Reserve Price Failure Rate Prediction with Header Bidding in Display Advertising

Achir Kalra[*][†]
Forbes Media LLC
Jersey City, New Jersey
akalra@forbes.com

Chong Wang[†]
S&P Global
New York, New York
chong.wang@spglobal.com

Cristian Borcea
New Jersey Institute of Technology
Newark, New Jersey
borcea@njit.edu

Yi Chen
New Jersey Institute of Technology
Newark, New Jersey
yi.chen@njit.edu

ABSTRACT

The revenue of online display advertising in the U.S. is projected to be 7.9 billion U.S. dollars by 2022. One main way of display advertising is through real-time bidding (RTB). In RTB, an ad exchange runs a second price auction among multiple advertisers to sell each ad impression. Publishers usually set up a reserve price, the lowest price acceptable for an ad impression. If there are bids higher than the reserve price, then the revenue is the higher price between the reserve price and the second highest bid; otherwise, the revenue is zero. Thus, a higher reserve price can potentially increase the revenue, but with higher risks associated. In this paper, we study the problem of estimating the failure rate of a reserve price, i.e., the probability that a reserve price fails to be outbid. The solution to this problem have managerial implications to publishers to set appropriate reserve prices in order to minimizes the risks and optimize the expected revenue. This problem is highly challenging since most publishers do not know the historical highest bidding prices offered by RTB advertisers. To address this problem, we develop a parametric survival model for reserve price failure rate prediction. The model is further improved by considering user and page interactions, and header bidding information. The experimental results demonstrate the effectiveness of the proposed approach.

CCS CONCEPTS

• Mathematics of computing → Survival analysis; **• Theory of computation → Computational advertising theory**;

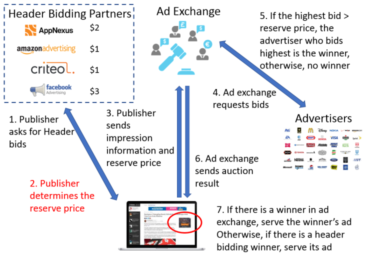
KEYWORDS

Computational Advertising; Header Bidding; Survival Analysis

ACM Reference Format:
Achir Kalra, Chong Wang, Cristian Borcea, and Yi Chen. 2019. Reserve Price Failure Rate Prediction with Header Bidding in Display Advertising. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3292500.3330729

1 INTRODUCTION

The revenue of online display advertising in the U.S. is projected to be 7.9 billion U.S. dollars by 2022, and it has become the most critical revenue source for online publishers [9]. Display advertising comes in the form of banner ads, rich media, and more.



Header Bidding Partners
AppNexus $2
amazonadvertising $1
criteo $1
facebook Advertising $3
Ad Exchange
5. If the highest bid > reserve price, the advertiser who bids highest is the winner, otherwise, no winner
4. Ad exchange requests bids
Advertisers
1. Publisher asks for Header bids
3. Publisher sends impression information and reserve price
6. Ad exchange sends auction result
2. Publisher determines the reserve price
7. If there is a winner in ad exchange, serve the winner's ad Otherwise, if there is a header bidding winner, serve its ad

Figure 1: Ad Impression Selling in RTB

In addition to assigning ad inventory based on offline contracts with advertisers (i.e., direct sale), publishers sell ad opportunities in real-time through *real-time bidding (RTB).* When a user requests a webpage on a publisher's website, a *page view* is displayed on a screen. One display of an ad in a page view is called an *ad impression.* Figure 1 (steps 2-7) shows the basic process of ad impression selling (step 1 is an improvement that will be discussed after the basic process). For each ad impression, the publisher first determines a reserve price, the lowest price acceptable for an ad impression (step 2). Then, the publisher sends the impression information (e.g. user profile, page metadata) and the *reserve price* to an ad exchange (step 3). An ad exchange, denoted as *adx* in the rest of this paper, is a digital marketplace that enables advertisers and publishers to buy and sell advertising space, often through real-time auctions. Adx requests bids from advertisers (step 4). Most RTB ad exchanges use the second-price auction model (step 5), where the highest bidder wins if the bidding price is higher than the reserve

[*]The author is also with the Department of Computer Science, New Jersey Institute of Technology, USA.
[†]Both authors contributed equally to the paper.

KDD '19, August 4–8, 2019, Anchorage, AK, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330729

price. The price paid is the higher between the second-highest bid and the reserve price [17]. If there is no bidding price higher than the reserve price, then the auction fails. If there is a winner in the auction (step 6), the winner's ad is shown as an ad impression.

An effective strategy of setting the reserve price can significantly increase publishers' revenue. Figure 2 illustrates the three RTB cases, based on the relationship among the reserve price r, the highest bid b_1, the second highest bid b_2, and the revenue generated.

- In Case 1, $r < b_2 < b_1$, the revenue is b_2.
- In Case 2, the publisher increases the reserve price r such that $b_2 < r <_b 1$; the revenue will be r, which gives a revenue boost of $r - b_2$ compared to case 1.
- In Case 3, $b_1 < r$, the publisher increases r too much, such that no advertisers in the auction are willing to bid more than r; the publisher obtains no revenue.

The objective of publishers is to set r as close as possible to b_1 but never higher than b_1. Then, the revenue will be boosted to nearly b_1. However, since publishers do not know a priori b_1 before an adx auction, it is difficult to set an optimal r.

To solve the problem of accurately predicting the reserve price value, we leverage *header bidding*, a recent improvement to the basic RTB process. Before selling impressions in ad exchanges, publishers send impression information to multiple *header bidding partners* to conduct an impression header auction (step 1 in Figure 1). Header auctions adopt first-price auction, instead of second-price auction as in RTB [8]. Without header bidding, advertisers can only bid leftovers after more premium channels, e.g., sponsored lines. With header bidding, advertisers have the benefit of looking first at the entire ad inventory: When a page loads, header bidding partners are called for all impressions in the pageview. Publishers can have more transparency into how much their impressions are worth and, thus, design adjust the reserve prices to increase revenue.

A simple strategy to benefit from header bidding information when setting the reserve prices is to always set r to h_1, the highest bidding price in header bidding (if available). In the example in Figure 1, the publisher can set r = \$3. If the publisher receives a higher winning bid in the RTB auction, the impression goes to the RTB winner; otherwise, it goes to the header bidding winner. If neither RTB nor header bidding has a winner, it triggers for a house ad or unfilled impression. This rarely happens in reality. Hence, no matter which case in Figure 2 occurs, the revenue is guaranteed to be $max(h_1, b_2)$. However, this is not optimal: it can be improved by predicting a better reserve price r', such that $b_2 \leq r' \leq b_1$. Thus, the expected revenue can be further boosted to $max(h_1, r')$. On the other hand, increasing r' increases the chance that r' fails to be outbid in adx and can be risky.

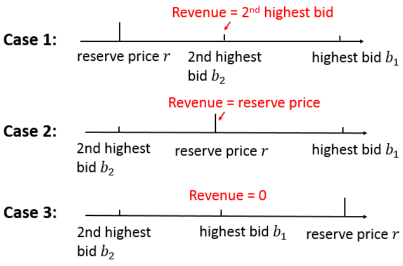


Figure 2: Impression Revenue in the Second Price Auction

The goal of this paper is to propose effective machine learning models for *r*eserve price failure rate prediction: Given information about an ad impression and a reserve price, the model outputs the probability that no advertisers in the RTB ad exchange will outbid the reserve price. The outcome of our approach has managerial implications for publishers to set appropriate reserve prices to maximize the expected revenue. It can also help publishers to find a balance between taking risks and maximizing revenue. For example, a conservative publisher may prefer to set a reserve price with a 20% failure rate, while a risk lover may set it to 80%.

Reserve price failure rate prediction is challenging due to three reasons. First, publishers do not know the bidding prices offered by RTB advertisers in past auctions, making the prediction of bidding price (and hence of the reserve price) for a target ad impression very challenging. We term this challenge censorship because the publishers are "censored" from knowing the highest bidding price; they just know the revenue generated by each ad impression (i.e., the second highest bid). There are two types of censorship in publishers' impression transaction data: left-censoring, where the unobserved highest adx bid is less than the known reserve price; and right-censoring, where the unobserved highest adx bid is greater than the observed revenue. Second, publishers do not have access to personally identifiable information of users, and thus do not know about users as much as advertisers and Demand-side Platforms (i.e. systems that help advertisers to buy impressions in real time). Without user profile information, it is difficult for publishers to predict the advertisers' bidding price in order to set the reserve appropriately. Third, although intuitively header bidding information can help with RTB prediction, it is not clear how to utilize this information. To the best of our knowledge, there is no existing work using header bidding in RTB-related predictions.

To address the censorship challenge, we propose to use parametric survival models. Unlike binary regressions, which cannot handle datasets with both censored and uncensored data, parametric survival models are more generic and can handle any dataset with or without uncensored data [13]. To deal with the challenge of limited user profile information, we use latent vectors to capture feature characteristics and add factorized pairwise interaction between users and pages in the objective function. For the header bidding challenge, we exploit the similarities between header bidding and RTB in adx and propose to improve the prediction model using header bidding regularization.

The paper presents empirical results of the proposed approach on a real dataset from Forbes Media, a large online publisher, which logs daily ad impression transaction data. The concordance index (C-Index) is the standard performance measure for model assessment in survival analysis. We develop a customized C-Index for datasets containing only left- and right-censored instances. The experiments show that our models with the Weibull distribution significantly outperform the baselines, i.e., a Kaplan-Meier model and a logistic regression with observed reserve price/revenue as the feature. Adding interaction factorization and header bidding regularization reduces log-loss compared with the best baseline by 67%.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3.3 presents the proposed parametric survival model for reserve price failure rate prediction; this section also includes a description of the real-life dataset, and the two improvements using pairwise interaction tensor factorization and header bidding regularization. Experimental results and insights derived from these results are presented in Section 4. The paper concludes in Section 5.

2 RELATED WORK

As a significant problem for online publishers, reserve price optimization has been studied in the past few years. Li et al. [15] evaluated the impact of reserve price on publisher revenue in real-time bidding. Yuan et al. [28] proposed a simple game-theoretic-based approach to obtain an optimal reserve price on the publishers' side. Mohri et al. [18] assumed that both the highest bids and the second highest bids are observed by publishers, and then proposed a machine learning approach to optimize publishers' real-time bidding revenue. Cesa-Bianchi et al. [5] showed a regret minimization algorithm for setting the reserve price in online second-price auctions. Austin et al. [4] described a scalable linear-function-based reserve price optimizer for real-time bidding. Medina et al. [19] assumed that bids are observed by publishers, and directly optimize the revenue by reducing reserve price optimization to the standard setting of prediction under squared loss, which can be conveniently minimized. Xie et al. [27] came up with an efficient method of improving the publisher revenue by mainly focusing on adjusting the reserve price for the high-value impressions. Jauvion et al. [11] proposed a new online learning algorithm based on classical multi-armed bandit strategies. However, these existing studies assume that publishers know the highest winning bids and the second highest bids of historical impressions, which is not the case for most publishers. Most of them evaluate their approaches using either synthetic data (e.g., [18] and [19]) or ad exchange data (e.g., [4]). In contrast, this paper addresses the problem based on censored data available to most publishers. Furthermore, our evaluation is done using real-life data from a large publisher.

Existing related work that takes censorship into account may have datasets that contain different combinations of right-, left-, and/or uncensored data, due to different practices and platforms used among publishers. Alcobendas et al. [3] proposed a game-theoretic-based model to optimize reserve prices in the context of online video advertising with left- and uncensored data. The model considers information about auctions (e.g., the number of bids higher than the reserve price) as input. However, such information is typically only available for the publishers that own ad exchanges (e.g., Google and Yahoo). Chahuara et al. [6] adopted a relatively simple non-parametric regression model of auction revenue based on an incremental time-weighted matrix factorization. The authors used an online extension of the Aalen's Additive model to estimate the first and second bids' distribution. However, this method cannot handle left- and right-censored data at the same time. In addition, the method has to discretize reserve prices due to the limitation of the Aalen's Additive model. In practice, feature binning is tricky and makes the model inflexible, as it is hard to determine the boundary of bins. Also, the method considers only user IDs and placement IDs. Thus, the model cannot make predictions for infrequent users [14].

In addition to reserve price optimization, another family of related work is the prediction of winning prices at the advertisers' side [7, 25, 26, 29]. These studies predict the adx bid of a given impression for an advertiser or a Demand-Side Platform (DSP) to win the ad opportunity. Reserve price optimization and RTB winning price prediction share a common feature: both evaluate the values of impressions. However, the algorithms in existing studies on winning price prediction do not make probabilistic predictions. Also, advertisers and DSPs typically know the highest bids (i.e., their own bids) when they win the impressions. However, most publishers are not able to access the highest adx bids. This makes the problem more challenging.

Compared with the existing studies, our work predicts the reserve price failure rate using data available at the publishers' side, which does not include details about the adx auctions. Our method is applicable to data with any type of censorship faced by a publisher. In addition, our proposed model outputs probabilistic predictions.

3 RESERVE PRICE FAILURE PREDICTION

This section first introduces our real-life dataset (Section 3.1) and discusses the data censorship (Section 3.2). It then presents our parametric survival model (Section 3.3) with factorized pairwise interactions (Section 3.4) and header bidding regularization (Section 3.5).

3.1 Real-Life Datasets

We use two datasets collected in one day in April 2018 at Forbes Media's website: 1) NetworkImpressions and 2) NetworkBackfillImpressions [1]. These two datasets are provided by Google DoubleClick for Publishers (DFP) [1]. The data preparation is illustrated in Figure 3. The final dataset contains above 16 million impressions with 2.6 million unique users and 132 thousand unique pages.

NetworkImpressions records the impressions which were allocated to direct sale or header bidding winners. Headerbidding-won impressions failed to receive higher bids during RTB than the highest header bids (which is the reserve price of the headerbidding-won impressions in our dataset). Unlike headerbidding-won, directsale-won impressions were not sent to RTB. They were never bid by advertisers. Therefore, we filter out direct sale impressions by their order IDs. We get nearly 6 million headerbidding-won impressions.

NetworkBackfillImpressions records the impressions which were allocated to real-time bidding winners. In the context of this paper, it contains adx-won impressions. It includes fields such as time, pageID, ad position, channel, section, geo-location, and header bids [2]. In addition, it records the revenue and reserve price of each impression. We obtain about 10 million adx-won impressions.

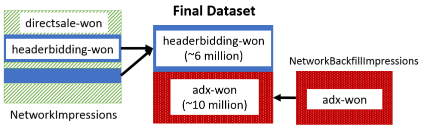


Figure 3: Data Preparation

[1] The description and data samples are available at https://support.google.com/admanager/answer/1733124

[2] The default data report does not contain header bids. This information added by the publisher.

Although our method is based on Google DFP datasets (which has a dominant market share), it can be adopted by a large majority of the other publishers as well. This is because DFP's competitors (e.g., OpenX [2]) provide similar datasets, and most publishers face the same restrictions/limitations in the datasets (e.g., censorship).

3.2 Data Censorship

The dataset has two types of impressions: adx-won and headerbidding-won. In the rest of the paper, they are referred to as *reserve-won* and *header-won*, respectively. Without loss of generality, they are subdivided into three censorship cases in Figure 4.

header-won is similar to case 3 in Figure 2. The only difference is that, with header bidding available, the revenue is the maximum header bid. To increase revenue, a publisher wants to decrease the reserve price, so that it may be outbid in adx, and then the revenue will be the higher value between the revenue from adx and header bidding. Since the highest bid was not disclosed to publishers, the impressions are left-censored at the reserve prices.

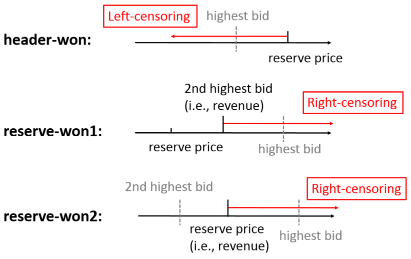


Figure 4: Three censorship cases. The black variables are observed, while the grey ones are unobserved.

reserve-won1 corresponds to case 1 in Figure 2. The reserve price and the revenue (i.e., the second highest bid) are known, but the highest bid is unknown. *Reserve-won1* impressions are right-censored at the second highest bid.

reserve-won2 corresponds to case 2 in Figure 2. Only the revenue (i.e. the reserve price) is known. That is, *reserve-won2* impressions are right-censored at the reserve price.

For both *Reserve-won1* and *reserve-won2*, to increase revenue, the publisher wants to increase the reserve price to be close to (but not exceed) the highest bid, which will then be the revenue [20].

3.3 The Proposed Parametric Survival Model

Problem Definition. *Given an ad impression A_i and a reserve price t_i, the goal is to predict the probability that no advertiser is willing to bid higher than t_i during RTB.*

Model Selection. Since our data are censored, it is natural to use survival analysis models [24]. The goal of survival analysis is to estimate the probability of a time-to-event of interest for a new instance with feature predictors. For instance, it can answer a question such as how likely it is that a patient has a disease at a certain point in time. To apply survival analysis to our case, we can make an analogy: one impression is an instance, which has a set of features. The event of interest is that all advertisers bid lower than the reserve price (i.e., reserve price failure). The time to event is the reserve price. With the increase in the reserve price (i.e., time to event), the probability of reserve price failure (i.e., event of interest) also increases. When the reserve price is $0, it is most likely that the event of reserve price failure does not occur. If the reserve price is high (e.g., $100), it can hardly receive a higher bid.

One important feature of survival analysis models is that they can output probabilistic predictions. In our application, they can both infer the optimal reserve price and tell how likely it is that a better bid can be received from advertisers. This is helpful for publishers that would like to perform revenue optimization and/or find the trade-off between risk and return.

Survival analysis models have been used in many applications [24]. As a widely used semi-parametric survival model, Cox proportional hazards model [13] assumes that the baseline hazard is a function of time to event t, but does not involve the feature predictors X. Thus, it is not necessary to specify the form of the baseline hazard (i.e., how the probability of event occurrence is changing with t).

Although the Cox proportional hazards model is popular due to its flexibility, it is insufficient for our application for several reasons. First, it does not directly accommodate left- or interval-censored data (it does handle right-censor data, though). Second, in model inference, partial likelihood requires observed data whose times to event are known. However, our dataset consists of both right- and left-censored data and does not contain any observed data, i.e., the highest adx bids of all impression are not accessible on the publishers' side. In addition, we need to deal with continuous time points because the reserve price is a continuous variable. Also, as the size of the impression data is huge, our model should be able to be learned stochastically.

The Proposed Method. We propose to use the parametric likelihood of a *parametric survival model* since it can easily accommodate left- and right-censored data or uncensored data. Furthermore, parametric survival models are more general compared to binary regressions, which cannot handle data containing both censored and uncensored instances [13].

A parametric survival model is one in which the outcome is assumed to follow a known distribution family. In other words, it assumes the probability that no advertiser bids higher than a reserve price follows a certain distribution. Commonly used distributions include Weibull, Exponential, and Log-logistic. Individual instances typically share the same family of distributions of similar form but with different parameters. Distribution parameter values are determined based on the feature predictors of instances.

Left- and right-censoring can be considered as special cases of interval-censoring (with zero as the lower bound and positive infinity as the upper bound). Hence, without loss of generality, given the ith ad impression, we aim to predict the probability that the reserve price t_i fails within the interval $[a_i, b_i]$.

$$P(a_i \le t_i \le b_i) = \int_{a_i}^{b_i} f(t)dt \tag{1}$$

where $P(a_i \le t_i \le b_i)$ is the probability that the true failure reserve price t_i is between a_i and b_i. $f(t)$ is the probability density function (PDF) of the specified distribution. a_i and b_i are the lower bound and the upper bound of the reserve price for the ith ad impression, respectively. Our particular cases are handled as follows (refer to Figure 4):

- For header-won impressions (i.e., left-censoring), a_i is zero and b_i is the historical reserve price because the true failure reserve price t_i must be less than the observed reserve price b_i. For the ith impression, we want to maximize the probability that a reserve price t_i fails within the interval $[0, b_i]$, i.e., $P(0 \le t_i \le b_i)$.

- For reserve-won impressions (i.e., right-censoring), a_i is the historical revenue and b_i is positive infinity because the true failure reserve price t_i must be higher than the observed revenue a_i. For the ith impression, we want to maximize the probability that a reserve price t_i fails within the interval $[a_i, +\infty]$, i.e., $P(a_i \le t_i \le +\infty)$.

If a publisher's data have uncensored impressions (i.e., the highest bids $b_i^{(1)}$ are observed), we can use $P(t_i = b_i^{(1)}) = f(t)$.

Taking Weibull distribution as an example, its PDF is $f(t) = \frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta-1} e^{-\left(\frac{t}{\alpha}\right)^{\beta}}$, where α is the scale parameter and β is the shape parameter. Both α and β are positive. Typically, for parametric survival models, the shape parameter β is pre-specified and held fix. The scale parameter α can be re-parameterized in terms of feature predictors X_i and regression coefficients W_i:

$$\alpha_i = w_0 + \sum_{j=1}^{m} w_j x_{ij} \tag{2}$$

where m is the number of impression features reflecting the user, the page, and the context, which are defined next.

Features. The features we consider come from several aspects:

- **User**: User information plays an important role when advertisers/DSPs are assessing ad opportunities. Most publishers do not have access to personally identifiable information. Hence, we model users by 1) user IDs, 2) state-level location, 3) operating system and Internet browser, 4) network bandwidth, and 5) devices.
- **Ad Placement**: The ad placement sizes and positions determine if the ads are visible and thus convertible [22]. A small ad placement at the bottom of the page may not receive high adx bids. We model ad placements by 1) ad unit size, e.g., "123x324" and 2) ad position (On the publisher's page template, each ad slot has a unique name represents its position).
- **Page**: Page information reflects user interests and the information that a user is looking for at the moment. It may also impact impression valuation. We model pages by 1) page URLs, 2) channels, e.g., "business" and "lifestyle," 3) sections, i.e., sub-channels, and 4) the trending status of the page (i.e., if the page is labeled as trending by the publishers' editors).
- **Context**: 1) hour of the day and 2) referrer URLs, i.e., which page the current page request originated from.

All of these features are constant over the reserve prices. In other words, no matter what the reserve price is, the feature values of a given impression do not change. Since these features are all categorical, we convert them to dummy variables.

Inference. The log-likelihood function is defined as:

$$\ln L = \sum_{i=1}^{N} \left[y_i \ln \hat{y}_i + (1 - y_i) \ln(1 - \hat{y}_i) \right] \cdot r_i \tag{3}$$

where y_i is the ground truth of the ith impression. If the ith impression is a header-won impression, $y_i = 1$; otherwise, it is 0. $\hat{y}_i$ is the prediction of the reserve price failure rate. The corresponding success rate can be calculated by $1 - \hat{y}_i$. r_i is the reserve price for header-won impressions and to the revenue (i.e., second highest bid) for adx-won impressions.

Given Equation 1, $\hat{y}_i$ is calculated by $\hat{y}_i = \int_0^{t_i} f(t)dt$. As Figure 4 shows, if the ith impression is header-won, t_i is its reserve price. If it is reserve-won, t_i is its revenue. $f(t)$ is the PDF of a distribution. The scale parameter α_i in $f(t)$ is calculated by Equation 2.

An example weight r_i is assigned to each training impression i to represent how important the impression is. In our application, publishers care more about high-value impressions [27]. For example, correctly predicting that an impression values \$5 can bring more revenue to a publisher than correctly predicting that one values \$0.05. Thus, r_i serves as a weight in Equation 3.

Therefore, to learn $\mathbf{w}$, we can minimize the negative log-likelihood, where the log-likelihood function is as below:

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \left\{ -\sum_{i=1}^{N} \left[y_i \ln \hat{y}_i + (1 - y_i) \ln(1 - \hat{y}_i) \right] \cdot r_i \right\} \tag{4}$$

3.4 Pairwise Interaction Tensor Factorization

To improve the prediction performance, we further consider the interactions between users and pages. Indeed, the adx bids on an impression may be jointly determined by the user, the page, and the ad placement. Matrix factorization-based predictive models, e.g., Factorization Machines [21, 23], have been used at the advertisers' side to optimize RTB [12]. Considering pairwise interactions between features using latent vectors captures better the features' characteristics and thus boosts the performance of reserve price failure rate prediction. It can also overcome data sparsity, which is especially challenging for interactions between users and pages.

We add a term of pairwise interaction tensor factorization to Equation 2.

$$\alpha_i = w_0 + \sum_{j=1}^{m} w_j x_{ij} + \sum_{j=1}^{m} \sum_{h=j+1}^{m} \langle \mathbf{v}_j, \mathbf{v}_h \rangle x_{ij} x_{ih} \tag{5}$$

where m is the number of feature predictors. $\mathbf{v}_j$ describes the jth feature with k factors. k is a hyperparameter that defines the dimensionality of the factorization. $\langle \mathbf{v}_j, \mathbf{v}_h \rangle$ is the dot product of two vectors of size k, i.e., $\langle \mathbf{v}_j, \mathbf{v}_h \rangle := \sum_{f=1}^{k} v_{j,f} \cdot v_{h,f}$.

To alleviate the data sparsity and the cold-start problem [16], we assign occasional users and infrequent pages, whose occurrences are less than 5 times, to dummy features "rare_user" and "rare_page", respectively. In our dataset, 39.04% users and 74.94% pages are involved in at least 5 impressions.

Adding factorization does not change substantially Equation 3 for log-likelihood. The only part the needs to be revised is how to compute the scale parameter α in the distribution PDF $f(t)$, i.e., replace Equation 2 with Equation 5.

Adding factorization increases the number of parameters that need to be learned from data, especially if k is large. Thus, we add an L2 penalty term, i.e., $\lambda_1 \|\mathbf{w}\|^2$ and $\lambda_2 \|\mathbf{v}\|^2$, at the end of Equation 4 to avoid overfitting [21]. This limits $\mathbf{w}$ and $\mathbf{v}$ from becoming extremely large or small. λ_1 and λ_2 are pre-specified parameters controlling the strength of regularization. In this case, the new loss function is shown in Equation 6.

$$\mathbf{w}^* = \underset{\mathbf{w},\mathbf{v}}{\operatorname{argmin}} \left\{ -\sum_{i=1}^{N} [y_i \ln \hat{y}_i + (1 - y_i) \ln(1 - \hat{y}_i)] \cdot r_i \right. \\ \left. + \lambda_1 \|\mathbf{w}\|^2 + \lambda_2 \|\mathbf{v}\|^2 \right\} \tag{6}$$

3.5 Header Bidding Regularization

Before undergoing a traditional RTB auction, an ad impression is offered for sale in a header auction. In fact, the same advertiser may join both auctions. Therefore, it can be assumed that header bids are pseudo-randomly sampled from adx bids. The header bids that publishers receive from header bidding partners are the maximums in the corresponding sample groups. Since header bidding uses first-price auctions, the highest header bid in each header bidding partner is known, which can shed insights on advertiser's bids in the later RTB auctions.

This section discusses how to leverage the known bids in header bidding to improve reserve price failure rate prediction. One option is to add the winning header bids as feature vectors in the prediction model. However, in our dataset, the publisher receives winning header bids from five header bidding partners, who may not provide bids for every impression; this often happens due to network latency that prevents publishers from receiving header bids on time. Due to the presence of a substantial number of missing values, directly adding these bids into feature vectors may not work.

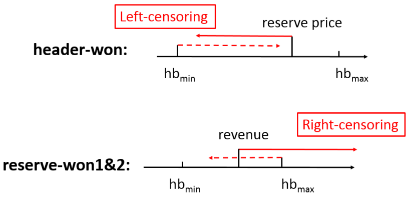


Figure 5: Two Cases for Header Bidding Regularization

Instead, we propose to use header bidding information to regularize model learning. The intuition is shown in Figure 5. The minimum and the maximum header bids of an impression are first identified. They are denoted as hb_{min} and hb_{max}, respectively. For a header-won impression, its reserve price was set too high. The advertisers in the RTB auction were willing to pay less than the reserve price set by the publisher (i.e., left-censoring), while the advertisers in header bidding auctions were willing to pay at least hb_{min}. If hb_{min} is less than the reserve price, it can be assumed that likely advertisers in the RTB auction were also willing to pay more than hb_{min}. Thus, as Figure 5 shows, the header-won impression is strictly left-censored at the reserve price, while the highest adx bid probably is more than hb_{min} (i.e., loosely right-censored at hb_{min}). For a reserve-won impression, its reserve price was set too low. The advertisers in the RTB auction were willing to pay more than the revenue received by the publisher (i.e., right-censoring), while the advertisers in header bidding auctions were willing to pay at most hb_{max}. If hb_{max} is less than the revenue, it can be assumed that likely advertisers in the RTB auction were willing to pay less than hb_{max}. Thus, as Figure 5 shows, the reserve-won impression is strictly right-censored at the revenue, while it is loosely left-censored at hb_{max}.

Thus, we add a term on header bidding regularization to the log-likelihood, i.e., from Equation 3 to Equation 7:

$$\begin{aligned} \ln L = & \sum_{i=1}^{N} [y_i \ln \hat{y}_i + (1 - y_i) \ln(1 - \hat{y}_i)] \cdot r_i \\ & + \lambda_3 \cdot I_{\{A_i \in H\}} \cdot \left[0 \cdot \ln \hat{y}_i^{min} + 1 \cdot \ln(1 - \hat{y}_i^{min})\right] \cdot r_i \\ & + \lambda_4 \cdot I_{\{A_i \in E\}} \cdot \left[1 \cdot \ln \hat{y}_i^{max} + 0 \cdot \ln(1 - \hat{y}_i^{max})\right] \cdot r_i \\ = & \sum_{i=1}^{N} [y_i \ln \hat{y}_i + (1 - y_i) \ln(1 - \hat{y}_i)] \cdot r_i \\ & + \lambda_3 \cdot I_{\{A_i \in H\}} \cdot \ln(1 - \hat{y}_i^{min}) \cdot r_i + \lambda_4 \cdot I_{\{A_i \in E\}} \cdot \ln \hat{y}_i^{max} \cdot r_i \end{aligned} \tag{7}$$

In Equation 7, the first term is the same as in Equation 3. $I_{\{A_i \in H\}}$ is an indicator variable, whose value is 1 if the impression A_i is in the header-won impressions H; otherwise, it is 0. $\hat{y}_i^{min}$ is the prediction at the minimum header bid of impression A_i (as shown in the first case of Figure 5). It is calculated by $\hat{y}_i^{min} = \int_0^{hb_{min}} f(t)dt$.

Since we assume adx advertisers are likely to bid more than hb_{min} (i.e., survive at hb_{min}), the closer $\hat{y}_i^{min}$ is to 0, the better. λ_3 is the strength of the regularization for header-won impressions. A large λ_3 encourages the model to predict correctly at hb_{min}. $I_{\{A_i \in E\}}$ is an indicator variable, whose value is 1 if A_i is in the reserve-won impressions E; otherwise, it is 1. $\hat{y}_i^{max}$ is the prediction at the maximum header bid of A_i (as shown in the second case of Figure 5). It is calculated by $\hat{y}_i^{max} = \int_0^{hb_{max}} f(t)dt$.

Since we assume adx advertisers likely pay less than hb_{max}, the closer $\hat{y}_i^{max}$ is to 1, the better. They are all weighted by r_i. λ_4 is the strength of the regularization for reserve-won impressions. A large λ_4 encourages the model to predict correctly at hb_{max}.

The loss function is revised from Equation 6 to Equation 8.

$$\begin{aligned} \mathbf{w}^* = \underset{\mathbf{w},\mathbf{v}}{\operatorname{argmin}} - & \left\{ \sum_{i=1}^{N} [y_i \ln \hat{y}_i + (1 - y_i) \ln(1 - \hat{y}_i)] \cdot r_i \right. \\ & + \lambda_3 \cdot I_{\{A_i \in H\}} \cdot \ln(1 - \hat{y}_i^{min}) \cdot r_i + \lambda_4 \cdot I_{\{A_i \in E\}} \cdot \ln \hat{y}_i^{max} \cdot r_i \\ & \left. + \lambda_1 \|\mathbf{w}\|^2 + \lambda_2 \|\mathbf{v}\|^2 \right\} \end{aligned} \tag{8}$$

4 EVALUATION

4.1 Experimental Dataset and Ground Truth

The dataset has been described in Section 3.1. The dataset is collected over one day in April 2018 on Forbes.com. All header-won and reserve-won impressions are shuffled. Training data, validation data, and test data are randomly picked by 8:1:1. Nearly 13 million impressions are in the training data, and 1.6 million impressions are in the validation/test data.

Test impressions are also weighted by r_i, as it was already described for training data in Section 3.3. r_i is set to the reserve price if the ith impression is header-won or the revenue (second highest bid) if reserve-won.

We consider reserve price failure rate prediction as a classification problem. Our model outputs the probability that no advertisers will outbid a reserve price for an impression. The *ground truth* is available in our dataset; it is known for each ad impression with a reserve price whether it is header-won or reserve-won.

4.2 Implementation

The proposed parametric survival model is implemented using Tensorflow. The experiments are run on a desktop with i7 3.60Hz CPU and 32GB RAM. The computation is sped up using NVIDIA GeForce GTX 1060 6G GPU. Running 5 epochs usually takes 5-6 hours depending on the parameter setting. In practice, the training phase can be offline. The prediction of an impression is done in less than 100ms, which demonstrates that the prediction phase can be deployed as an online process.

The training goal is to minimize the log-loss. Since the large training dataset does fit the memory, the optimizer we adopt is Stochastic Gradient Descent (SGD) with a learning rate of 10^{-3}.

Considering the training speed and memory consumption, we set the training batch size to 2048. Although using smaller batch sizes, in theory, can speed up convergence, it may also lead to longer training time for one epoch due to more I/O with the GPU. In this experiment, we find a batch size of 2048 is a good trade-off between convergence and training time for one epoch. The training process usually can converge at the second epoch.

To avoid overfitting, across all 10 epochs, the model that performs the best on the validation data is applied to the test data.

The parameter values are empirically set: λ_1 and λ_2 (introduced in Section 3.4) are both set to 10^{-7}. They control the complexity of the feature weights, $\mathbf{w}$, and the complexity of the weights for feature latent vectors, $\mathbf{v}$. λ_4 (introduced in Section 3.5) and λ_5 are set to 10^{-4}. They control the strength of header bidding regularization for header-won and reserve-won impressions, respectively.

4.3 Experimental Metrics

Accuracy computes the percentage of the test instances correctly predicted at the reserve prices (for header-won impressions) or the revenues (for reserve-won impressions). Higher values are better.

Log-Loss is widely used in probabilistic classification. It penalizes a method more for being both confident and wrong. $logloss = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$, where N is the number of test impressions. $\hat{y}_i$ is the probabilistic prediction and y_i is the ground truth (either 0 or 1). Lower values are better.

Concordance-Index (C-Index) [10] is a well-recognized measure of discrimination for models that predicts a time-to-event and equals the proportion of impression pairs in which the predicted event probability is higher for the subject who experienced the event of interest than that of the subject who did not.

The original C-Index requires uncensored instances. However, our dataset contains censored instances: either left- or right-censored. Figure 6 shows the four relationships of any pair of impressions, i.e., $< A_i, A_j >$. In the first case, A_i is left-censored (i.e., header-won), while A_j is right-censored (i.e., reserve-won). As it is known that A_i's reserve price t_i failed and A_j's reserve price t_j was outbid, we can expect that $1 \geqslant \hat{y}_i > \hat{y}_j \geqslant 0$, where $\hat{y}_i$ is the reserve price failure rate of A_i. The failure rates of $< A_i, A_j >$ are comparable. In the second case, both are left-censored because their reserve prices are higher than the (unobserved) highest adx bid prices. The highest bid of A_i, $t_i{}'$, is within $[0, t_i)$. The highest bid of A_j, $t_j{}'$, is within $\left[0, t_j\right)$. In the case of $t_2{}' < t_1{}' < t_1 < t_2$, we may expect $0 \leqslant \hat{y}_i < \hat{y}_j \leqslant 1$ because $t_2 - t_2{}' > t_1 - t_1{}'$. In other words, $\hat{y}_i$ is farther from its highest bid price than $\hat{y}_j$. Thus, $\hat{y}_j$ should be closer to 1. However, in the case of $t_1{}' \ll t_1 < t_2{}' < t_2$, we expect $\hat{y}_i > \hat{y}_j$ because $t_2 - t_2 < t_1 - t_1{}'$. Since $t_1{}'$ and $t_2{}'$ are censored, The failure rates of $< A_i, A_j >$ is not comparable. Likewise, the third and the fourth cases are also not comparable on failure rates.

Thus, we create a customized C-Index based on the original one:

$$c\ index = \frac{1}{|\varepsilon|}\sum_{A_i \in L}\sum_{A_j \in R} I_{\{\hat{y}_i > \hat{y}_j\}} \tag{9}$$

where L and R are the set of left- and right-censored impressions, respectively. The customized C-Index only considers the first case, which is the only comparable one. Higher values are better.

This customized C-Index provide a measure that considers only the test instance pairs whose relationships are known. Along with log-loss, it is complementary for the data in which the highest bids are far away from the observed reserve prices/revenues, since the observed reserve prices/revenues are no longer reliable. Since the customized C-Index only considers one case (refer to Figure 6), it may be hacked by a naive method that always outputs low failure rates for large reserve prices and large failure rates for low reserve prices. Therefore, in the experiments, we evaluate the model using C-index along with log-loss and accuracy.

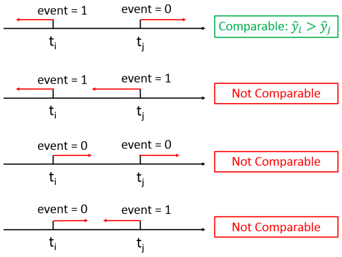


Figure 6: Pairwise relationships

In our experiments, we ran each test three times and reported the averages.

4.4 Comparison Systems

Observed Reserve Price/Revenues (OR). The simplest way to predict failure rates is to use reserve prices as the only feature in the model. Thus, we build a logistic regression with one feature: reserve prices for header-won and revenues for reserve-won.

Kaplan-Meier (KM): Kaplan-Meier [13] is an extensively-used non-parametric statistic used to estimate the survival function from lifetime data. We slightly modify it: $\hat{y}_i = 1 - \prod_{j:t_j \leq t_i}\left(1 - \frac{d_j}{n_j}\right)$, where t_i is the reserve price/revenue of the ith test instance. t_j is a price less than t_i. d_j is the number of impressions that failed to be sold at t_j. n_j is the number of impressions that did not fail at t_j.

4.5 Comparison of Different Distributions

The proposed parametric survival model requires the assumption of a distribution of the reserve price failure rate. The distribution type can impact the prediction performance. In practice, publishers can plug in commonly-used distributions for survival analysis and pick the one with the best performance [13]. We evaluate the performance of Exponential, Weibull, and Log-Logistic distributions. Section 3.3 described how Weibull distribution is used. The PDFs of Exponential and Log-Logistic distributions are $f(t) = \alpha e^{-\alpha t}$ and

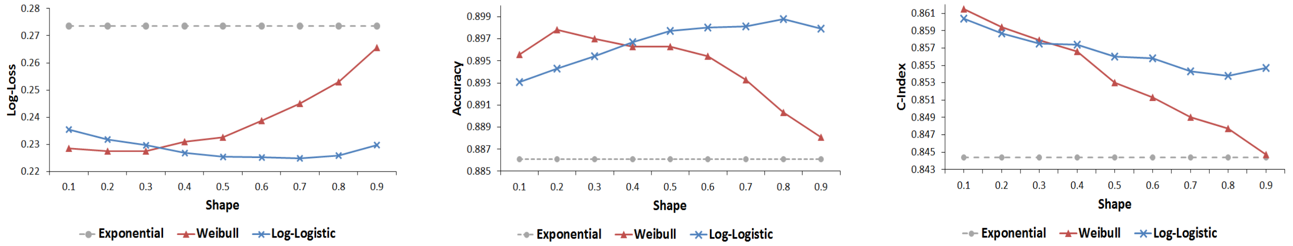


Figure 7: Comparison of Different Distributions for Our Model

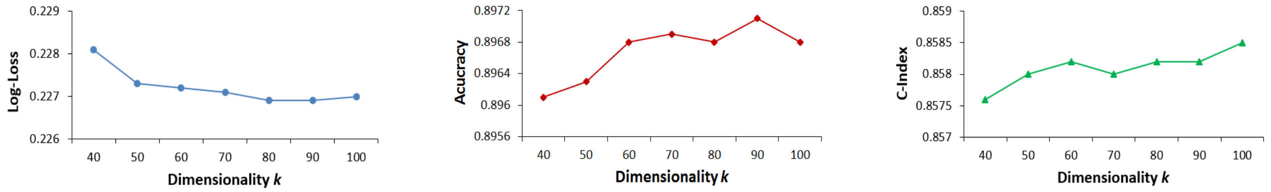


Figure 8: Comparison of Different ks

$f(t) = \left[\frac{\beta}{\alpha}\cdot(\frac{t}{\alpha})^{\beta-1}\right] / \left[(1+(\frac{t}{\alpha})^{\beta})^{2}\right]$, respectively. The scale parameter, α, can be learned from the training data. It is re-parameterized in terms of feature predictors X_i and regression coefficients W_i (Equation 2). The optimal shape β, is found through experiments.

The performance of different distributions and different shape parameters is presented in Figure 7. The performance is reported for the proposed model with factorization and header bidding regularization. As the Exponential distribution has no shape parameter, its performance does not change across different βs. As one can see, the performance of the Exponential distribution is not as good as Weibull and Log-Logistic.

The Log-Logistic obtains its lowest log-loss (0.2248) when $\beta = 0.7$ and the highest accuracy (0.8988) when $\beta = 0.8$. However, $\beta = 0.7$ and 0.8 make the model with the Log-Logistic misclassify more test instances of the "sure case" (case 1 in Figure 6). $\beta = 0.1$ leads to the highest C-Index, while its log-loss and accuracy are unsatisfactory: $\beta = 0.1$ tends to favor impressions with large reserve prices rather than those with low reserve prices (in terms of lowest failure rates).

The Weibull distribution obtains the lowest log-loss (0.2274) when $\beta = 0.3$, the highest accuracy (0.8978) when $\beta = 0.2$, and the highest C-Index (0.8615) when $\beta = 0.1$. The Weibull parametric survival model with a small β (e.g., 0.2) can obtain outstanding performance for all three metrics. Thus, in the rest of the experiments, we use the Weibull distribution with $\beta = 0.2$ as the model setting.

4.6 Comparison of Different Dimensionalities

In Section 3.4, each feature is represented by a k-length latent vector that carries characteristics of the feature. k is an important parameter which can significantly impact model performance. In theory, a larger k has better representation and results in a more complex model; however, it has a higher chance to overfit. Dimentionality k is usually determined through experiments.

The results are shown in Figure 8. They indicate that, with the increasing in vector dimensionality, the performance on the test data generally improves. The reason is that longer latent feature vectors can better capture the signals in the training data so as to improve model complexity. In addition, the performance of log-loss and accuracy increases fast when k grows from 40 to 60. After 60, the growth slows down and even reduces. This is because most signals in the training data have been captured, and further increasing k results in overfitting. C-Index also follows such a trend. In addition, we observe that $k = 100$ leads to a jump when compared with $k = 90$. The possible reason is that $k = 100$ may happen to correctly predict many test instances in the first case of Figure 6, which increases the customized C-Index. Finally, we notice that there is no single k that results in best performance for all three metrics: When $k = 80$, we get the lowest log-loss in the test data. $k = 90$ leads to the highest accuracy. $k = 100$ wins the C-Index test.

Publishers can select the k based on their objectives: Log-loss can make balance between ad revenue and failure risk. Accuracy maximize the total number of impressions that are correctly classified. C-index can be used when the observed reserve prices and revenues are believed to be very off from the actual highest bids.

Since minimizing the overall log-loss is the training objective in Equation 8, we set $k = 80$ in the rest of the experiments.

4.7 Overall Comparison

We compare the proposed model with two baselines: *KM* and *OR*. For our model, we use three versions: the plain parametric survival model *param-surv*, the model with interaction factorization *param-surv-f*, and the model with both pairwise interaction factorization and header bidding regularization *param-surv-fhb*. All use Weibull distribution with $\beta = 0.2$.

Table 1 presents the results. All three versions of our model outperform the two baselines. Among the three version, *param-surv-fhb* is the best.

The C-Index values of the two baselines are all zero because: 1) KM makes predictions based on the percentages of impressions whose reserve prices that are less than a given price have already failed. As the overall failure rate increases with the increase of the

	Log-Loss	Accuracy	C-Index
KM	1.5762	0.5495	0.0
OR	0.6883	0.5486	0.0
Param-surv	0.2425	0.8880	0.8532
Param-surv-f	0.2305	0.8953	0.8577
Param-surv-fhb	0.2266	0.8972	0.8583

Table 1: Comparison for all impressions

	Log-Loss	Accuracy	C-Index
KM	1.5766	0.5493	0.0
OR	0.6881	0.5487	0.0
Param-surv	0.2438	0.8879	0.8533
Param-surv-f	0.2311	0.8946	0.8573
Param-surv-fhb	0.2186	0.9011	0.8597

Table 2: Comparison for impressions with header bids only

reserve price, KM always “thinks” an impression with a higher reserve price has a higher failure rate (i.e., case 1 in Figure 6 never happens). 2) Likewise, OR is a linear model. It learns from the data that the failure rate is positively correlated with the reserve price. It always gives a higher reserve price a greater failure rate than a lower reserve price.

The *param-surv* model has good performance and clearly outperform the baselines. Adding interaction factorization, *param-surv-f* reduces the log-loss by 5% because latent feature vectors can better capture the regularities in the training data and overcome data sparsity compared to one-hot encoding. Furthermore, utilizing header bidding to regularize the model, *param-surv-fhb* can reduce the log-loss by additional 1.7% from *param-surv-f*.

Header bidding regularization is only applicable on impressions with header bids. To fully present its effect, we filter out impression without header bids from the test set. The results are shown in Table 2, and they demonstrate that *Param-surv-fhb* has a larger performance improvement compared with the other models.

5 CONCLUSIONS

This paper proposes a parametric survival model to predict the failure rate of the reserve price of an online display ad impression in an ad exchange auction. The model is further augmented by user-page pairwise interaction tensor factorization and header bidding factorization. We also develop a customized C-Index for datasets containing only left- and right-censored instances. The experimental results show that the proposed models with the Weibull distribution significantly outperforms a Kaplan-Meier model and a logistic regression with observed reserve price/revenue as the feature. Adding factorized interaction and header bidding regularization further boost performance. Our model can be adopted by the majority of online publishers because similar data can be conveniently collected on most publishers’ platforms.

ACKNOWLEDGMENT

This research was supported in part by the National Science Foundation (NSF) under Grant No. DGE 1565478 and a grant from the Leir Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] 2018. Google DoubleClick for Publishers. https://www.google.com/dfp/.

[2] 2018. OpenX Ad Server for Publishers. https://community.openx.com/s/article/Ad-Server-Report/.

[3] Miguel Angel Alcobendas Lisbona, Sheide Chammas, and Kuang-chih Lee. 2016. Optimal reserve prices in upstream auctions: Empirical application on online video advertising. In *KDD’16*. ACM, 1395–1404.

[4] Daniel Austin, Sam Seljan, Julius Monello, and Stephanie Tzeng. 2016. Reserve price optimization at scale. In *2016 IEEE 3rd International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 528–536.

[5] Nicolo Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. 2015. Regret minimization for reserve prices in second-price auctions. *IEEE Transactions on Information Theory* 61, 1 (2015), 549–564.

[6] Pedro Chahuara, Nicolas Grislain, Gregoire Jauvion, and Jean-Michel Renders. 2017. Real-Time Optimization of Web Publisher RTB Revenues. In *KDD’17*. ACM, 1743–1751.

[7] Ying Cui, Ruofei Zhang, Wei Li, and Jianchang Mao. 2011. Bid landscape forecasting in online ad exchange marketplace. In *KDD’11*. ACM, 265–273.

[8] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. 2007. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American economic review* 97, 1 (2007), 242–259.

[9] eMarketer. 2016. US Digital Display Ad Spending to Surpass Search Ad Spending in 2016. https://www.emarketer.com/Article/US-Digital-Display-Ad-Spending-Surpass-Search-Ad-Spending-2016/1013442. Accessed: 2019-01-23.

[10] Frank E Harrell. 2001. Regression modeling strategies, with applications to linear models, survival analysis and logistic regression. *Springer* (2001).

[11] Grégoire Jauvion, Nicolas Grislain, Pascal Dkengne Sielenou, Aurélien Garivier, and Sébastien Gerchinovitz. 2018. Optimization of a SSP’s Header Bidding Strategy using Thompson Sampling. In *KDD’18*. ACM, 425–432.

[12] Yuchin Juan, Damien Lefortier, and Olivier Chapelle. 2017. Field-aware factorization machines in a real-world online advertising system. In *WWW’17*. International World Wide Web Conferences Steering Committee, 680–688.

[13] David G Kleinbaum and Mitchel Klein. 2012. Introduction to survival analysis. In *Survival Analysis*. Springer, 1–54.

[14] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.

[15] Juanjuan Li, Xiaochun Ni, Yong Yuan, Rui Qin, Xiao Wang, and Fei-Yue Wang. 2017. The impact of reserve price on publisher revenue in real-time bidding advertising markets. In *IEEE SMC’17*. IEEE, 1256–1261.

[16] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. 2014. Facing the cold start problem in recommender systems. *Expert Systems with Applications* 41, 4 (2014), 2065–2073.

[17] Brendan Lucier, Renato Paes Leme, and Eva Tardos. 2012. On revenue in the generalized second price auction. In *WWW’12*. ACM, 361–370.

[18] Andres M Medina and Mehryar Mohri. 2014. Learning theory and algorithms for revenue optimization in second price auctions with reserve. In *ICML’14*. 262–270.

[19] Andres Munoz and Sergei Vassilvitskii. 2017. Revenue Optimization with Approximate Bid Predictions. In *NIPS’17*. 1858–1866.

[20] Michael Ostrovsky and Michael Schwarz. 2011. Reserve prices in internet advertising auctions: A field experiment. In *ACM EC’11*. ACM, 59–60.

[21] Steffen Rendle. 2012. Factorization machines with libfm. *TIST* 3, 3 (2012), 57.

[22] Chong Wang, Achir Kalra, Li Zhou, Cristian Borcea, and Yi Chen. 2017. Probabilistic models for ad viewability prediction on the web. *TKDE* 29, 9 (2017), 2012–2025.

[23] Chong Wang, Shuai Zhao, Achir Kalra, Cristian Borcea, and Yi Chen. 2018. Predictive models and analysis for webpage depth-level dwell time. *JASIST* (2018).

[24] Ping Wang, Yan Li, and Chandan K. Reddy. 2017. Machine Learning for Survival Analysis. In *ACM Computing Surveys*, Vol. 1. ACM, 1–38.

[25] Wush Wu, Mi-Yen Yeh, and Ming-Syan Chen. 2018. Deep Censored Learning of the Winning Price in the Real Time Bidding. In *KDD’18*. ACM, 2526–2535.

[26] Wush Chi-Hsuan Wu, Mi-Yen Yeh, and Ming-Syan Chen. 2015. Predicting winning price in real time bidding with censored data. In *KDD’15*. ACM, 1305–1314.

[27] Zhihui Xie, Kuang-Chih Lee, and Liang Wang. 2017. Optimal Reserve Price for Online Ads Trading Based on Inventory Identification. In *ADKDD’17*. ACM, 6.

[28] Shuai Yuan, Jun Wang, Bowei Chen, Peter Mason, and Sam Seljan. 2014. An empirical study of reserve price optimisation in real-time bidding. In *KDD’14*. ACM, 1897–1906.

[29] Wen-Yuan Zhu, Wen-Yueh Shih, Ying-Hsuan Lee, Wen-Chih Peng, and Jiun-Long Huang. 2017. A gamma-based regression for winning price estimation in real-time bidding advertising. In *IEEE Big Data’17*. IEEE.