Predicting Micro-Level Behavior in Online Communities for Risk Management

Philippa A. Hiscock[1], Athanassios N. Avramidis[2] and Jörg Fliege[3]

[1] University of Southampton, SO17 1BJ, UK P.A.Hiscock@soton.ac.uk
[2] University of Southampton, SO17 1BJ, UK aa1w07@soton.ac.uk
[3] University of Southampton, SO17 1BJ, UK J.Fliege@soton.ac.uk

Abstract. Online communities amass vast quantities of valuable knowledge and thus generate major value to their owners. Where these communities are incorporated in a business as the main means of sharing ideas and issues regarding products produced by the business, it is important that the value of this knowledge endures and is easily recognized. For good management of such a business, risk analysis of the integrated online community is required.

We choose to focus on the process of knowledge creation rather than the knowledge gained from individual messages isolated from context. Consequently, we model collections of messages, linked via tree-like structures; these message collections we call threads. Here we suggest a risk framework aimed at managing micro-level thread related risks. Specifically, we target the risk that there is no satisfactory response to the original message after a period of time. Risks are considered as binary events; the event can therefore be flagged when it is predicted to occur for the attention of the community manager. To predict such a binary response, we use several methods, including a Bayesian probit regression estimated via Gibbs sampling; results indicate this model to be suitable for classification tasks such as those considered.

1 Introduction

Online communities have evolved at an ever-increasing rate in the recent past and continue to grow steadily. Their use is not limited to domestic domains, being widespread in various business, scientific and public service domains. Likewise, substantial economic value is no longer only generated by high profile public communities, e.g. Twitter and Facebook, but also by business communities, such as the SAP Community Network (SCN) (http://scn.sap.com/) and IBM's Connections (http://www-03.ibm.com/software/products/us/en/conn). Online communities are now pivotal elements in corporate management and marketing, product support, customer relations management, product innovation and targeted advertising. Members of such communities are connected in a way that opinion, knowledge and ideas may be shared to facilitate collaboration.

Each online community is a valuable ecosystem that is full of information, the micro and macro dynamics (i.e. structure, behavior and economics) of which are unclear. It is obvious that risks and overlooked emerging opportunities present threats to the health of such an ecosystem. Techniques that enable the health in online communities to be measured, managed, analyzed, protected and optimized are therefore invaluable. This paper outlines tools utilized and developed to enable timely analysis and decision support of micro-level risks and/or opportunities in the SCN.

In the following, we consider a classification prediction task based around a thread-level opportunity relevant to managers of online communities (Section 2). Anderson et al. (2012) study binary classification on thread-based events; the set of features they used inspired and informed our choice (Section 2.2). However, we consider a different and broader set of methods described shortly in Section 3: Bayesian probit; generalized linear model with probit link and with logit link; linear discriminant analysis. Finally, we discuss results obtained (Section 4) and draw corresponding conclusions (Section 5).

2 The Online Community Considered and Problem Definition

2.1 SCN: The SAP Community Network

SAPs community network (SCN) is a business community platform where any uniquely registered person, referred to as *user*, may discuss and share their ideas and issues regarding SAP products. This community mainly consists of a number of fora, each relating to a unique product or topic. A user may post a message in any forum and a collection of messages form a *thread*. The first message in a thread is the parent message (i.e. 'question') and subsequent messages are linked via a tree-like structure. As messages are linked, they are given a *time rank* and *wall clock*, that is an arrival order and minutes since thread creation. The user who posts the parent message, is known as the *original poster* (OP). A user who makes a post in response to the parent message is called a *respondent*. Within the tree-like message structure of a thread, the *most responded to message* (MRTM) is that with greatest number of messages posted in direct response. Similarly, the *most responded to user* (MRTU) (including the OP) is the user to whom the greatest number of direct responses is made of all users to post in the thread.

The OP is the only user capable of making certain actions with respect to their thread. Each respondent may be awarded *points* by the OP based on the quality of their response, see Table 1. The SCN places light restrictions on the way an OP awards points in a thread such that only one 10 and two 6 point scores may be awarded. Consequently we define a thread to be *solved* only if the OP has awarded a 10 point score to a response; the associated respondent is known as the *thread solver* (TS). A more relaxed version of the

TS is the *highest point scorer* (HPS). Where the HPS is not the TS, there may be more than one HPS. In the SCN, points awarded are connected to the corresponding respondents message allowing respondents to increase their *reputation*. A user's reputation is the total points accumulated over time. We view respondent reputation to be forum-specific due to forum topic inhomogeneity. Assuming a thread has at least one respondent, the *most reputable respondent* (MRR) is that with greatest lifetime reputation.

The OP in addition can change the *status* of a thread from the default 'Unanswered' to 'Answered'. However, there are no restrictions on when an OP may change the status of a thread. For example, a thread does not have to be solved to have status 'Answered' - the converse also holds true.

Table 1. SCN's point awarding system via the original poster.

Original poster's view of respondent's post	Points awarded
Respondent 'solved' the issue of the parent post	10
Respondent was 'very helpful' towards the issue of the parent post	6
Respondent was 'helpful' towards the issue of the parent post	2

SAP made available a complete trace of actions of 95 fora (a third of the total byte size of the SCN) from February 2004 to July 2011. We select three fora showing variance in micro-level activity during the period analyzed: forum 50 spiking; forum 142 staying mostly level; and forum 246 decreasing.

2.2 Problem Definition

Problem motivation arose after observing only 23.26% of threads created within the dataset to be solved. Of the unsolved threads, approximately 12% are never responded to. We consider a classification event which may be viewed either as a risk or an opportunity. That is, after a time threshold, t_s minutes, of creation, the thread is solved (opportunity) or unsolved (risk).

Assuming the i^{th} thread to be eventually solved, we note the wall clock time (minutes since thread creation) of this event as w_i. The default value of w_i is ∞. Thus the binary response observed, y_i, for the i^{th} thread, is

$$y_i = \begin{cases} 1 & \text{if } 0 \leq w_i \leq t_s, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $i = 1, \ldots, n_o$ and n_o is the number of threads observed within the sample population.

Considering only those threads created at least one year prior to our last observation and having at least one respondent, 13.76% are not responded to within the first 24 hours and 0.56% are only responded to a year after thread creation. This highlights that the vast majority of threads receive greatest attention within the first 24 hours after creation. Of the threads responded to

within the first 24 hours, 28.56% are subsequently solved and of these 74.57% are solved within the first 24 hours. Within the threads solved in the first 24 hours, 62.64% are solved by the first response. In comparison, of the threads responded to only after the initial 24 hour period, 15.18% are eventually solved. However, within these latter solved threads, 64.77% are solved by the first response. Thus, although a thread seems less likely to be solved if not responded to within the first 24 hours of creation, it is still most likely to be solved by the first respondent.

Figure 1 illustrates the percentage of threads solved across all fora, grouped by year, over hours since thread creation. In all cases, the curve incline begins to reduce six hours after thread creation and starts to level off 24 hours after thread creation. We therefore take t_s in (1) to be 1440 minutes (24 hours).

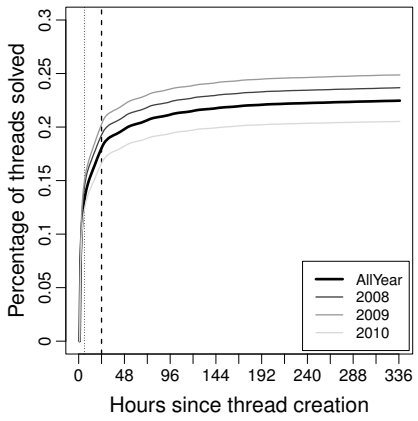

Fig. 1. Percentage of threads solved by hours since thread creation for all threads within dataset created between 2008 and 2010.

Features available for prediction

The full set of features available for prediction following t minutes since thread creation is given below. The choice of t affects feature inclusion. For t sufficiently close to t_s, one could argue prediction is made too closely to the occurrence of the event. In addition, for t close to zero (thread creation) there exist uninformative features where all observations hold the same value. We trialled $t \in \{30, 60, 180, 360\}$ minutes; here we report on $t = 30$ minutes due to space constraints and lack of improvement for larger t. Those features marked by an asterisk are included in our feature space for modeling y_i in (1) given t.

- **OP features:** OP reputation*; OP reputation in past year*; # thread OP participated*; # thread OP created*; # thread OP created subsequently solved; # thread OP solved*; # messages OP posted*; # messages OP

posted in thread*; # days since OP registration (first appearance in relevant forum)*.

- **TS features:** TS reputation; TS reputation in past year; # thread TS participated; # thread TS created # messages TS posted in thread.
- **MRR features:** MRR reputation*; MRR reputation in past year*; # messages MRR posted in thread*.
- **HPS features:** # HPSs*; mean HPS reputation*; # responses to HPS*.
- **MRTM features:** # MRTM*; # responses to MRTM*; mean MRTM reputation*; mean MRTM points earned*.
- **MRTU features:** # MRTU*; # responses to MRTU*; mean MRTU reputation*; mean MRTU points earned*; mean # thread MRTU solved; mean # thread MRTU created*; mean # messages MRTU posted*.
- **Temporal features:** minutes till first reply*; mean minutes till respondents first message*; mean minutes between messages*; median minutes between messages*; minimum minutes between messages*; TS time rank; TS wall clock; MRR time rank*; MRR wall clock*; mean HPS time rank*; mean HPS wall clock*; minimum MRTM time rank*; minimum MRTM wall clock*; minimum MRTU time rank*; minimum MRTU wall clock*.
- **Thread summary features:** indicator for TS is MRR; indicator of thread status*; indicator of thread solved; # users to participate*; sum points awarded*; # messages posted*; mean respondent reputation*; median respondent reputation*; mean respondent reputation in past year*.

3 Classification Methods Applied

We apply four linear methods for classification against the baseline model of randomized prediction (RAND) informed by observed class proportion in the training set. These models are: Bayesian probit (BP) model; generalized linear model with probit link (GLMP); generalized linear model with logit link (GLML); and linear discriminant analysis (LDA). Here, only the first model is non-standard, being taken from (Albert and Chib 1993), as such, some detail is given below. Details of GLM and LDA methods fitted are described in Venables and Ripley (2002) Chapters 7 and 12 respectively (function names glm() and lda()). For a comprehensive guide to generalized linear models see McCullagh and Nelder (1989). A more general introduction to linear models for classification, including LDA, is provided by Hastie et al. (2011). In Section 4 we compare quality characteristics of classification predictions made with respect to the problem defined in Section 2.2.

First, we introduce some general notation. Let X be the normalized column matrix with rows $x_i^T = [x_{i,1}, \ldots, x_{i,p}]$ where $x_{i,j}$ is the i^{th} observation of the j^{th} feature; and $i = 1, \ldots, n_o$, for n_o the number of observations (here the number of threads) within the sample population. To avoid identifiability or non-integrability issues later on, we assume that $X^T X$ is non-singular. Given that X has full column-rank, this assumption is always satisfied. In

addition let β be the corresponding p length vector of (elasticity) coefficients. For all four methods considered, the i^{th} binary response Y_i is modeled via the corresponding latent variable Z_i; in the methods taking the probit link function

$$Z_i \overset{\text{i.d.}}{\sim} N\left(x_i^T\beta, \sigma\right), \tag{2}$$

$$Y_i = \begin{cases} 1 & \text{if } Z_i > 0, \\ 0 & \text{if } Z_i \leq 0. \end{cases}$$

Where "$\overset{\text{i.d.}}{\sim}$" means "independently distributed" and $N(\mu, \sigma)$ denotes a Normal distribution with mean μ and variance σ. Note that we set $\sigma = 1$ in (2) such that the distribution on the error terms is the standard normal.

Given a sample $y = [y_1, \ldots, y_{n_o}]^T$ and an associated column matrix X, a statistical inference problem about the coefficients β arises. Frequentist treatment of the above model, assuming probit link, leads to the generalized linear model with probit link. Here, one maximizes the β-likelihood; integrating out the latent variables analytically to give $L(\beta) = \prod_{i:y_i=1} \mathbb{P}(Z_i > 0) \prod_{i:y_i=0} \mathbb{P}(Z_i \leq 0) = \prod_{i:y_i=1} \Phi(x_i^T\beta) \prod_{i:y_i=0} (1 - \Phi(x_i^T\beta))$, where Φ is the standard normal (cumulative) distribution function. This can be similarly shown for the generalized model with logit link.

The Bayesian probit model takes a Bayesian approach to inference, following Albert and Chib (1993). We use the notation (β, Z) to denote the $(p + n_o)$-dimensional random vector consisting of the elements of β and of $Z = [Z_1, Z_2, \ldots, Z_{n_o}]$ and the symbol "$\propto$" as "is proportional to". Let β have prior probability density function, $\pi_0()$, then the posterior of (β, Z) is

$$\pi(\beta, Z) \propto \pi_0(\beta) \prod_{i=1}^{n_o} \left[1(y_i = 1)1(Z_i > 0) + 1(y_i = 0)1(Z_i \leq 0)\right] \times \phi\left(Z_i; x_i^T\beta\right) \tag{3}$$

assuming this is integrable. In (3), $\phi\left(Z_i; x_i^T\beta\right) \propto \exp\left(-\left(Z_i - x_i^T\beta\right)^2/2\right)$ is the normal (Gaussian) density with mean $x_i^T\beta$ and variance 1; and 1() is the indicator function. More concretely, letting $x = (\beta, Z)$ and taking μ as the $(p + n_o)$-dimensional Lebesgue measure, the function in (3) is (a version of) the density of a probability measure on $\mathbb{R}^{(p+n_o)}$ with respect to μ only where $C \overset{\text{def}}{=} \int_{\mathbb{R}^{p+n_o}} \pi(x)\mu(\mathrm{d}x)$ is finite.

We take a flat prior for π_0, meaning that all points in $\mathbb{R}^p$ are, essentially, "equally likely", as is later mentioned, other choices are available. As the support is unbounded and the intended "density" is a positive constant, this π_0 does not give a probability measure on $\mathbb{R}^p$ and is hence *improper*. This is not a problem where (3) defines a probability measure. Thus the target of inference is the resulting β-marginal of (3). This target is denoted π_β.

The Bayesian probit method for binary response data as prescribed by Albert and Chib (1993) utilizes Gibbs sampling. Gibbs sampling is a particular method of Markov Chain Monte Carlo class and works by sampling from

conditional distributions of the target probability measure, the (3) here. See Casella and George (1992) for an introduction; for a thorough treatment, see Chapters 9 and 10 of Robert and Casella (2004). Whilst the conditional distributions of the target (3) are easy to sample from (Albert and Chib (1993)), the level of ease depends on the choice of π_0. For conditionals with uniform π_0 see Albert and Chib (1993). In addition, Albert and Chib (1993) give integrable, that is *proper*, possibilities for the prior π_0 which directly enable Gibbs sampling. Given that we assume a uniform prior distribution for the regression coefficients, related issues of the propriety of the posterior distribution is studied by Chen and Shao (1999). In practice, the initial state of the Markov chain for β is taken to be the maximum likelihood (ML) estimate, $\tilde{\beta}_{ML} = \left(X^T X\right)^{-1} X^T y$.

Given the predictors x_i^T, the posterior mean of Y_i is $\mathbb{P}_{\pi_\beta}(Y_i = 1) = \mathbb{P}_{\pi_\beta}(Z_i > 0) = \mathbb{E}_{\pi_\beta}[\Phi(x_i^T \beta)]$, where $\mathbb{P}_{\pi_\beta}$ and $\mathbb{E}_{\pi_\beta}$ denote the probability and expected value with respect to π_β. Assuming certain conditions, a consistent estimator of this mean is the corresponding sample average of the Gibbs sample; whereby consistency we mean convergence with probability one as the sample size tends to infinity (Robert and Casella (2004), Theorem 6.63; Cappé et al. (2005), Theorem 14.2.53). Hence, given the sample $\{\beta^{(1)}, \beta^{(2)}, \ldots, \beta^{(M)}\}$, with M sufficiently large, $M^{-1} \sum_{m=1}^{M} \Phi(x_i^T \beta^{(m)})$ is an appropriate estimator of $\mathbb{P}_{\pi_\beta}(Y_i = 1)$.

4 Results

We implement our methods in the language and environment R (version 3.0.1) (R Core Team (2013)) on a stand-alone computer with 64-bit operating system and 16 gigabytes of memory. With regard to the Bayesian probit model (Section 3), ad-hoc experimentation led us to believe a "burn-in" period of $t_b = 90,000$ and subsequent sample of $t_r = 10,000$ to result in estimates accurate for our purpose. In all instances, we implement 10-fold cross-validation.

To assess the quality of our classification predictions, we consider the receiver operating characteristic (ROC). This characterizes true positive rate (TPR) and false positive rate (FPR) as the discrimination threshold (d) is varied; where $\hat{y}_i = 1$ if and only if the posterior probability $\mathbb{P}(Y_i = 1) > d$ and

$$\text{TPR} = \frac{\sum_{i=1}^{n_o} 1(\hat{y}_i = 1, y_i = 1)}{\sum_{i=1}^{n_o} 1(y_i = 1)}, \ \text{FPR} = \frac{\sum_{i=1}^{n_o} 1(\hat{y}_i = 1, y_i = 0)}{\sum_{i=1}^{n_o} 1(y_i = 0)} \tag{4}$$

(Fawcett (2006)). In (4), 1() is the indicator function, $\hat{y}_i$ and y_i are the predicted and observed classifications for the i^{th} thread where $i = 1, \ldots, n_o$. The area under ROC curve (AUC) is used to summarize our observations of the ROC curves, calculated using the R package ROCR (Sing et al. 2005).

Predictions for the event of Section 2 are made both for the entire thread population and thread subpopulations, segregated by forum. As stated (Section 2.1), we discuss here only three fora of the SCN. We started with our full

feature set, described in Section 2.2, and performed classification with subsets of these, partitioned by feature type (indicated in bold feature within the feature list). The complete set of features is noted S_{All}; the subset of original poster features, S_1; the subset of MRR features, S_2; the subset of HPS features, S_3; the subset of both MRTM and MRTU features, S_4; and the subset of temporal and thread summary features, S_5.

Results for the application of the methods in Section 3 are reported in Figure 2. Observe that the original poster features (S_1) have good predictive power across all fora. As expected, the most reputable respondent features (S_2) are very useful — increasing the AUC by almost 20 points and doubling the TPR for BP, GLMP and GLML methods in every fora. The quality of the LDA method classifications is greatly improved by including features for the highest point scorer (S_3). Here, the AUC improves for all methods, although the TPR dips for those methods involving probit link. By including the features regarding both the most responded to message and user (S_4), very little appears to be gained. When all features are included (S_{All}), the quality of method classifications is high, both with regard to AUC and TPR measures. The GLML method classification quality sees a substantial increase in TPR, more than trebling with regard to all fora. However, the GLML method consistently has lowest TPR of all methods across all fora (excluding the random baseline method). On the other hand, considering only the AUC, the quality of the classifications is lowest for the LDA method.

Thus we see that incorporating the rich micro-level community dynamics surrounding an original post significantly aids in determining whether a satisfactory response will be made in good time — no matter the method. In addition, we stress that these features are extracted only 30 minutes after the original post was made and are predicting whether a satisfactory response will arrive during the subsequent 1410 minutes. We find it promising towards real-time application that after a mere 30 minutes there is sufficient information to predict, comparatively long-term, whether a thread will be solved. In addition, that the main value is from those features which are not direct evaluations of the original post.

5 Conclusion

Given the question-answer nature of the online community considered and the ever increasing complexity of community dynamics, it is valuable to think of each 'question' related set of messages as a series of connected information. We have demonstrated how the rich structure of the SAP community network can be used to identify important characteristics of linked messages such that original posts needing additional help via the manager of the community can be identified. In our ongoing work, we found Bayesian probit models to be promising tools for predicting such binary classification risk events. We see our approach to be promising for question-answer communities in general.

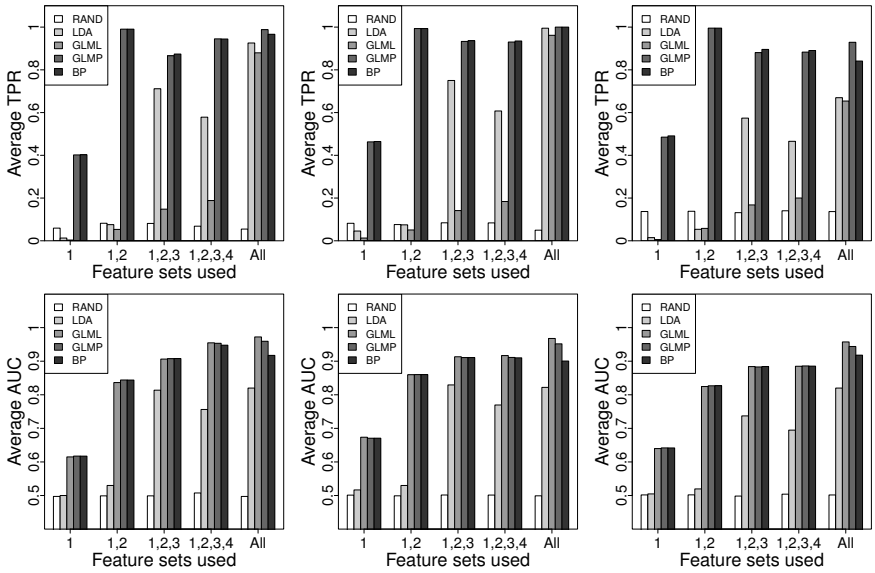

Fig. 2. Classification quality characteristics, true positive rate with discrimination threshold 0.5 (top) and area under ROC curve (bottom), averaged over cross-validation sets; given the event in Section 2.2 for fora 50, 142 and 246 (left to right) with 13236, 35933 and 34102 unique threads respectively.

Our Bayesian probit model incorporating a Gibbs sampler does require care when implementing. First, one must ensure that the multivariate Markov chain for β has converged to the desired target, regardless of the initial state of the chain. This typically involves verifying conditions of irreducibility, aperiodicity, positivity, and Harris recurrence; see Robert and Casella (2004), Chapter 6, for example. Second, selecting an appropriate burn-in and retained sample size tends to be challenging; see Robert and Casella (2004), Chapter 12.

Further investigation into quality characteristics for comparing binary classifiers is required. This is motivated by the discussion in the literature on the validity of AUC as a standalone measure of classification performance occurring primarily between Flach and Hand (Berrar and Flach (2012)), (Flach (2010)), (Hand 2009) and (Hand (2006)). Consequently, with increased automation, classifying streaming data would become more flexible.

Acknowledgments

We thank Dr. Adrian Mocan of SAP for his contribution to defining the risk events. We also thank Edwin Tye, School of Mathematics, University of Southampton, for his assistance in processing the data. This work has been supported by the EU FP7 project ROBUST, EC Project Number 257859.

References

ALBERT, J. H., and CHIB, S. (1993): Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association, 88(422), 669–679.*

ANDERSON, A., HUTTENLOCHER, D., KLEINBERG, J., and LESKOVEC, J. (2012): Discovering value from community activity on focused question answering sites: a case study of stack overflow. In: *KDD*. ACM, New York, NY, USA, 850–858.

BERRAR, D. and FLACH, P.A. (2012): Caveats and pitfalls of ROC analysis in clinical microarray research (and how to avoid them). *Briefings in bioinformatics, 13, 83–97.*

CAPPÉ, O., MOULINES, E., and RYDÉN, T. (2005): *Inference in hidden Markov models*. Springer Series in Statistics. New York: Springer.

CASELLA, G., and GEORGE, E.I. (1992): Explaining the Gibbs sampler. *The American Statistician, 46(3), 167–174.*

CHEN, M.H., and SHAO, Q.M. (1999): Properties of prior and posterior distributions for multivariate categorical response data models. *Journal of Multivariate Analysis, 71(2), 277–296.*

FAWCETT, T. (2006): An introduction to ROC analysis. *Pattern Recognition Letters, 27(8), 861–874.*

FLACH, P.A. (2010): ROC analysis. In: C. Sammut and G.I. Webb (Eds.): *Encyclopedia of Machine Learning*. Springer US, Boston, MA, 869–875.

HAND, D.J. (2006): Classifier technology and the illusion of progress (with discussion). *Statistical Science, 21(1), 1–34.*

HAND. D.J. (2009): Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine Learning, 77(1), 103–123.*

HASTIE, T., TIBSHIRANI, R., and FRIEDMAN, J. (2011): *The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Second Edition)*. Springer.

LIN, J. and KOLCZ, A. (2012): Large-Scale Machine Learning at Twitter. In: *SIGMOD 2012*. ACM Press, New York, NY,793–804.

MCCULLAGH, P., and NELDER, J.A. (1989): *Generalized Linear Models (Second Edition)*. Chapman and Hall/CRC.

ROBERT, C.P., and CASELLA, G. (2004): *Monte Carlo Statistical Methods (Second Edition)*. Springer-Verlag.

R CORE TEAM (2013): *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/

SING, T., SANDER, O., BEERENWINKEL, N. and LENGAUER, T. (2005): ROCR: visualizing classifier performance in R. *Bioinformatics, 21(20), 3940–3941.* URL: http://rocr.bioinf.mpi-sb.mpg.de

TAUSCZIK, Y.R. and PENNEBAKER, J.W. (2011): Predicting the Perceived Quality of Online Mathematics Contributions form Users' Reputations. In: *CHI 2011*. ACM Press, New York, NY, 1885–1888.

VENABLES, W., and RIPLEY, B. (2002): *Modern Applied Statistics with S (Fourth Edition)*. Springer.