Gradient Boosting Survival Tree with Applications in Credit Scoring

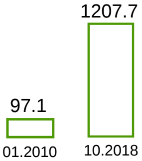
Miaojun Bai, Yan Zheng, **Yun Shen**

360 Finance Inc. (Nasdaq: QFIN)

Credit Scoring and Credit Control XVI, Edinburgh, 29.08.2019

Outline

Chinese consumer finance market



Market size ($ billion)

- Rapid growth
- Heterogeneous data
  - PBC report: 1/3 has credit ratings
  - personal info.
  - device info.
  - third party rate agencies
- Changing market conditions
  - regulation
  - macroeconomic factor

Motivation

- Pros of tree ensemble methods (e.g., XGB, LightGBM)
  - robust for heterogeneous data
  - fast modeling for credit scoring
  - utilize numerous "weak" attributes

Motivation

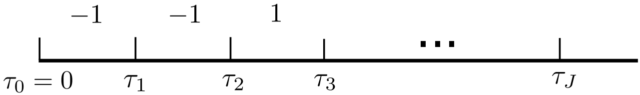
- Pros of tree ensemble methods (e.g., XGB, LightGBM)
    - robust for heterogeneous data
    - fast modeling for credit scoring
    - utilize numerous "weak" attributes
- Pros of survival analysis
    - predict the probability of default time
    - take long-term behavior into consideration

Motivation

- Pros of tree ensemble methods (e.g., XGB, LightGBM)
  - robust for heterogeneous data
  - fast modeling for credit scoring
  - utilize numerous “weak” attributes
- Pros of survival analysis
  - predict the probability of default time
  - take long-term behavior into consideration
- Idea: **survival analysis** + **tree ensemble methods**?

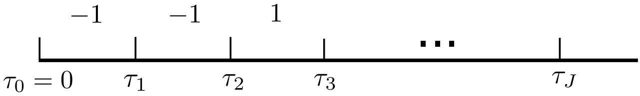
Survival analysis

- Survival function: $S(t) = \mathbb{P}(T > t)$
- Discrete time periods



Survival analysis

- Survival function: $S(t) = \mathbb{P}(T > t)$
- Discrete time periods



- Hazard function:

$$h(\tau_j) := \mathbb{P}(\tau_{j-1} < T \leq \tau_j | T > \tau_{j-1}), \quad j = 1, 2, \ldots,$$

Hence,

$$S(\tau_j) = \prod_{l=1}^{j} (1 - h(\tau_l))$$

- Likelihood

$$\mathbb{P}(\tau_{j-1} < T \leq \tau_j) = h(\tau_j) S(\tau_{j-1}) = h(\tau_j) \prod_{l=1}^{j-1} (1 - h(\tau_l))$$

Likelihood

- Log hazard function

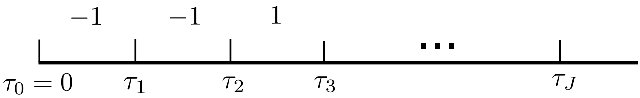
$$f(t) := \log \frac{h(t)}{1 - h(t)}$$

- Likelihood

$$\mathbb{P}(T = t) = \prod_{j=1}^{J(t) \wedge J} \frac{1}{1 + e^{-y_j(t) f(\tau_j)}},$$

where

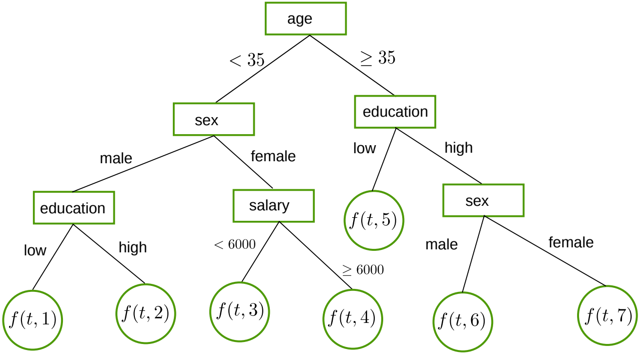
$$J(t) := \begin{cases} j, & \text{if } t \in (\tau_{j-1}, \tau_j] \\ J + 1, & \text{if } t > \tau_J \end{cases} \qquad y_j(t) := \begin{cases} -1, & \text{if } t > \tau_j \\ 1, & \text{if } t \leq \tau_j \end{cases}$$



Learning objective

- For each individual $\boldsymbol{x}$, f is approximated by a survival tree ensemble

$$f(t;\boldsymbol{x}) \cong \hat{f}(t;\boldsymbol{x}) := \sum_{k=1}^{K} f_k(t;\boldsymbol{x})$$



Learning objective

- To minimize the negative log-likelihood

$$\mathcal{L} = \sum_{i=1}^{N} \sum_{j=1}^{J(t_i)\wedge J} \log\left(1 + \exp\left\{-y_j(t_i)\hat{f}(\tau_j;\boldsymbol{x}_i)\right\}\right) + \frac{\lambda}{2}\|w\|^2$$
$$= \sum_{j=1}^{J} \sum_{i\in N_j} \log\left(1 + \exp\left(-y_j(t_i)\hat{f}(\tau_j;\boldsymbol{x}_i)\right)\right) + \frac{\lambda}{2}\|w\|^2$$

  where $N_j := \{i \in \{1,2,\ldots,N\} | J(t_i) \geq j\}$ is the set of samples surviving longer than τ_{j-1}.
- Regularization term
  - punish model complexity
  - avoid over-fitting
  - overcome numerical problems

Gradient tree boosting

- Boosting algorithm:
  - At mth iteration, given $\hat{f}^{(m-1)}$

$$\min_f \mathcal{L}^{(m)} = \sum_{j,i} \log\left(1 + \exp\left\{-y_j(t_i)\left(\hat{f}^{(m-1)}(\tau_j; \boldsymbol{x}_i) + f(\tau_j; \boldsymbol{x}_i)\right)\right\}\right) + \frac{\lambda}{2}\|w\|^2 \quad \Rightarrow \quad f_m$$

  - update $\hat{f}^{(m)}(t; \boldsymbol{x}) = \hat{f}^{(m-1)}(t; \boldsymbol{x}) + f_m(t; \boldsymbol{x})$
- Approximate by Taylor expansion up to the 2nd order

$$\mathcal{L}^{(m)}(f) \cong \sum_{j,i}\left(r_{i,j}^{(m-1)} f(\tau_j; \boldsymbol{x}_i) + \frac{1}{2}\sigma_{i,j}^{(m-1)} f^2(\tau_j; \boldsymbol{x}_i)\right) + \frac{\lambda}{2}\|w\|^2$$

Gradient tree boosting

- Survival tree with L nodes: $f(\tau_j; \mathbf{x}_i) = \sum_{l=1}^{L} w_l(\tau_j)\mathbf{1}(i \in I_l)$
- The objective function is strictly convex with optimal solution

$$w_l^{(m)}(\tau_j) = -\frac{\sum_{i \in N_j \cap I_l} r_{i,j}^{(m-1)}}{\sum_{i \in N_j \cap I_l} \sigma_{i,j}^{(m-1)} + \lambda}$$

- Split rule: $I = I_L \cup I_R$

$$\begin{aligned}\tilde{\mathcal{L}}_{split} = & \frac{1}{2} \sum_j \frac{\left(\sum_{i \in N_j \cap I_L} r_{i,j}^{(m-1)}\right)^2}{\sum_{i \in N_j \cap I_L} \sigma_{i,j}^{(m-1)} + \lambda} + \frac{\left(\sum_{i \in N_j \cap I_R} r_{i,j}^{(m-1)}\right)^2}{\sum_{i \in N_j \cap I_R} \sigma_{i,j}^{(m-1)} + \lambda} \\ & - \frac{\left(\sum_{i \in N_j \cap I} r_{i,j}^{(m-1)}\right)^2}{\sum_{i \in N_j \cap I} \sigma_{i,j}^{(m-1)} + \lambda}.\end{aligned}$$

Summary

- Log hazard function is approximated by a survival tree ensemble
- maximum likelihood as the objective function
- boosting algorithm
- for each step, a gradient method applied to optimize the approximated objective up to 2nd order

Datasets

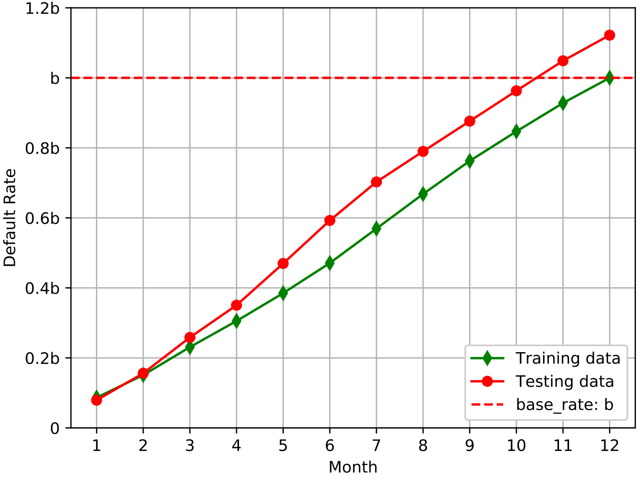
- Installment loans with 12 months
- Definition of default: if on any scheduled repayment due date the borrower is overdue for at least 10 days
- Early repayments: regarded as “repaying on time” in the rest time
- training and testing datasets

dataset	time	sample size
training set	January 2018	200,000
testing set	March 2018	120,000

- Default rate

$$\text{default rate}(t) = \frac{\#\text{default accounts up to month } t}{\#\text{total accounts}}$$

Default rates on datasets



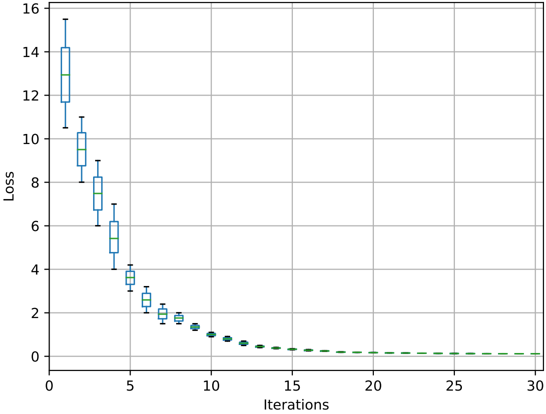
Dataset and preprocessing

- Over 400 original attributes are collected
  - exclude attributes with missing rate higher than 80%
  - one-hot encoding for categorical attributes
- 50 features are selected by xgboost

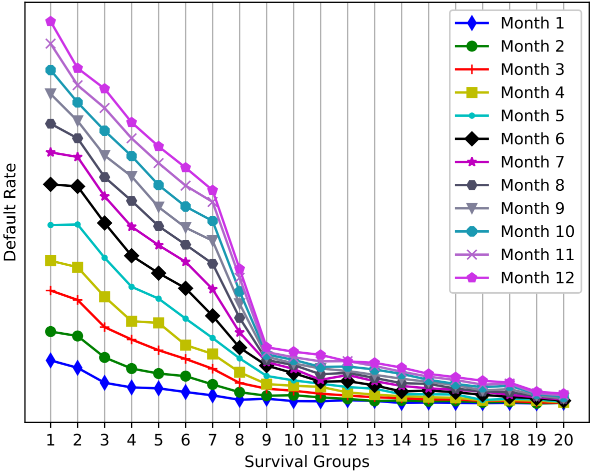
source	**feature**
PBC report	income score
	credit score
	overdue information of credit cards
personal information	age
	sex
	education level
device information	location
third-party rate agency	no. of loans in other lending platforms
	travel intensity
other information	whether possessing a car
	application channel

Convergence

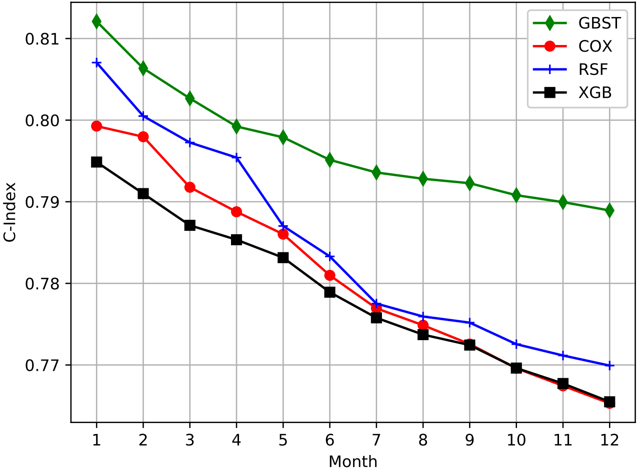
- 1000 runs with $\lambda = 0.001$ and the max tree depth 6



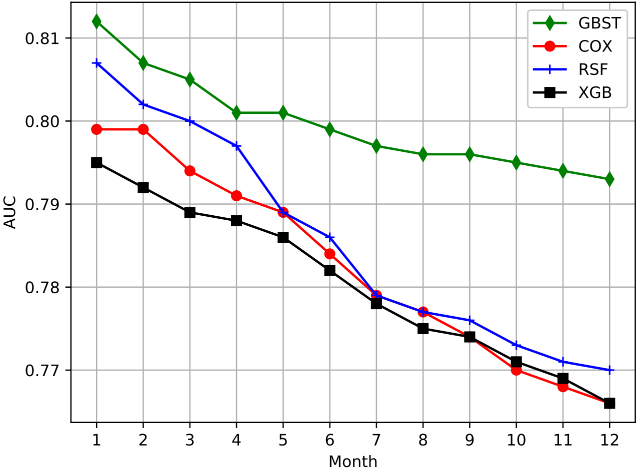
Performance



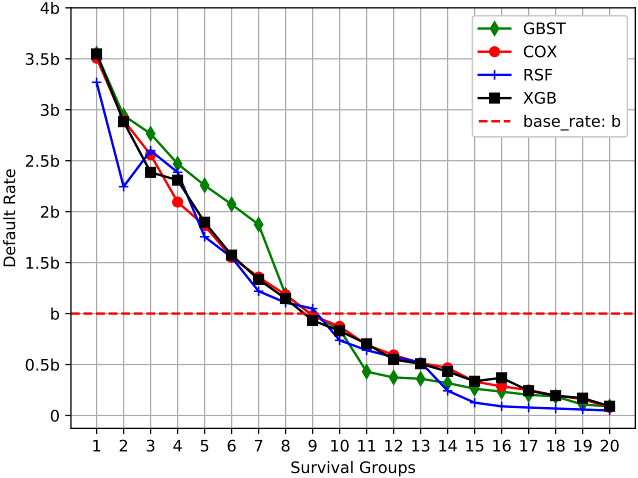
Comparison with existing models: C-Index



Comparison with existing models: AUC



Comparison with existing models



Conclusion

- Propose the gradient boosting survival tree (GBST) model
- Confirm the convergence of GBST with a real dataset
- GBST outperforms existing survival analysis and machine learning models

Conclusion

- Propose the gradient boosting survival tree (GBST) model
- Confirm the convergence of GBST with a real dataset
- GBST outperforms existing survival analysis and machine learning models

Thank you!