Introduction to Survival Analysis

Kan Ren

Apex Data and Knowledge Management Lab
Shanghai Jiao Tong University

Seminar Tutorial at Apex Lab

Outline

Outline

Probability

Probability Density Function (P.D.F.):

$$p_t(t) = Pr(T = t) \ . \tag{1}$$

Cumulative distribution function (C.D.F.):

$$w_t(t) = Pr(T < t) = \int_0^t p_t(v)dv \ . \tag{2}$$

Outline

Censored Data

Right Censored Data

The event happens after the observation time.

- E: Event; t_{obsv}: The observe time;
- $\{(\boldsymbol{x}, t_{obsv}, e = \text{True}/\text{False})\}$;
- $\{(\boldsymbol{x}, T_E)\}$, T_E is the event happening log.

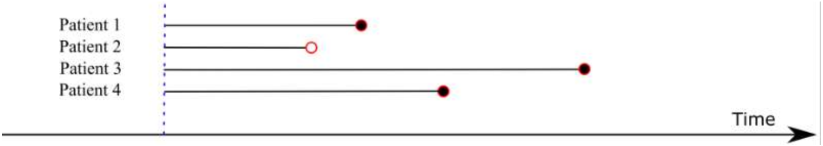
Example

- Patient's survival time.
- The true winning price of a bidding auction.
- The next visit time of the user.

Outline

1. Background
   - Probability
   - Censored Data
   - Challenges
2. Methodology
   - Non-parametric Models
     - Kaplan Meier Estimator
     - Survival Tree
   - Parametric Model
     - Cox Hazard Proportional Model
     - Deep Survival Analysis
3. Evaluation

Challenges

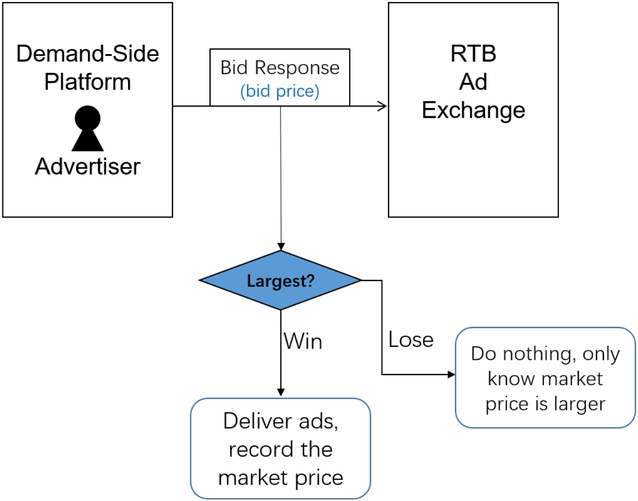


Right Censorship

- Partially data usage: discard large data for learning.
- Right Censorship: only know that the event happening time is greater than the observing time window.
- Evaluation: proper evaluation metric is needed.

Modeling Right Censored Data in Display Ads

Losing and Winning in 2nd-price Auction

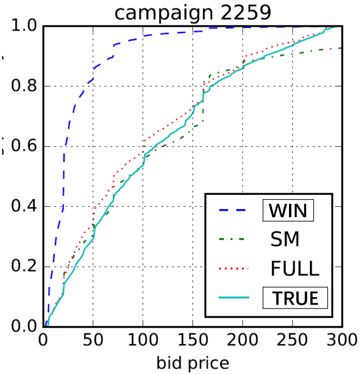


Modeling Right Censored Data

Right Censored

Right Censorship

As in 2^{nd} price auction, if you *lose*, you only know that the *market price* is higher than your bidding price, which result in *right censorship*.



Outline

Kaplan Meier Estimator

Preliminaries

- $S(t) = Pr(t < T_E)$: Survival rate
- $F(t) = 1 - S(t)$: Failing rate.

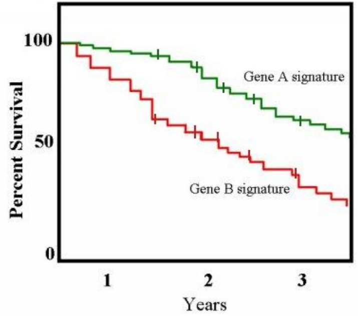
Algorithm

The estimator for an individual is given by

$$S(t) = \prod_{i:t_i \leq t} \left(1 - \frac{d_i}{n_i}\right) , \tag{3}$$

where d_i is the number of events and n_i is the total individuals at risk at time i.

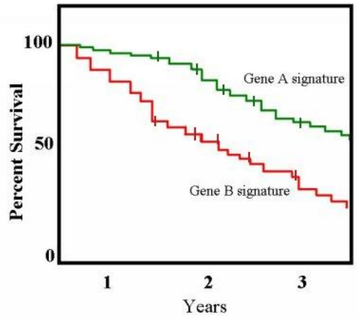
Survival Tree with Kaplan Meier Methods



Cons of KM

- Corse grained, the same for all individuals.
- Statistcal method, cannot apply personalized forecasting.

Survival Tree with Kaplan Meier Methods



Cons of KM

- Corse grained, the same for all individuals.
- Statistcal method, cannot apply personalized forecasting.

Question

How to apply an appropriate *clustering* method for one individual?

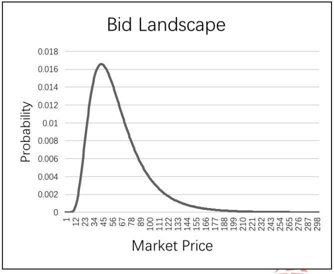
Tree-based Mapping

Goal

Given the auction feature $\boldsymbol{x}$, forecast the market price distribution $p_{\boldsymbol{x}}(z)$[a].

[a] Yuchen Wang, Kan Ren, Weinan Zhang, Yong Yu. Functional Bid Landscape Forecasting for Display Advertising. ECML-PKDD, 2016.

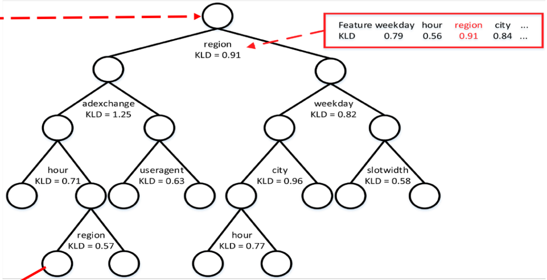
- Date: 20160320
- Hour: 14
- Weekday: 7
- IP: 119.163.222.*
- Region: England
- City: London
- Country: UK
- Ad Exchange: Google
- Domain: yahoo.co.uk
- URL: http://www.yahoo.co.uk/abc/xyz.html
- OS: Windows
- Browser: Chrome
- Ad size: 300*250
- Ad ID: a1890
- User tags: Sports, Electronics

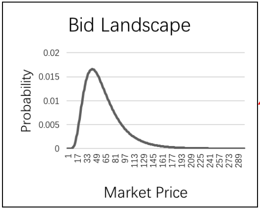


Tree-based Mapping

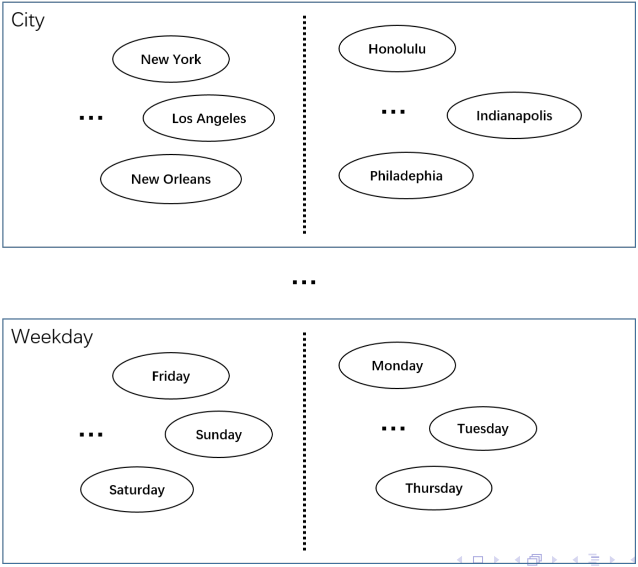
Methodology

- Date: 20160320
- Hour: 14
- Weekday: 7
- IP: 119.163.222.*
- Region: England
- City: London
- Country: UK
- Ad Exchange: Google
- Domain: yahoo.co.uk
- URL: http://www.yahoo.co.uk/abc/xyz.html
- OS: Windows
- Browser: Chrome
- Ad size: 300*250
- Ad ID: a1890
- User tags: Sports, Electronics





Node Splitting



Node Splitting

KLD and Clustering

Kullback-Leibler Divergence (KLD)

A measure of the difference between two probability distributions P and Q.

Node Splitting
KLD and Clustering

Kullback-Leibler Divergence (KLD)

A measure of the difference between two probability distributions P and Q.

Node Splitting (one step)

Divide all the category (including in this node) values into two sets, maximizing KLD between the resulted two sets.

Node Splitting

KLD and Clustering

Kullback-Leibler Divergence (KLD)

A measure of the difference between two probability distributions P and Q.

Node Splitting (one step)

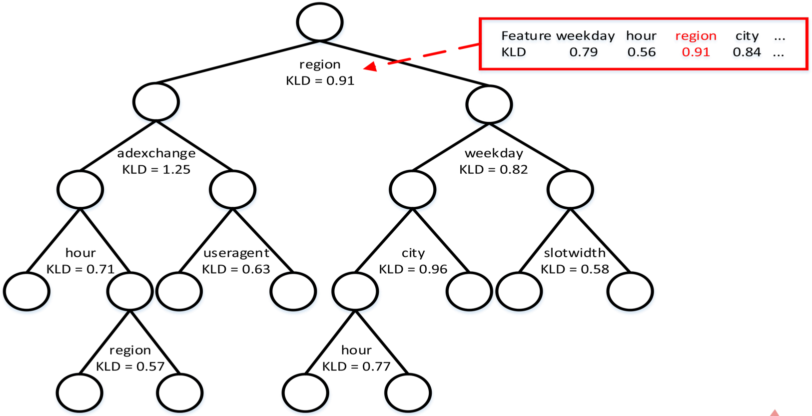
Divide all the category (including in this node) values into two sets, maximizing KLD between the resulted two sets.

Algorithm

Using K-Means Clustering according to KLD values.

Node Splitting

KLD and Clustering



Handling Censorship

Survival Model

- For winning auctions: We have the true market price value.
- For lost auctions: We only know our proposed bid price and know that the true market price is higher than that.

Intuition

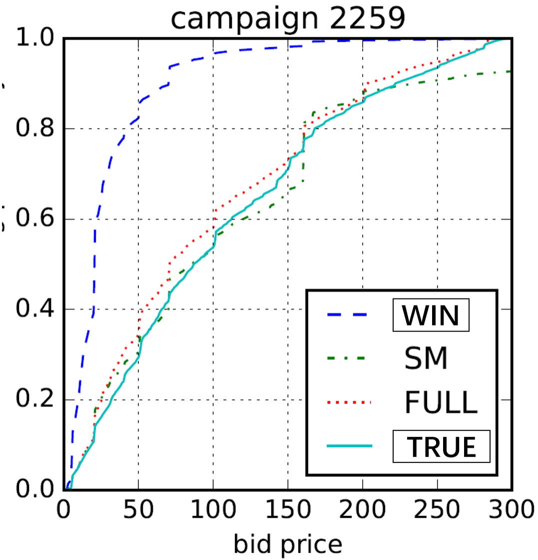
Most related works focus only on the winning auctions without considering the lost auction, which contains the information to infer the true distribution.

$$(b_i, w_i, m_i)_{i=1,2,\cdots,M} \quad \longrightarrow \quad (b_j, d_j, n_j)_{j=1,2,\cdots,N}$$

$b_j < b_{j+1}$, d_j is number of winning auctions by $b_j - 1$, n_j is number of lost auctions by $b_j - 1$. So

$$w(b_x) = 1 - \prod_{b_j < b_x} \frac{n_j - d_j}{n_j}, \quad p(z) = w(z+1) - w(z). \tag{4}$$

Survival Model



Outline

Cox Hazard Proportional Model

Hazard Rate The rate of the event happening given *not happened* before.

Hazard Function The function $\lambda(t|\boldsymbol{x})$ to predict the hazard rate w.r.t. the covariate input $\boldsymbol{x}$.

Hazard Proportional Model The hazard function which models with the proportional relationship with the input covariate, where $\lambda(t|\boldsymbol{x}) = \lambda_0(t)\exp(h(\boldsymbol{x}))$.

Example

Linear Cox Hazard Model: $h(\boldsymbol{x}) = \boldsymbol{\beta x}$.
Question: What if $h(\boldsymbol{x})$ is non-linear?

Discussion

Relationship among hazard rate λ, P.D.F. function $p(z)$, C.D.F. function $S(b)$

$$
\begin{aligned}
\lambda(b) &= \lim_{db \to 0} \frac{Pr(b \le z \le b + db | z > b)}{db} \\
&= \lim_{db \to 0} \frac{Pr(b \le z \le b + db)/Pr(z > b)}{db} \\
&= \lim_{db \to 0} \frac{(w_z(b + db) - w_z(b))/S(b)}{db} = \frac{p_z(b)}{S(b)} = -\frac{S'(b)}{S(b)} \; .
\end{aligned} \tag{5}
$$

$$
\begin{aligned}
p_t(t|\boldsymbol{x}) &= \frac{\partial w_t(t|\boldsymbol{x})}{\partial t} = \frac{-\partial S(t|\boldsymbol{x})}{\partial t} \\
&= \frac{\partial \exp(\int_0^t \lambda(v|\boldsymbol{x}) dv)}{\partial t} \\
&= \exp\left(\int_0^t \lambda(v|\boldsymbol{x}) dv\right) \lambda(t|\boldsymbol{x}) \; .
\end{aligned} \tag{6}
$$

Cost Function: Partial Likelihood

$$
\begin{aligned}
\text{Likelihood}_i &= \frac{\lambda(t_i|\boldsymbol{x}_i)}{\sum_{j:t_j>t_i} \lambda(t_i|\boldsymbol{x}_j)} \\
&= \frac{\lambda_0(t_i) e^{h(\boldsymbol{x}_i)}}{\sum_{j:t_j>t_i} \lambda_0(t_i) e^{h(\boldsymbol{x}_j)}} \\
&= \frac{e^{h(\boldsymbol{x}_i)}}{\sum_{j:t_j>t_i} e^{h(\boldsymbol{x}_j)}} \ .
\end{aligned} \tag{7}
$$

$$
\begin{aligned}
\mathcal{L}_{\text{PL}} &= -\log \prod_{i:(\boldsymbol{x}_i, t_i)} \text{Likelihood}_i \\
&= -\sum_{i:(\boldsymbol{x}_i, t_i)} \left(h(\boldsymbol{x}_i) - log \sum_{j:t_j>t_i} e^{h(\boldsymbol{x}_j)} \right)
\end{aligned} \tag{8}
$$

Base Hazard Function

Example

Weibull Distribution: $\lambda_0(t) = \frac{k}{\eta}\left(\frac{t}{\eta}\right)^{k-1} \cdot e^{-(t/\eta)^k}$.

Question: formulation assumption; without considering $\boldsymbol{x}$.

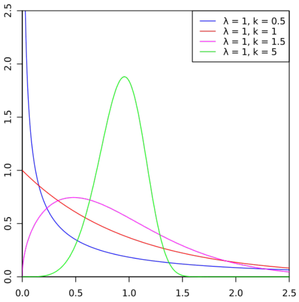


Figure: Probability Density Function

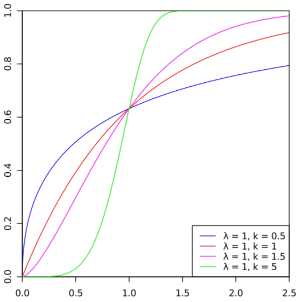


Figure: Cumulative Distribution Function

Deep Survival Analysis

NN-based Cox Model

Using deep neural network to model $h(\boldsymbol{x})$.[a] [b] [c]

[a] Faraggi D, Simon R. A neural network model for survival data[J]. Statistics in medicine, 1995.

[b] Ranganath R, Perotte A, Elhadad N, et al. Deep Survival Analysis[C]//Machine Learning for Healthcare Conference. 2016.

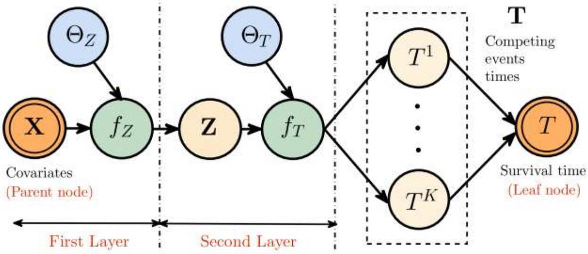
[c] Luck M, Sylvain T, Cardinal H, et al. Deep Learning for Patient-Specific Kidney Graft Survival Analysis[J]. arXiv preprint arXiv:1705.10245, 2017.

Deep Survival Analysis

Generative NN-based Survival Time Estimation[a]

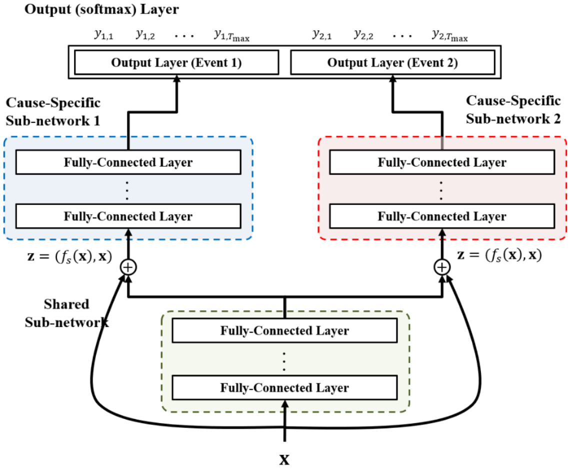
[a] Deep Multi-task Gaussian Processes for Survival Analysis with Competing Risks, NIPS 2017

$$
\begin{aligned}
& f_Z \sim \mathcal{GP}(0, \mathbb{K}_{\Theta_Z}), \quad f_T \sim \mathcal{GP}(0, \mathbb{K}_{\Theta_T}) \\
& \mathbb{Z}_i \sim \mathcal{N}(f_Z(\mathbb{X}_i), \sigma_Z^2 \mathbb{I}), \quad \mathbb{T}_i \sim \mathcal{N}(f_T(\mathbb{X}_i), \sigma_T^2 \mathbb{I}) \\
& T_i = min(T_i^1, \ldots, T_i^K) \ .
\end{aligned}
\tag{9}
$$



Deep Survival Analysis

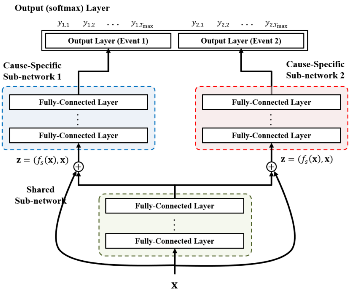
DeepHit (Lee et al. AAAI 2018)



DeepHit: A Deep Learning Approach to Survival Analysis with Competing Risks. Lee et al. AAAI 2018.

Deep Survival Analysis

DeepHit (Lee et al. AAAI 2018)



$$
\begin{aligned}
S(b) &= P(z \leq b | \boldsymbol{x}) \\
&= \sum_{j=0}^{b} P(z = z_j | \boldsymbol{x})
\end{aligned}
\tag{10}
$$

Evaluation

Log-Likelihood

$$\bar{P} = -\frac{1}{N} \sum_{(\boldsymbol{x}_i, z_i) \in D^{\text{test}}} \log p'_z(z_i | \boldsymbol{x}_i) \ , \tag{11}$$

where $N = |D^{\text{test}}|$ is the number of the test dataset and $p'_t(t|\boldsymbol{x})$ is the learned P.D.F.

Evaluation

Relationship between hazard and P.D.F.

$$
\begin{aligned}
\lambda(b) &= \lim_{db \to 0} \frac{Pr(b \leq z \leq b + db | z > b)}{db} \\
&= \lim_{db \to 0} \frac{Pr(b \leq z \leq b + db)/Pr(z > b)}{db} \\
&= \lim_{db \to 0} \frac{(w_z(b + db) - w_z(b))/S(b)}{db} = \frac{p_z(b)}{S(b)} = -\frac{S'(b)}{S(b)} .
\end{aligned}
\tag{12}
$$

$$
\begin{aligned}
p_t(t|\boldsymbol{x}) &= \frac{\partial w_t(t|\boldsymbol{x})}{\partial t} = \frac{-\partial S(t|\boldsymbol{x})}{\partial t} \\
&= \frac{\partial \exp(\int_0^t \lambda(v|\boldsymbol{x}) dv)}{\partial t} \\
&= \exp\left(\int_0^t \lambda(v|\boldsymbol{x}) dv\right) \lambda(t|\boldsymbol{x}) .
\end{aligned}
\tag{13}
$$

Evaluation

Concordance Index (C-index)

Considering all possible pairs $(T_i, E_i), (T_j, E_j)$ for $i \leq j$, the C-index is calculated by considering the number of pairs correctly ordered by the model divided by the total number of admissible pairs.
admissible: can be ordered in a meaningful way. (uncensored, uncensored); (uncensored, right-censored). admissible pairs.