Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 Keller Hall
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 13-033

A Hazard Based Approach to User Return Time Prediction

Komal Kapoor, Jaideep Srivastava, Mingxuan Sun, and Tao Ye

November 18, 2013

A Hazard Based Approach to User Return Time Prediction

Komal Kapoor
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455
kapoo031@umn.edu

Mingxuan Sun
Pandora Media Inc.
2101 Webster Street
Oakland, CA 94612
msun@pandora.com

Jaideep Srivastava
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455
srivasta@cs.umn.edu

Tao Ye
Pandora Media Inc.
2101 Webster Street
Oakland, CA 94612
tye@pandora.com

ABSTRACT

In the competitive environment of the internet, retaining and growing one's user base is of major concern to most web services. Furthermore, the economic model of many web services is allowing free access to most content, and generating revenue through advertising. This unique model requires securing user time on a site rather than the purchase of good. Hence, it is crucially important to create new kinds of metrics and solutions for growth and retention efforts for web services. In this work, we first propose a new retention metric for web services concentrating on the rate of user return. Secondly, we apply predictive analysis to the proposed retention metric on a service. Finally, we set up a simple yet effective framework to evaluate a multitude of factors that contribute to user return. Specifically, we define the problem of return time prediction for free web services. Our solution is based on the Cox's proportional hazard model from survival analysis. The hazard based approach offers several benefits including the ability to work with censored data, to model the dynamics in user return rates, and to easily incorporate different types of covariates in the model. We compare the performance of our hazard based model in predicting the user return time and in categorizing users into buckets based on their predicted return time, against several baseline regression and classification methods and find the hazard based approach to far surpass our baselines.

Categories and Subject Descriptors

J.4 [**Computer Applications**]: Social and Behavioral Sciences
; H.3.5 [**Online Information Services**]: Web-based services

General Terms

Human Factors, Algorithms, Experimentation

Keywords

online user behavior, customer relationship management, growth and retention, hazard based methods

1. INTRODUCTION

User attention is perceived as the most important resource in the internet era [14]. The web is described [33] as a *'virtual theme park where most rides are free such that revenue is generated through "selling eyeballs" to advertisers'*. The ad-supported economy of the web has the web-services vying for users' time rather than their money. Having a large loyal and dedicated user base has several indirect benefits as well. Many services grow with their users, improving themselves based on their feedback. The power of data analysis and machine learning methods coupled with *big data* from user activities logs have revolutionized how businesses function today. A common example is the Google search engine, which has perfected its query auto-complete feature primarily using user click-through data, as well as improved its search performance regularly using user search histories. Furthermore, an active community can be tapped to create new content that benefits the other users of the service and the service as a whole. Recent rise of the social networks such as Facebook and Twitter is a testimony to the power of user generated content and connections; development of the Wikipedia has been a collaborative effort born entirely out of user contributions.

There is tremendous competition among the rapidly increasing number of web services for the finite and limited resource corresponding to user attention. This directly results in a great deal of emphasis being placed by them on retaining and further engaging their current user base. Although, attracting new customers is crucial for any business, it is generally much easier and cheaper to retain existing customers [40, 5]. Customer retention efforts have been heavily researched in sectors such as telecommunication [40], financial services [32], insurance [30], internet services [18] and other utilities etc. which tend to follow the subscription based model. The methods in these domains have focused on identifying potential churners in the user population, where

churners are identified as those current subscribers that are not likely to renew their subscription in the coming months.

Several data mining and machine learning methods have been successfully used to classify users as potential churners or non churners [3, 31]. Also, researchers have focused on identifying and constructing useful predictors of churn behavior [13, 23]. However, such methods cannot be directly applied to solving the user retention problem for web services due to the following reasons:

1. **It is difficult to define churn for a non-contractual setting:** In a non-contractual setting a definitive action by the user to terminate one's service does not exist. To counteract the problem, some alternative definitions of churn have been proposed. Churn has been defined as a significant drop in a users' activity levels [22]. In another case, first a definition is provided for the loyal users of a service, then a user who was loyal to the service but is no longer so is defined as a churned user [31]. However, the lack of a precise definition for churn results in methods that are sensitive to the definition of churn being used.

2. **The user visitation patterns are highly dynamic:** Web services offer none or negligible switching costs to users. With no financial commitments towards a service, users switch quite frequently between different services. The highly dynamic nature of user visitation behavior makes it difficult to define typical activity volumes for a user and to segregate users as active and inactive with respect to the service.

To adapt to the incentive structures and users of dynamic activities, a novel retention metric tracking the user return rate and time is crucial to address growth and retention in web services. The user return rate is defined as the fraction of the existing users returning to the service on a particular day. It is beneficial for a web service to improve its user return rate in order to increase its revenue. Predictive analysis of user return times can direct such improvements efforts. It allows a service to identify indicators of earlier (longer) return times for their users. Identifying such indicators and the magnitude of their impact on user return times offers a service insights into its practices. It also enables a service to employ corrective measures and improve the experience to its users. Secondly, a service can identify sections of its user base that are not likely to return soon. Studies have found that the longer the users are found to stay away from a service, the less likely they are to return in the future [34]. Early identification of users who are not likely to return soon to the service allows the deployment of suitable marketing strategies to encourage those users to engage with the service again.

In this work, we address the gap between the tradition growth and retention solutions and the needs of free web services via user return time prediction. In particular, we propose a hazard model [10] from survival analysis to predict when users will return to the service. The hazard based models are preferred over the standard regression based methods due to their ability to model aspects of duration data such as censoring. More importantly, the Cox's proportional hazard regression model can incorporate the effects of covariates. We develop for the first time useful return time predictors and conclude the correlations between user behavioral and usage patterns and their return times. Such insights can potentially be very useful for a web service as a feedback for developing more engaging environments. We apply the model on real-world datasets from two popular online music services.

The rest of the paper is organized as follows. In **Section 2** we provide a brief overview of the related research in the area of churn prediction and the use of hazard based methods. We then formally define our problem and lay out our contributions in **Section 3**. In **Section 4** we describe our hazard based predictive model and provide details of the covariates used and the model estimation procedure. In **Section 5** and **Section 6** we discuss the experimentation setup and the results. We summarize the conclusions from our experimental analysis in **Section 7**. We finally conclude with future directions in **Section 8**.

2. RELATED WORK

Classical efforts in growth and retention have focused on the problem of churn prediction. Churn prediction is essentially a binary classification problem such that users are categorized based on several behavioral and demographic features into two categories: future churners or non-churners. The popular data mining techniques used for building classifiers for churn prediction include decision tress such as CART and C4.5 etc. [40], logistic regression [5], generalized additive models [8], support vector machines [9, 42] and neural networks [38, 28], though random forests [9, 41] have often been found to be superior in performance. Ensemble methods have been used to combine multiple classifiers to construct powerful meta-classifiers and to handle the class imbalance problem typical to churn prediction [4, 6, 25].

Survival analysis, a branch of statistics used for modeling time to event data, provide another class of powerful methods for Customer Relationship Management. Two types of functions from survival analysis literature are of particular interest. The survival function captures the probability of survival as a function of time where the occurrence of the event corresponds to death. The hazard function captures the instantaneous event rate given the time the event has not occurred. This class of methods differ from the previous classification based approach to churn prediction in explicitly modeling the dynamics in the churn event rate, and researchers have reported on the value of such an approach [17]. Methods from survival analysis have been used to study the event probability corresponding to customer churn given their tenure with the service [20, 11, 39, 26]. The churn event rate is found to decline with user tenure such that new users are much more likely to churn than tenured users.

In this work we do not directly predict the likelihood of churning for a user, which is dependent on the precise definition of churn used. Instead, we focus on predicting the time after which a user is expected to return to the service. We use a hazard based predictive model. The hazard based models have been extensively used for predicting similar duration data in marketing research such as interpurchase times [17, 19, 16], time until failure of new ventures [24], turnovers in employees [1] and time until adoption of new products [2], responses to promotions [15] etc. They are preferred over standard regression based approaches due to their ability to handle censored data. We specifically use the Cox's proportional hazard regression model [10] for making predictions about user return times as the Cox's model

can easily incorporate different types of covariates (including time-varying covariates) in the prediction model.

Several types of covariates have been used for churn prediction. RFM models [13] propose the use of three variables, Recency, Frequency and Monetary value of their previous interaction for identifying potential churners. Other covariates based on demographics, contractual details, service logs, use patterns, complaints, customer service responses [3, 40] have been found useful. We use some of these covariates in our model. In addition, we also incorporate user behavior related covariates in our prediction model in order to understand how user interactions while engaging with the service affect the rate of their return in the future. A special feature of our model is that it can handle the recency variable implicitly by computing the expected future time of return for the users given their length of absence from the service.

3. RETURN TIME PREDICTION PROBLEM FOR WEB SERVICES

It is difficult to define actions constituting events like user acquisition and loss for a free web service. Registering for a free web service does not entail any commitments on the part of the users due to the lack of financial investments. This is in contrast to the strong association with a service through the purchase of a subscription in the telecommunication, banking and insurance sectors. As a result, users' visitation behaviors tend to be quite flexible and arbitrary post registration. The length of the tenure of the users on a web service are in fact found to display a power law distribution with most of the users never returning back to the service [12]. In this work, we adopt a unique methodology for analyzing the dynamic user visitation data by directly modeling the user return time.

3.1 Problem Statement

We define users as belonging to either of the two activity states - the *in* and the *out* states. When users are active on the service, they are said to be in the *in* state; when they are not active on the service, they are said to be in the *out* state. In this work, we consider users to have been active on the service if they choose to visit the service on a particular day. One can define more stringent criterion on user activity on the service such as minimum time spent and/or minimum number of interactions etc.

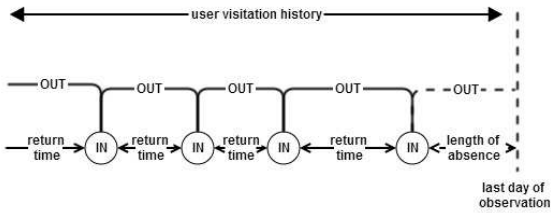


Figure 1: State Space Diagram

We focus on the problem of predicting the return time of the users which is the time the user spends in the *out* state. Since, the return time for a user can potentially extend to infinity (for users who never return back to the service), we define a threshold, t_d, on the return time. Hence, we are interested in predicting the return time for the users up to time t_d. The return time prediction problem may be formally defined as follows:

Definition 1 Given that the last time the user was in the *in* state was at time t_0, the return time prediction problem is to predict the quantity $min(t_r, t_d)$, also called the truncated return time (T_{rd}), where t_r is the total time the user spends in the *out* state and ranges from 0 to ∞ and t_d is a finite threshold on the return time; such that

(a) the user is expected to return to the *in* state at time $t_0 + t_r$, if $T_{rd} = t_r$, or

(b) the user is expected to stay in the *out* state for at least t_d units of time, if $T_{rd} = t_d$

Figure 1 provides a diagrammatic representation of the user return time prediction problem.

3.2 Time Dependence in User Return Times

Time between events have been studied extensively in queuing theory where one of the things of interest is the waiting time between customer arrival and customer service events. Customer arrivals are commonly assumed to be generated by a Poisson process such that the waiting times is modeled using the exponential distribution. An attractive property of the exponential distribution is the memoryless property. The memoryless property entails that the future rate of occurrence of the event is independent of the elapsed time; i.e. for a random variable T denoting the time of occurrence of the event, the following equation is said to hold if the memoryless property is satisfied:

$$P(T > t + s|T > s) = P(T > t) \quad (1)$$

However, several phenomena are seen to defy the simple memoryless property in interesting ways. For, example the rate of adoption of new products is found to increase with the elapsed time [35]. Alternatively, for events like responses to surveys, promotions [21] etc. the rate of these events is seen to decline with the elapsed time. The decline in future event rate with the elapsed time, has been referred to as *'inertia'*. We suspect similar type of inertia in user return behavior. For duration data showing time dependence, it becomes meaningful to compute the expected future time of the event given the elapsed time, $E(T|T > s)$. We, now define the problem of predicting the expected future time of return of the users given their length of absence (LOA) from the service.

Definition 2 Given that the last time the user was in the *in* state was at time t_0, such that he has already been in the *out* state for time t_s, the future return time prediction problem is to predict the quantity $min(t_{fr}, (t_d - t_s))$, also called the truncated future return time T_{frd}, where t_{fr} is the additional time the user spends in the *out* state and ranges from 0 to ∞ and t_d is a finite threshold on the return time; such that

(a) the user expected to return to the *in* state at time $t_0 + t_s + t_{fr}$, if $T_{frd} = t_{fr}$, or

(b) the user is expected to stay in the *out* state for atleast $t_d - t_s$ more units of time, if $T_{frd} = t_d - t_s$

3.3 Contributions

We now summarize the key contributions made by us in this paper:

(a) We formally define an approach to predictive modeling for user retention for free web services via user return time prediction.

(b) We propose the Cox's proportional hazard to model dynamic user events and incorporate the effects of covariates for return time prediction. We develop useful return time predictors and conclude correlations between user behavioral and usage patterns and their return times.

(c) The model further improves the prediction accuracy by suitably utilizing the extra information about the length of absence (LOA) of individual user from the service.

(d) The Cox's proportional hazard model outperforms state-of-the-art baselines in both expected return time prediction and user churn classification based on expected return time.

4. METHOD

We consider a time window over which user return time observations are collected. Each return time observation can be associated with a set of covariates which influence its magnitude. Hence, the data can be represented as a set of tuples: $< X, T >$ where, T is the return time observation and X is the vector of covariates associated with that observation. Since, a user can return to the service multiple times during the considered time window, we can have multiple tuples corresponding to a single user.

There are two aspects of the collected data that need special attention.

1. Censoring: Duration data which is collected over a fixed time period tends to have incomplete observations corresponding to events which were yet to happen at the end of the study period. Such observations are said to be censored. Censored observations cannot simply be discarded as this biases the analysis towards events which occur earlier. In order to capture censored observations as well, a special variable *status* is added to the representation of duration times. The *status* variable is set to 0 when the time variable represents the actual observation of return time whereas it is set to 1 when the time variable represents a censored observation. In the latter case the time duration represents the time gap between the user's last visit and the end of the study period. Hazard based methods can handle censored observations quite well. The type of censoring described here is called right censoring.

2. Recurrent observations: The collected data may contain more than one return time observation per user corresponding to multiple return events associated with him/her during the study period. Such events are called recurrent events. The active users have many more return times observations than inactive users. Throwing away multiple observations from a single user leads to loss of information. Instead, we use a simple weighting scheme for handling recurrent events. We weight each observation corresponding to a user with the inverse of the number of observations made for that user. Hence, each user has a unit weight in the data but we incorporate all observations made for him/her.

Hence, the final representation for the duration data is specified using the following tuple:$< X, T, S, W >$, where, S is the status variable and W is the weight variable.

4.1 Hazard Based Prediction Model

Survival analysis is a branch of statistics which deals with time of occurrence of events, also called duration modeling. It offers a rich set of methods which allow us to easily address questions like what is the probability that an event is going to happen after t units of time or what is the future rate of occurrence of the event given it has not happened in t units of time. In this work we deal with discrete measures of time. Two functions are useful for analysing duration information:

The survival function at time t is defined as:

$$S(t) \;=\; P(T > t) \tag{2}$$

where, T is a random variable denoting the time of occurrence of the event.

The instantaneous rate of occurrence of the event at time t, conditioned on the elapsed time t, is captured using the hazard function.

$$\lambda(t) \;=\; P(T \;=\; t \mid T \geq t) = -S'(t)/S(t-1) \tag{3}$$

The Cox's proportional hazard model is popularly used to incorporate the effect of covariates on the hazard rate. The model is based on the simple assumption that the covariates affect the magnitude of individual hazard rates but not the shape of the hazard function. Expressed mathematically,

$$\lambda(t) = \lambda_0(t) * \exp(\beta_1 * X_1(t) + \beta_2 * X_2(t) + ...) \tag{4}$$

where, λ_0 is the baseline hazard function, $X_1(t)$, $X_2(t)$, etc. are the covariates which may be static or may vary with time and β_1, β_2 etc. are the regression coefficients. The ability of the Cox's model to handle time-varying covariates can be very important for encoding such covariates in our return time prediction model. In the *Future Directions* section, later in the paper, we show how we used this feature to model the effect of external factors on user return rates.

One can obtain the survival function from the hazard function using the following equations:

$$\Lambda(t) = \sum_{0}^{t} \lambda(u)\, du \tag{5}$$

$$S(t) = \exp(-\Lambda(t)) \tag{6}$$

Λ is defined as the cumulative hazard function. The expected time of return can then be computed using the following equation:

$$E(T) = \sum_{0}^{\infty} S(t) \tag{7}$$

Furthermore, the expected future time of return given the time not returned for (t_s) can be computed as follows:

$$E(T|T > t_s) = \frac{1}{t_s} \sum_{t_s}^{\infty} S(t) \tag{8}$$

The survival function is often truncated beyond a certain point of time or when the probability of survival drops below a certain threshold in order to prevent the return time

estimate from diverging. For our prediction problem, we impose an upper bound on the return time estimate which is denoted by t_d. Hence, the equations for the expected return time and the expected future return time can be re-defined as:

$$E(T) = \sum_{0}^{t_d} S(t) \tag{9}$$

$$E(T|T > t_s) = \frac{1}{t_s} \sum_{t_s}^{t_d} S(t) \tag{10}$$

4.2 Model Estimation

The Cox's proportional hazard model is a semiparametric model as it does not assume a mathematical form for the baseline hazard function. Instead, the model can be broken down into two factors. The first factor represents the effect of the covariates on the hazard rate. The effect parameters (regression coefficients) are learnt by maximizing the partial likelihood which is independent of the baseline hazard function. Once the regression coefficients have been learnt, the non-parametric form of the baseline hazard function is estimated using the Breslow's method. Cox's seminal paper [10] is a good reference for the details of the estimation procedure.

We use the standard survival package in R for estimating the Cox's model on the training dataset. The survival package can handle weighted data instances which allowed us to directly incorporate the observation weights during the training process. We use days as the unit of time for our analysis and the threshold (t_d) for the return time prediction problem was set to 60 days, which appeared as a reasonably large threshold beyond which users tend to already be the focus of retention efforts. As a result, any return time observations larger than 60 days were assumed to be censored.

5. EXPERIMENTAL SETUP

We now evaluate the performance of the Cox's proportional hazard model for solving our proposed return time prediction problem. We evaluate the hazard based approach in two ways. Firstly, we access the performance of the model in predicting the return time of the user upto the threshold value. Secondly, we also test the performance of the approach in classifying users into buckets based on their expected return times. Such a categorization procedure is the logical next step for a service wanting to implement targeted marketing strategies for users who are not likely to return soon. For both the problems we also evaluate how well the Cox's model does at incorporating the LOA information by re-estimating the expected future return time.

5.1 Data Collection

For our experiments we use a small public and a larger proprietary dataset. The details of the two datasets are provided below:

- The Last.fm dataset. Last.fm, is an online music website catering to millions of active users. Recently, Last.fm made available, the complete music listening histories of around a 1000 of its users as recorded until May, 2009 [7]. For every song the user listened to, the dataset includes the song title, the artist name and the timestamp at which the song was heard. We use two separate time windows for creating the training and the testing datasets. All user visits observed during Oct, 2008 - Dec, 2008 were used to train our model. While we tested our model on user visits observed from Jan, 2009 - Mar, 2009. Table 1 summarizes some basic statistics about the dataset.

- Large-scale dataset. Our proposed approach was applied as a part of the growth and retention efforts for a large ad-supported music service (referred to as the Music service from now on). A dataset of around 73,465 users, collected over 3 months from May, 2012 - July, 2012, was used for training and testing our model.

	Training Data	**Testing Data**
No of users	722	752
Average no of observations per user	44.8	47.7

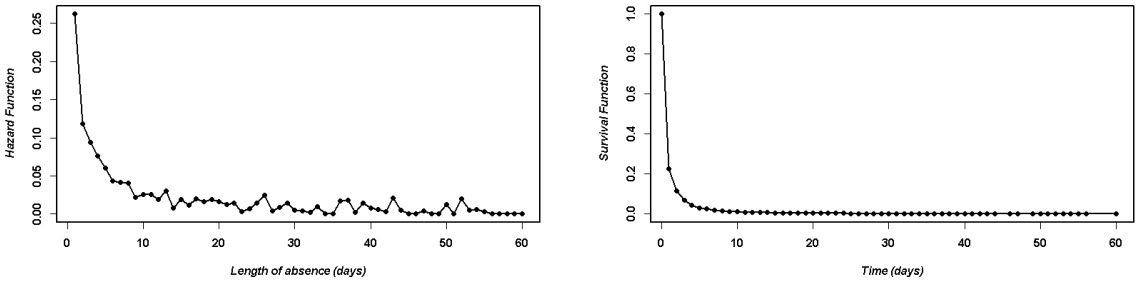
Table 1: Statistics for the Last.fm dataset

5.2 Covariates

We constructed the following covariates for the return time prediction problem.

- **Covariates related to the typical visitation patterns of a user**. Such covariates seek to predict the future return behaviour of the users based on how their visitation behaviour has been historically. For example, users who have been highly frequent in the past (loyal to the service) are likely to remain frequent in the future and similarly users who have been infrequent in the past (casual visitors) are likely to visit infrequently in the future.

  - Active Weeks: This covariate is defined as the ratio of the number of weeks since registration during which the user visited the service atleast once to the total number of weeks elapsed since registration.
  - Density of Visitation: This covariate captures the volume of user activity on the service for the weeks the user is active on the service. It is defined as the average number of days the user visited the service during the weeks the user visited the service atleast once.
  - Visit Number: This covariate is used to measure how tenured the user is with the service.
  - Previous Gap: This covariate represents the most recent return time observation (which is the gap between the user's last and prior to the last visit) for the user. For first time users this covariate is set as -1.
  - Time weighted average return time (TWRT): This covariate measures the average return time for a user. The return times are further weighted by the inverse of the length of time elapsed since they were observed under the premise that the more recent return times are more informative about the user's current visitation behavior.



(a) Baseline Hazard Function

(b) Survival Function

Figure 2: The baseline hazard function and the survival function computed on the Last.fm training dataset.

- **Covariates related to user satisfaction/engagement with the service**. Satisfaction and engagement related covariates are more difficult to construct as they attempt to capture latent user emotions about the service. Such can be extracted from any explicit (likes, dislikes, complaints etc.) or implicit (time spend, unique activities etc.) feedback indicators using user past interactions. In this work, we constructed these covariates based on user activities recorded on the last visit to the service (last *In* state)
  - Duration: This covariate captures the time spend at the service measured by the number of songs heard by the user.
  - % Distinct Songs: This covariate measures the fraction of the number of distinct songs listened by the users over the total number of songs listened by them.
  - % Distinct Artists: This covariate measures the fraction of the number of distinct artists listened by the users over the total number of songs listened by them.
  - % Skips: This covariate measures the fraction of the number of songs skipped by the users over the total number of songs listened by them. The skip information is not directly available for the Last.fm dataset. Instead, we indirectly identified skips by comparing the gap between two consecutive songs (s_1 and s_2) listened by the user with the length of the song s_1. If the time gap was found to be less than the length of song s_1 by more than 30 seconds, then song s_1 was identified to have been skipped by the user. The API, track.getInfo made freely available by Last.fm was used to retrieve the duration for the songs in the dataset.
  - Explicit feedback indicators: These covariates include information obtained directly from the users such as ratings, comments, complaints etc. Explicit feedback measures tend to be highly accurate and are an important source of information about user's satisfaction with the service. However, they are hard to acquire as providing explicit feedback requires user effort. We did not have any explicit feedback indicators for the Last.fm dataset. We had such ratings for our proprietary dataset which were included in the model.
- **Covariates used for abstracting the effects of external factors**. External factors include public holidays and weekends, marketing campaigns and promotions or personal factors which impact the rate of user return. The ability to model external factors is very useful not only because it allows quantifying the impact of these factors but also because controlling for these effects can improve the analysis performed on the other covariates. For simplicity, we have not considered any external covariates in our experiments. However we show later in future work how the Cox's proportional model can be used to model the day of the month covariate allowing us to incorporate weekly effects and holiday effects in our predictions.

5.3 Evaluation Metrics and Baselines

The instances in our test set are weighted using the same weighting scheme as applied to the training set. Each test observation was assigned a weight equal to the inverse of the number of observations from the same user, which ensures that each user has a unit weight in the test set. Different baselines are used for evaluating the performance of the Cox's model at the regression and the classification tasks. For the regression problem we compared the Cox's model against simple average (trivial baseline), linear regression, decision tree regression (RepTree), support vector machine (with a linear kernel) and neural networks (multilayer perceptron). The performance of the models were evaluated using Weighted Root Mean Square Error(WRMSE). In weighted root mean square error is computed by weighting the error between the true return time and predicted return time with the weight of the test instance. Therefore, the weighted mean square error is computed as follows:

$$WRMSE = \sqrt{\frac{\sum_{i=0}^{N} w(i) * (T^p_{rd}(i) - T_{rd}(i))^2}{\sum_{i=0}^{N} w(i)}} \qquad (11)$$

where, N is the number of test instances, $T^p_{rd}(i)$ denotes the truncated return time predicted for the i-th observation and $T_{rd}(i)$ denotes the true truncated return time the i-th observation. We can replace $T^p_{rd}(i)$ with T^p_{frd} and $T_{rd}(i)$ with $T_{frd}(i)$ for computing the WRMSE for the expected future return time predictions.

Our classification baselines included logistic regression,

random forest, support vector machine (with a linear kernel) and neural networks (multilayer perceptron). We used weighted F-measure for the minority class for measuring performance at the classification task. The weighted f-measure is defined as the harmonic mean of the weighted precision and weighted recall scores which are defined as follows. Given that sets A and P denote the set of observation belonging and predicted to belong to the minority class ;

$$\text{Weighted Precision} = \frac{\text{sum of weights of instanes in } A \cap P}{\text{sum of weights of instances in } P} \quad (12)$$

$$\text{Weighted Recall} = \frac{\text{sum of weights of instanes in } A \cap P}{\text{sum of weights of instances in } A} \quad (13)$$

The experiments for the baselines were conducted using Weka, the open source data mining software available under the GNU General Public License. The baselines were run with the default values of the parameters. Also, Weka provides support for handling weighted data instances allowing us to easily incorporate the weight vector while training the models.

6. RESULTS

In this section we analyze the results of the experimental evaluation of the Cox's model.

6.1 Model Parameters

We only discuss the parameters of model trained on the Last.fm dataset.

The importance of the covariates for the prediction problem can be assessed using different importance indicators (Table 2). The regression coefficients for the covariates and the significance of the effect of the covariates can be obtained directly from the output of R function for fitting the Cox's model. We compute the mean of the product of the covariate and its coefficient (MEAN($X * \beta$)) measured for all instances in the training set. This provides an average score for how much the covariate impacts the magnitude of the baseline hazard function. For the Last.fm dataset, we find most of the covariates associated with typical patterns of visitation for the user (Active Weeks, Density, Previous gap) to be highly significant for predicting the future time of return for the users. Also, some of the engagement/satisfaction related covariates, namely duration and % artists have significant effects on the hazard rate.

Figure 2 displays the baseline hazard function and the survival function computed for the training dataset from Last.fm. The baseline hazard function has a sharply declining shape typical of processes exhibiting inertia. This implies that the users who do not return soon to the service are likely to spend much longer time away from the service than expected without the extra information about their length of absence. Hence, it is important for a web service to ensure that the user are motivated to return back to the service soon enough. The survival function has a value of 0.0009 at 60 days. This suggests that 0.09% of users for this dataset did not return within 60 days.

6.2 Return Time Prediction

Table 3 and Table 4 display the weighted root mean square error scores obtained using the hazard based approach and the standard regression based approaches for the Last.fm dataset and the large-scale proprietary dataset respectively. We find that the hazard based approach is superior in predictive performance than the other baselines.

	Training Data (10-fold Cross Validation)	**Test Data**
Average	10.55	10.40
Linear Regression	9.61	9.37
Decision Tree Regression	9.45*	9.15*
Support Vector Machine	10.76	10.33
Neural Networks	9.58	9.36
Hazard Based Approach	8.76**	8.45**

Table 3: Weighted Root Mean Square Error for User Return Time Prediction using the Last.fm Dataset. The best and second best performing models are marked with '' and '*', respectively.**

	Training Data (10-fold Cross Validation)
Average	18.55
Linear Regression	18.33
Decision Tree Regression	18.14*
Support Vector Machine	-
Neural Networks	18.26
Hazard Based Approach	16.58**

Table 4: Weighted Root Mean Square Error for User Return Time Prediction using the Proprietary Dataset. The best and second best performing models are marked with '' and '*', respectively. The SVM method could not be implemented on the proprietary dataset because of its high computational cost.**

As discussed earlier, the hazard based approach allows us to compute the expected future return time for a user given their length of absence (LOA) by incorporating the dynamics in the hazard function. We evaluate the performance of the hazard-based approach in updating its prediction given the LOA values. Since the standard regression approach do not provide the same feature, we re-learn those model by incorporating the LOA values for the users as a separate feature. The values for this feature were generated by replicating each return time observation T, T times for all values of LOA ranging from (0) $-$ $(T-1)$. The future return time were appropriately reassigned to range from (T) $-$ (1). Doing so significantly increased the size of the dataset. The data instances were re-weighted to ensure that each user still holds a unit weight in the test and the training sets. Due to space limitations we only show the comparisons between two of our baselines: decision tree regression, which is our best performing baseline and linear regression, because of its ease of use; with the hazard based approach for the large-scale proprietary dataset. We find that the hazard based approach is superior than decision tree regression and linear regression in estimating the expected future return time (Fig.3).

6.3 Classification into user buckets

The users are classified into different categories based on their predicted return times. For the Last.fm dataset we bucketed users based based on their predicted return times

Covariates	Coefficient	Significance	MEAN$(X * \beta)$
Active Weeks	9.313e-02	0.0214 *	0.4370
Density	2.366e-01	1.05e-13 ***	1.2437
Visit Number	4.941e-05	0.7318	2.336e-02
Previous Gap	-5.175e-03	0.00147 **	-0.01222
TWRT	-1.484e-02	0.28174	-0.02492
Duration	1.315e-03	0.02538 *	0.06171
% Distinct Songs	6.849e-02	0.7653	0.06040
% Distinct Artists	-2.251e-01	0.08553 .	-0.1064
% Skips	3.740e-01	0.23229	0.04873

Table 2: Covariate Importance Indicators for the Last.fm Dataset. Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.'**

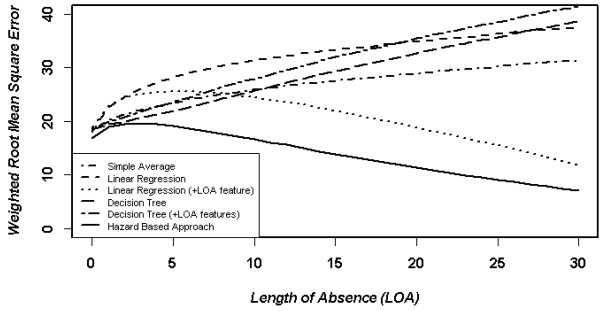
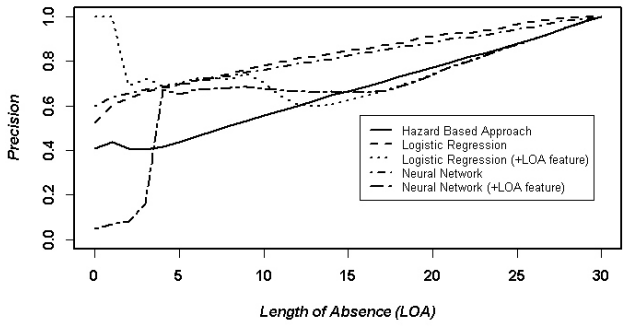


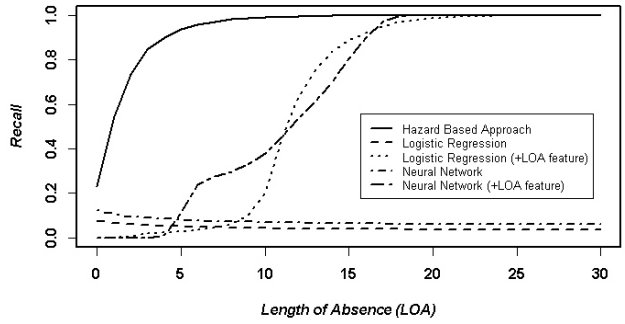
Figure 3: Weighted Root Mean Square Error for different values of LOA for the large-scale proprietary dataset

being larger or within 7 days, while for the larger proprietary dataset we classified them based on their predicted return times being larger or within 30 days. The shorter time period was used for the Last.fm dataset due to scarcity of users in the test set that returned after 7 days. Table 5 and Table 6 provide the performance scores for the hazard based approach and the other baselines for classifying instances into the minority class for the Last.fm and the proprietary datasets. Although, the hazard based model is not learnt as a classification model, it still performs superior to the the state-of-the-art baselines for our proprietary dataset and is comparable in performance to the best performing baselines for our Last.fm dataset.

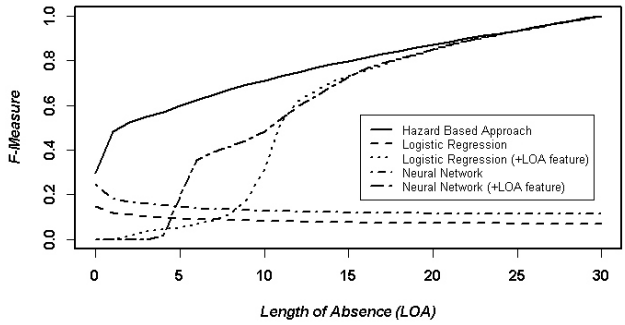
We also evaluate the performance of the hazard based approach in classifying users into buckets given the LOA values for the user. Again, the classification baselines do not offer similar capabilities for updating their prediction scores given LOA values. Hence, we incorporate LOA values as an additional feature for classification and replicate instances to populate the values for the feature as done for the standard regression methods earlier. As seen for the regression analysis, this approach considerably increases the size of the dataset. To avoid clutter we provide comparison results against the best performing baseline classification approaches - logistic regression and neural networks. We find that the hazard-based approach can incorporate the LOA information and update its prediction much effectively as compared to both logistic regression and neural networks (Fig. 4).



(a)



(b)



(c)

Figure 4: Figures (a), (b) and (c) are the plots of the weighted precision, recall and f-measure scores respectively, for different values of LOA for the Large-Scale Proprietary Dataset

	Training Data (10-fold Cross Validation)			Test Data		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
Random Forest	0.64	0.24	0.35	0.72	0.29	0.41
Logistic Regression	0.68	0.44	0.53**	0.66	0.40	0.50*
Support Vector Machine	0.61	0.11	0.18	0.82	0.15	0.25
Neural Networks	0.77	0.39	0.52*	0.71	0.36	0.48
Hazard Based Approach	0.39	0.79	0.52*	0.37	0.81	0.51**

Table 5: Weighted precision, recall and f-measure scores for the minority class (expected return time > 7) for the Last.fm Dataset. The best and second best performing models are marked with '' and '*', respectively.**

	Training Data (10-fold Cross Validation)		
	Precision	Recall	F-Measure
Random Forest	0.47	0.10	0.18
Logistic Regression	0.52	0.08	0.15
Support Vector Machine	0	0	0
Neural Networks	0.48	0.17	0.25*
Hazard Based Approach	0.41	0.23	0.29**

Table 6: Weighted precision, recall and f-measure scores for the minority class (expected return time > 30) for the Large-scale Proprietary Dataset. The best and second best performing models are marked with '' and '*', respectively.**

7. DISCUSSION

In this work, we have formulated the return time prediction problem which aims to predict the next time of return for the users. We have formally defined two prediction problems for return time prediction: (a) predicting the expected time of return of the users (b) predicting the expected future time of return for the users given their LOA. We propose the Cox's proportional hazard model as the single solution for solving both these problems effectively. We are interested in not only the performance of this model at return time prediction but also in classifying users into categories based on their predicted return time. We now discuss how well the Cox's model is found to meet our objectives.

The Cox's model was found to perform superior to the standard regression based approaches in predicting the return time of the users. Since, standard regression based approaches cannot handle censored data, they are limited to only those observations that are complete. This introduces a bias while learning the model and can potentially be the reason behind their lower performance as compared to the hazard based approach which can incorporate censored observations quite well. Also, we evaluated the ability of the Cox's model in classifying users into buckets based on their predicted return times. The Cox's model is at a disadvantage as compared to the other classification approaches as it is not trained to identify class boundaries as other classification methods are. However, it still performs performs either superior or comparably well to the the state-of-the-art baselines on both our datasets. The classification based methods suffer from additional disadvantage as compared to the regression based approaches. The classifiers are dependent on cut off used to categorize users into buckets. Any potential change in the bucketing policy would require retraining the classifiers. Also, if multiple users buckets need to be generated, different classifier would have to be trained for each bucket specification. A single regression based model is however, flexible enough to handle any bucketing criterion post training.

We were also interested in evaluating whether our return time prediction model can improve the prediction performance given LOA values for the users. The LOA values is a crucial piece of information and it is available as users remain away from the system. A plot of the prediction performance scores against the LOA values allows a service to identify the right amount of gap since the user's last visit needed to start retention efforts. We find that the Cox's model is better at using LOA information to improve its performance at both the regression and the classification tasks. For the classification problem, the Cox's model can reach a very high recall and a good precision score much sooner than the baseline approaches. It is also important to note that the Cox's model needs not be retrained to handle the LOA information, while the standard regression and classification approaches had to. Also, the data had to be considerably duplicated to populate the values for the LOA covariate while training the standard regression and classification approaches.

8. FUTURE DIRECTIONS

In this work we have provided a solution to the return time prediction problem using the Cox's proportional hazard model. The hazard-based prediction model proposed by us can however, be refined in multiple ways to incorporate the complexities of real data. We describe a few such improvements here.

8.1 Incorporating Time-varying Covariates

Our analysis till now has been limited to static covariates. However, some covariates effecting the user return rate may be dependent on time. An example of time-varying covariates include those pertaining to external factors such as holiday and weekend effects, promotional offers, launch of marketing campaigns etc. In our final model for the Music Service, we incorporated the effect of the day of the month covariate on the user return rates as we found from our exploratory analysis that the number of users visiting the service differed significantly on different days. The Cox's model can easily accommodate such time-varying covariates [36] and several applications of these have been proposed before [5, 37] etc.

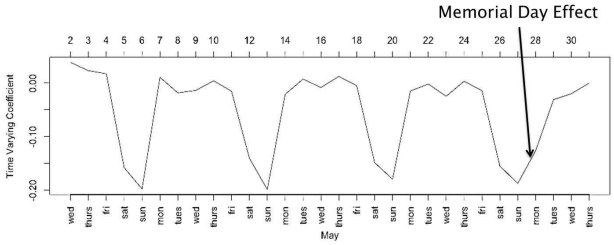


Figure 5: The regression coefficient for time-varying covariates corresponding to the different days of the month

Figure 5 shows the regression coefficient for the different days of the month for the month of May, 2012 computed using the large-scale proprietary dataset. The absolute values are omitted here. Its easy to spot the weekly cycles in the user visitation behavior and the anomaly in the systematic pattern which corresponded to a holiday.

8.2 User Segmentation

The Cox's proportional hazard model is based on the proportional hazard assumption, which implies that the covariates only affect the magnitude of the hazard function and not its shape. Some covariates may not follow the assumption and can be handled either using time-varying covariates or by segmenting the population based on different value of the covariate such that different proportional hazard models are learnt for different segments of the population. We consider the segmentation approach here. For our particular application we found it useful to segment the population based on user tenure with the service and whether or not the users were visiting the service after a long gap. We constructed the following four segments:

1. First Time Visitors
2. New visitors (Who have been less than 3 months with the service)
3. Old visitors (who have been more than 3 months with the service) and visiting after a long time
4. Old visitors (who have been more than 3 months with the service) and those who have been quiet frequent recently

Comparing and contrasting the proportional hazard models learnt for the different segments of the population revealed interesting similarities and differences in different types of users.

The final model prepared for the Music Service was implemented as a part of a daily process that produced 2 separate lists of users (a) users who were not expected to return within 7 days (b) users who were not expected to return within 30 days. Different marketing strategies could then be applied to the two lists of users.

Other directions for future work include accounting for heterogeneities among users. The survival analysis literature offers several solutions for either controlling for such differences between users [29] or for extracting different users segments through clustering [27]. Such approaches can be also be applied to the return time prediction problem.

9. REFERENCES

[1] Gerald J Adams. Using a cox regression model to examine voluntary teacher turnover. *The Journal of experimental education*, 64(3):267–285, 1996.

[2] Frank M Bass. Comments on âĂIJa new product growth for model consumer durables the bass modelâĂİ. *Management science*, 50(12 supplement):1833–1840, 2004.

[3] Alex Berson, Stephen Smith, and Kurt Thearling. *Building data mining applications for CRM*. McGraw-Hill New York, 2000.

[4] Zoheb Borbora, Jaideep Srivastava, Kuo-Wei Hsu, and Dmitri Williams. Churn prediction in mmorpgs using player motivation theories and an ensemble approach. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 157–164. IEEE, 2011.

[5] Wouter Buckinx and Dirk Van den Poel. Customer base analysis: partial defection of behaviourally loyal clients in a non-contractual fmcg retail setting. *European Journal of Operational Research*, 164(1):252–268, 2005.

[6] Jonathan Burez and Dirk Van den Poel. Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 36(3):4626–4636, 2009.

[7] O. Celma. Music recommendation datasets for research. 2010.

[8] Kristof Coussement, Dries F Benoit, and Dirk Van den Poel. Improved marketing decision making in a customer churn prediction context using generalized additive models. *Expert Systems with Applications*, 37(3):2132–2143, 2010.

[9] Kristof Coussement and Dirk Van den Poel. Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert systems with applications*, 34(1):313–327, 2008.

[10] David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 187–220, 1972.

[11] James H Drew, DR Mani, Andrew L Betz, and Piew Datta. Targeting customers with statistical and data-mining techniques. *Journal of Service Research*, 3(3):205–219, 2001.

[12] Gideon Dror, Dan Pelleg, Oleg Rokhlenko, and Idan Szpektor. Churn prediction in new users of yahoo! answers. In *Proceedings of the 21st international conference companion on World Wide Web*, pages 829–834. ACM, 2012.

[13] Peter S Fader, Bruce GS Hardie, and Ka Lok Lee. Rfm and clv: Using iso-value curves for customer base analysis. *Journal of Marketing Research*, pages 415–430, 2005.

[14] Rishab Aiyer Ghosh. Cooking pot markets: an economic model for the trade in free goods and services on the internet. *First Monday*, 3(2), 1998.

[15] Füsun F Gönül, Byung-Do Kim, and Mengze Shi. Mailing smarter to catalog customers. *Journal of Interactive Marketing*, 14(2):2–16, 2000.

[16] Sunil Gupta. Stochastic models of interpurchase time with time-dependent covariates. *Journal of Marketing*

Research, pages 1–15, 1991.

[17] Kristiaan Helsen and David C Schmittlein. Analyzing duration times in marketing: Evidence for the effectiveness of hazard rate models. *Marketing Science*, 12(4):395–414, 1993.

[18] Bing Quan Huang, M Tahar Kechadi, and Brian Buckley. Customer churn prediction for broadband internet services. In *Data Warehousing and Knowledge Discovery*, pages 229–243. Springer, 2009.

[19] Dipak C Jain and Naufel J Vilcassim. Investigating household purchase timing decisions: A conditional hazard function approach. *Marketing Science*, 10(1):1–23, 1991.

[20] Zainab Jamal and Randolph E Bucklin. Improving the diagnosis and prediction of customer churn: A heterogeneous hazard modeling approach. *Journal of Interactive Marketing*, 20(3):16–29, 2006.

[21] Michael D Kaplowitz, Timothy D Hadlock, and Ralph Levine. A comparison of web and mail survey response rates. *Public opinion quarterly*, 68(1):94–101, 2004.

[22] Marcel Karnstedt, Tara Hennessy, Jeffrey Chan, and Conor Hayes. Churn in social networks: A discussion boards case study. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, pages 233–240. IEEE, 2010.

[23] Hee-Su Kim and Choong-Han Yoon. Determinants of subscriber churn and customer loyalty in the korean mobile telephony market. *Telecommunications Policy*, 28(9):751–765, 2004.

[24] Erkki K Laitinen. Prediction of failure of a newly founded firm. *Journal of Business Venturing*, 7(4):323–340, 1992.

[25] Aurélie Lemmens and Christophe Croux. Bagging and boosting classification trees to predict churn. *Journal of Marketing Research*, pages 276–286, 2006.

[26] Junxiang Lu and O Park. Modeling customer lifetime value using survival analysisâĂŤan application in the telecommunications industry. In *Proceedings of the SAS Conference*. Citeseer, 2003.

[27] Patrick Mair and Marcus Hudec. Analysis of dwell times in web usage mining. In *Data Analysis, Machine Learning and Applications*, pages 593–600. Springer, 2008.

[28] Brij Masand, Piew Datta, DR Mani, and Bin Li. Champ: A prototype for automated cellular churn prediction. *Data Mining and Knowledge Discovery*, 3(2):219–225, 1999.

[29] CA McGilchrist and CW Aisbett. Regression with frailty in survival analysis. *Biometrics*, pages 461–466, 1991.

[30] Katharina Morik and Hanna Köpcke. Analysing customer churn in insurance data–a case study. In *Knowledge Discovery in Databases: PKDD 2004*, pages 325–336. Springer, 2004.

[31] Scott A Neslin, Sunil Gupta, Wagner Kamakura, Junxiang Lu, and Charlotte H Mason. Defection detection: Measuring and understanding the predictive accuracy of customer churn models. *Journal of Marketing Research*, pages 204–211, 2006.

[32] Joe Peppard. Customer relationship management (crm) in financial services. *European Management Journal*, 18(3):312–327, 2000.

[33] Jeffrey F Rayport. The truth about internet business models. *Strategy and Business*, pages 5–7, 1999.

[34] Werner J Reinartz and Vijay Kumar. On the profitability of long-life customers in a noncontractual setting: An empirical investigation and implications for marketing. *The Journal of Marketing*, pages 17–35, 2000.

[35] David C Schmittlein and Vijay Mahajan. Maximum likelihood estimation for an innovation diffusion model of new product acceptance. *Marketing science*, 1(1):57–78, 1982.

[36] Terry Therneau and Cindy Crowson. Using time dependent covariates and time dependent coefficients in the cox model. *The Survival Package (R help guide)*, 2013.

[37] Lu Tian, David Zucker, and LJ Wei. On the cox model with time-varying regression coefficients. *Journal of the American statistical Association*, 100(469):172–183, 2005.

[38] Chih-Fong Tsai and Yu-Hsin Lu. Customer churn prediction by hybrid neural networks. *Expert Systems with Applications*, 36(10):12547–12553, 2009.

[39] Dirk Van den Poel and Bart Larivière. Customer attrition analysis for financial services using proportional hazard models. *European Journal of Operational Research*, 157(1):196–217, 2004.

[40] Chih-Ping Wei, I Chiu, et al. Turning telecommunications call details to churn prediction: a data mining approach. *Expert systems with applications*, 23(2):103–112, 2002.

[41] Yaya Xie, Xiu Li, EWT Ngai, and Weiyun Ying. Customer churn prediction using improved balanced random forests. *Expert Systems with Applications*, 36(3):5445–5449, 2009.

[42] Yu Zhao, Bing Li, Xiu Li, Wenhuang Liu, and Shouju Ren. Customer churn prediction using improved one-class support vector machine. In *Advanced data mining and applications*, pages 300–306. Springer, 2005.