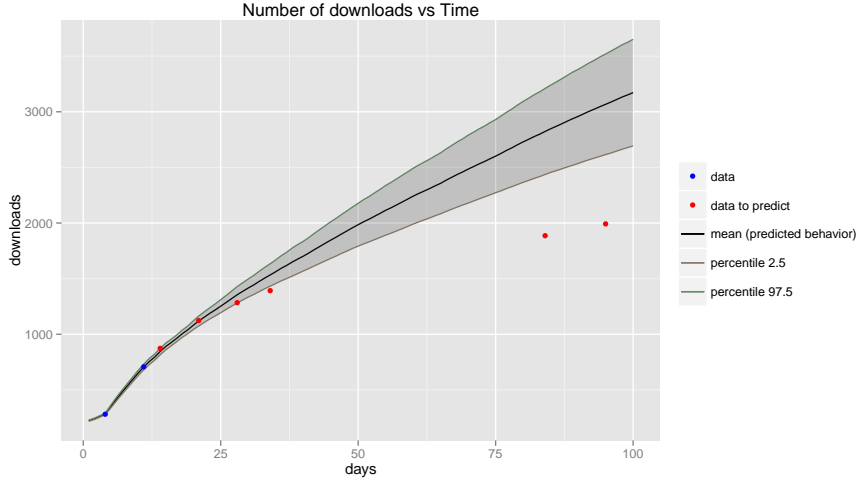
7

Figure 2: App1. Predicted behavior and 95% confidence interval of the number of accumulated downloads vs. time, based on data $d(4)$ and $d(11)$ corresponding to the 4-th and 11-th days, respectively.

Now, we consider an app with a low level of downloads, whose figures are listed in Table 2.

i_k-th day ($1 \leq k \leq 6 = p$)	6	8	13	20	28	70
# of accumulated downloads ($d(i_k)$)	98	131	170	186	200	281

Table 2: App4. Total number of accumulated downloads during six different days.

In this case, using real data $d(6)$ and $d(8)$, the proposed model is able to predict the total number of accumulated of downloads in the 13-th day. This has been plotted in Figure 5. If the model is feedback using data $d(13)$ corresponding to the 13-th day, then the model also captures the real data $d(20)$ and $d(70)$, being small the error corresponding the prediction at the 28-th day. The results can be seen in Figure 6.

Summarizing, in this section we have shown the results provided by our method with two different types of apps. It has been shown that the proposed method improves the prediction when it is feedback. To show the robustness of the the proposed method, it has been tested with thirteen additional apps from [11]. The results with these thirteen additional apps can be found in [18]. In all the cases, good results have been obtained. On the one hand, we have shown the proposed model is able to predict the behaviour of apps with an average (standard) or low total number of downloads. On the other hand, the method
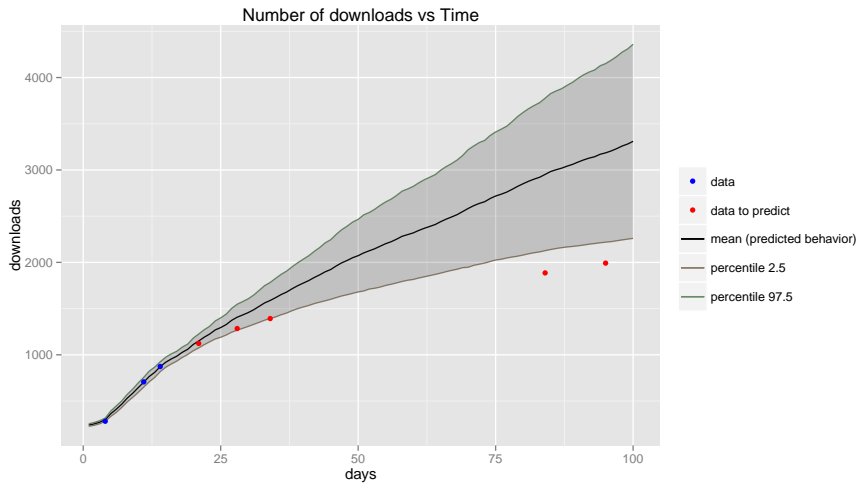
8

Figure 3: App1. Predicted behavior and 95% confidence interval of the number of accumulated downloads vs. time, based on data $d(4)$, $d(11)$ and $d(14)$ corresponding to the 4-th, 11-th and 14-th days, respectively.

does not provide correct results for the behavior of apps with a high number of downloads, i.e. high *virality*, at the first stages due to their fast growth. However, the predictions improve when the model is feedback providing correct results.

The proposed method shows that it is possible to provide an approximation for the behavior of the number of downloads using confidence intervals. The key for the prediction accuracy is to select the adequate parameters for the model building. Depending on the type of app that we want to anticipate its behavior, we should fix the set of parameters as retention time and infection rate according to its characteristics. For example, an app with a marketing campaign should increase its infection parameter according to the expected impacts of that campaign.

4   Conclusion

In this work, an epidemiological random network model to estimate the evolution of download of apps has been proposed. The model's goals have been to predict the total number of accumulated apps' download as well as to show the network effect on the apps' downloads validating our assumption that exogenous factors due to app popularity and spontaneous app installation after browse an app market by the user are weaker and less significant than face-to-face relations. In addition, the proposed model generalizes the results obtained in other works [6, 7, 8] with very high networks.
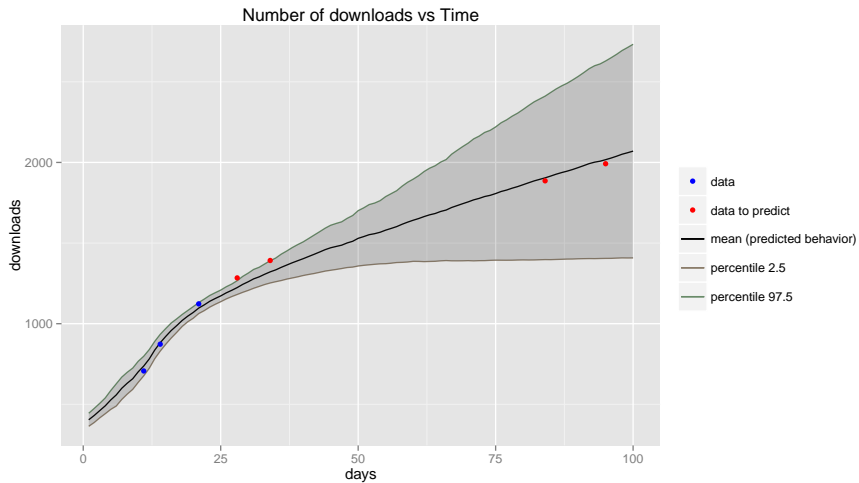
Figure 4: App1. Predicted behavior and 95% confidence interval of the number of accumulated downloads vs. time, based on data from $d(11)$, $d(14)$ and $d(21)$ corresponding to the 11-th, 14-th and 21-th days, respectively.

The results show that the prediction of the evolution of the number of downloads of an app over the time is possible via computational methods whenever proper parameters are adequately chosen. The capability of the model to capture the behavior of the app by means of confidence intervals has been shown. Therefore the face to face relations are more important than other mechanisms for apps' adoption. Although, the proposed method does not consider exogenous factors, it is capable to forecast correctly, using confidence intervals, the evolution of the number of downloads for monitored apps.

The study has been based on 100,000 simulations. This permits to generalize the results obtained in other contributions about the face-to-face network effect in apps adoption, that were based in only one realization of the experiment.

Recently, in [19], the authors have just focused on Facebook apps using, among other approaches, a SIRS epidemiological model. This approach may not be appropriate for mobile Apps, since when a person uninstall a mobile App it is unlikely he/she installs it again [16]. In addition, in our contribution we consider that the contagion (or user adoption) may change over the time due to the feedback technique used. This feature is not considered in [19].

Marketing researchers and strategy business managers can benefit from the proposed model since it can be helpful to predict app behavior over the time, anticipating the spread of an app as well as predicting its expected value.
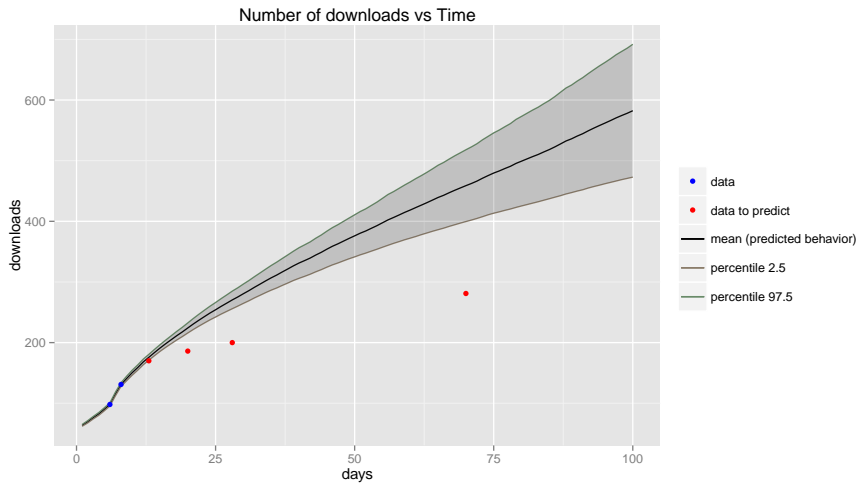
Figure 5: App4. Predicted behavior and 95% confidence interval of number of accumulated downloads vs. time, based on data from $d(6)$ and $d(8)$ corresponding to the 6-th and 8-th days, respectively.

Short biographies

- Juan Alegre-Sanahuja is Ph.D. in Applied Mathematics and works as freelance in Engineering projects in Valencia, Spain.

- Juan-Carlos Cortés is Ph. D. in Applied Mathematics and full professor at the Universitat Politècnica de Valéncia, Valencia, Spain. He is also Subdirector of the Universitary Institute of Multidisciplinary Mathematics.

- Rafael-J. Villanueva is Ph. D. in Applied Mathematics and full professor at the Universitat Politècnica de Valéncia, Valencia, Spain, in the Universitary Institute of Multidisciplinary Mathematics.

- Francisco-J. Santonja is Ph. D. in Applied Mathematics and professor in the Universitat de Valéncia, Valencia, Spain, Department of Statistics and Operations Research.

References

[1] Smartphone Vendor Market Share; [cited May 7th, 2017]. Available from: http://www.idc.com/prodserv/smartphone-market-share.jsp.

[2] Gartner Says Sales of Smartphones Grew 20 Percent in Third Quarter of 2014; [cited May 7th, 2017]. Available from: https://www.gartner.com/newsroom/id/2944819.
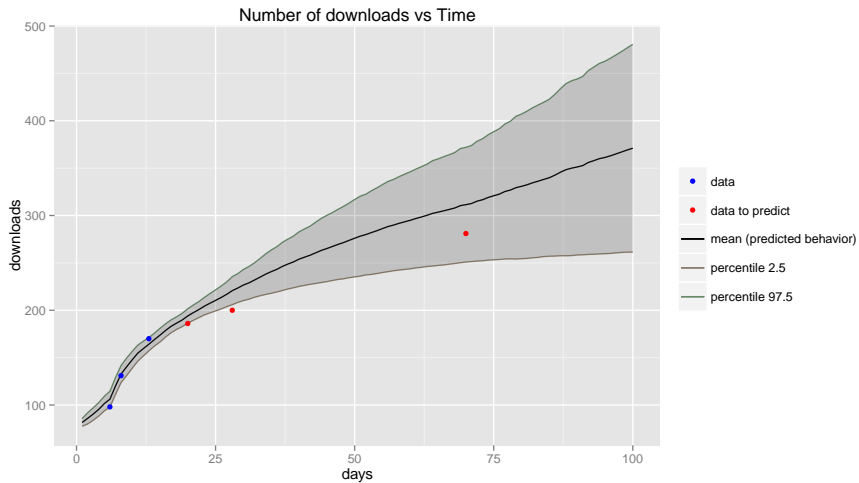
Figure 6: App4. Predicted behavior and 95% confidence interval of number of accumulated downloads vs. time, based on data from $d(6)$, $d(8)$ and $d(13)$ corresponding to the 6-th, 8-th and 13-th days, respectively.

[3] List of mobile software distribution platforms; [cited May 7th, 2017]. Available from: http://en.wikipedia.org/wiki/List_of_mobile_software_distribution_platforms.

[4] Yang J, Leskovec J. Modeling Information Diffusion in Implicit Networks. In: 2010 IEEE International Conference on Data Mining. IEEE; 2010. Available from: https://doi.org/10.1109%2Ficdm.2010.22.

[5] Luarn P, Yang JC, Chiu YP. The network effect on information dissemination on social network sites. Computers in Human Behavior. 2014 aug;37:1–8. Available from: https://doi.org/10.1016%2Fj.chb.2014.04.019.

[6] Taylor DG, Voelker TA, Pentina I. Mobile Application Adoption By Young Adults: A Social Network Perspective. International Journal Of Mobile Marketing. 2011;6(2):60–70. Available from: http://digitalcommons.sacredheart.edu/wcob_fac/1/.

[7] Aharony N, Pan W, Ip C, Pentland A. Tracing mobile phone app installations in the "friends and family" study. In: Proceedings of the 2010 Workshop on Information in Networks (WIN'10); 2010. Available from: http://web.media.mit.edu/~panwei/pub/funf_highlevel_n_apps_win_final.pdf.

[8] Pan W, Aharony N, Pentland A. Composite Social Network for Predicting Mobile Apps Installation. In: AAAI Conference on Artificial Intel-

ligence; 2011. Available from: https://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3729.

[9] Santonja FJ, Morales A, Villanueva RJ, Cortés JC. Analysing the effect of public health campaigns on reducing excess weight: A modelling approach for the Spanish Autonomous Region of the Community of Valencia. Evaluation and Program Planning. 2012 feb;35(1):34–39. Available from: https://doi.org/10.1016%2Fj.evalprogplan.2011.06.004.

[10] Cortés JC, Sánchez F, Santonja FJ, Villanueva RJ. A Probabilistic Analysis to Quantify the Effect of March 11, 2004, Attacks in Madrid on the March 14 Elections in Spain: A Dynamic Modelling Approach. Abstract and Applied Analysis. 2015;2015:1–8. Available from: https://doi.org/10.1155%2F2015%2F387839.

[11] Slideme. Your Marketplace for android apps; [cited May 7th, 2017]. Available from: http://www.slideme.org.

[12] Navarro-Pertusa E, Reig-Ferrer A, Barber-Heredia E, Ferrer-Cascales RI. Grupo de iguales e iniciación sexual adolescente: Diferencias de género (Groups of pairs and adolescent sexual initiation: Gender differences). International Journal of Clinical and Health Psychology. 2006;6(1):79–96. Available from: https://dps.ua.es/es/documentos/pdf/1999/grupo-de-iguales.pdf.

[13] The Rise of the Mobile Addict. Flurry insights; [cited May 7th, 2017]. Available from: http://flurrymobile.tumblr.com/post/115191945655/the-rise-of-the-mobile-addict.

[14] Bollobas B. Random Graphs. 2nd ed. Cambridge University Press; 2001.

[15] Benchmarking the Half-Life and Decay of Mobile Apps. Flurry insights; [cited May 7th, 2017]. Available from: http://flurrymobile.tumblr.com/post/115191376315/benchmarking-the-half-life-and-decay-of-mobile.

[16] App Engagement: The Matrix Reloaded. Flurry insights; [cited May 7th, 2017]. Available from: http://flurrymobile.tumblr.com/post/113379517625/app-engagement-the-matrix-reloaded.

[17] Hoare A, Regan DG, Wilson DP. Sampling and sensitivity analyses tools (SaSAT) for computational modelling. Theoretical Biology and Medical Modelling. 2008;5(1):4. Available from: https://doi.org/10.1186%2F1742-4682-5-4.

[18] Predicting mobile apps spread. Experiments and additional material; [cited May 7th, 2017]. Available from: http://app.imm.upv.es.

[19] Kloumann I, Adamic L, Kleinberg J, Wu S. The Lifecycles of Apps in a Social Ecosystem. In: Proceedings of the 24th International Conference on World Wide Web - WWW'15. ACM Press; 2015. Available from: https://doi.org/10.1145%2F2736277.2741684.