Journal of Biostatistics and Epidemiology

J Biostat Epidemiol. 2023;9(3):378-387

Original Article

Comparison of Two Methods, Gradient Boosting and Extreme Gradient Boosting to Predict Survival in Covid-19 Data

Nadiasadat Taghavi Razavizadeh[1], Maryam Salari[1], Mostafa Jafari[2], Ehsan Sabaghian[3], Vahid Ghavami[1*]

1Department of Biostatistics, School of Health, Social Determinants of Health Research Center, Mashhad University of Medical Sciences, Mashhad, Iran.
2Department of Internal Diseases, Mashhad University of Medical Sciences, Mashhad, Iran.
3Department of Plant Biotechnology and Bioinformatics,Ghent University, Ghent, Belgium, and VIB Center for Plant Systems Biology, Ghent, Belgium.

ARTICLE INFO

Received 26.03.2023
Revised 27.03.2023
Accepted 04.05.2023
Published 15.09.2023

Key words:
Gradient boosting algorithm;
Extreme gradient boosting algorithm;
Survival analysis;
Covid-19.

ABSTRACT

Introduction: The present study discusses the importance of having a predictive method to determine the prognosis of patients with diseases like Covid-19. This method can assist physicians in making treatment decisions that improve survival rates and avoid unnecessary treatments. This research also highlights the importance of calibration, which is often overlooked in model evaluation. Without proper calibration, incorrect decisions can be made in disease treatment and preventive care. Therefore, the current study compares two highly accurate machine learning algorithms, Gradient boosting and Extreme gradient boosting, not only in terms of prediction accuracy but also in terms of model calibration and speed.

Methods: This study involved analyzing data from Covid-19 patients who were admitted to two hospitals in Mashhad city, Razavi Khorasan province, over a span of 18 months. The k-fold cross-validation method was employed on the training dataset (K=5) to conduct the study. The accuracy and calibration of two methods (Gradient boosting and Extreme gradient boosting) in predicting survival were compared using the Concordance Index and calibration.

Results: The Concordance Index values obtained for gradient boosting and Extreme gradient boosting models were 0.734 and 0.736, in the imbalanced and In the balanced data, the Concordance Index values were 0.893 for gradient boosting and 0.894 for Extreme gradient boosting. The surv.calib_beta index, the gradient boosting model had an estimated value of 0.59 in the imbalanced data and 0.66 in the balanced data. The Extreme gradient boosting model had an estimated value of 0.86 in the balanced data and 0.853 in the imbalanced data. The Extreme gradient boosting model was faster in the learning process compared to the gradient boosting model.

Conclusion: The Gradient boosting and Extreme gradient boosting models exhibited similar prediction accuracy and discrimination power, but the Extreme gradient boosting model demonstrated relatively good calibration compare to Gradient boosting model.

*.Corresponding Author: ghavamiv@mums.ac.ir

Introduction

The survival rate for patients with Covid-19 is generally high, but severe cases can result in mortality.[1] Therefore, the development of a predictive method to determine patient prognosis based on their characteristics can assist physicians in selecting appropriate treatment options for better survival outcomes and avoiding unnecessary treatments.[2] Numerous methods exist to achieve this goal; however, selecting the most accurate method is crucial. Research suggests that machine learning methods typically yield more precise results compared to traditional survival methods.[3-5] Machine learning algorithms utilize statistical and probabilistic techniques, optimizing them to analyze large, complex, and unstructured datasets. By learning from past experiences, these algorithms can identify appropriate patterns and predict survival rates while identifying related factors.[6] Due to the substantial amount of data involved, training these algorithms can be time-consuming. Therefore, it is vital to identify algorithms that offer both high accuracy and speed. Another essential criterion, often overlooked in model evaluations, is calibration, which is essential to ensure appropriate decisions regarding disease treatment and preventive care.[7] The aim of this study is to compare two highly accurate machine learning algorithms, Gradient boosting and XGBoost, considering prediction accuracy, model calibration, and computational speed.

Methods

Gradient boosting Algorithm

The gradient boosting machine (GBM) is an machine learning method, which constructs a predictive model by additive expansion of sequentially fitted weak learners. The general problem is to learn a functional mapping $y=(x;\beta)$ from data

$\{x_i, y_i\}_{i=1}^{n}$ where β is the set of parameters of F, such that some cost function

$$\sum_{i=1}^{n} \Phi(y_i, F(x_i; \beta))$$

is minimized. Boosting assumes $f(x)$ follows an additive expansion for.

$$F(x) = \sum_{m=0}^{M} \rho_m f(x; \tau_m),$$

where f is called the weak or base learner with a weight ρ and a parameter set τ. Accordingly,

$$\{\rho_m, \tau_m\}_{m=1}^{M}$$

compose the whole parameter set β. They are learnt i n a greedy "stage-wise" process: (1) set an initial estimator $f_0(x)$; (2) for each iteration $m \in \{1, 2, \ldots, M\}$, solve

$$(\rho_m, \tau_m) = \arg\min_{\rho,\tau} \sum_{i=1}^{n} \Phi(y_i, F_{m-1}(x_i) + \rho f(x_i; \tau)).$$

GBM approximates (2) with two steps. First, it fits $(x; \tau_m)$ by

$$\tau_m = \arg\min_{\tau} \sum_{i=1}^{n} \left(g_{im} - f(x_i;\tau)\right)^2 \quad (1)$$

Where

$$g_{im} = -\left[\frac{\partial \Phi(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)} \quad (2)$$

Second, it learns ρ by

$$\rho_m = \arg\min_{\rho} \sum_{i=1}^{n} \Phi(y_i, F_{m-1}(x_i) + \rho f(x_i; \tau_m)) \quad (3)$$

Then, it updates $F_m(x)=F_{m-1}(x) + pf(x; \tau_m))$. In practice, however, shrinkage is often introduced to control overfitting and the update becomes $F_m(x)=F_{m-1}(x) + \vartheta p_m f(x; \tau_m)$ where $0<\vartheta \leq 1$. If the weak learner is the regression tree, the complexity of $f(x)$ is determined by tree parameters for example, the tree size (or depth), and the minimum number of samples in terminal nodes. Besides using proper shrinkage and tree param eters, one could improve the GBM performance by subsampling, that is, fitting each base learner on a random subset of the training data. This method is called stochastic gradient boosting.[8]

GBM has been implemented in the popular open-source R package "gbm" which supports several regression models.[9]

Ridgeway adapted GBM for the Cox model. The cost function is the negative log partial likelihood:

$$\Phi(y,F) = -\sum_{i=1}^{n} \delta_i \left\{ F(x_i) - log\left\{ \sum_{j:t_j \geq t_i} e^{F(x_j)} \right\} \right\} \quad (4)$$

One can then apply (1), (2), and (3) to learn each additive model.

In our study, we utilized the GBM algorithm, which was implemented using the gbmcox package named gbmcox.[8] Following Chen et al.'s research, this algorithm was executed with the negative log partial likelihood cost function.

Extreme gradient boosting (XGBoost) Algorithm

Both the gradient boosting and XGBoost methods share the same principle of gradient boosting and have similar mathematical concepts. However, the XGBoost method incorporates more regularization compared to the Gradient Boosting method. Specifically, the XGBoost method employs two types of regularization, L1 and L2, whereas the Gradient Boosting method only utilizes L1 regularization. This increase in regularization techniques has enhanced the model's ability to generalize to new data. L1 regularization, also known as Lasso Regression, involves multiplying the sum of absolute values of all coefficients by a constant and adding it as a penalty term to the loss function.

$$\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|) \quad (5)$$

Ridge regression, also known as L2 regularization, is an additional term used in regularization. In L2 regularization, the penalty term in the loss function is derived from the sum of the squared values of all the weights on the connections within the neural network of the machine learning model.[10-11]

$$\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j{}^2) \quad (6)$$

The L2 norm, in contrast to the L1 norm, allows for the learning of intricate patterns in the input data. However, the L2 norm does not handle outlier data well in the dataset used. This is because outliers noticeably increase the prediction error of the model, causing the weights of the model to become smaller due to the normal penalty term L2 in the function. On the other hand, the L1 norm exhibits better performance and resilience when confronted with outliers in the input data. Therefore, it is advisable to combine both norms to compensate for each other's limitations and achieve a higher-performing model. In this

study, the xgboost and survXgboost packages in RStudio software were utilized for the implementation of this algorithm.[12-13]

Data source and data extraction

This research is a historical cohort study that investigates 34,925 patients who were hospitalized with covid-19 at Imam Reza and Qaem hospitals in Mashhad, Razavi Khorasan Province, during the period of March 2020 to September 2021.
The diagnosis of covid-19 was based on clinical examinations, blood and PCR tests, as well as lung CT scans. All relevant information, such as admission date, demographic characteristics, underlying diseases, clinical symptoms, diagnostic test results, and date of discharge or death, was collected from the University Health Information System (HIS). A total of 63 features were examined in this study.

Comparison of models

After the data was extracted, it underwent refinement and sorting. Clinical data often faces the issue of imbalanced classes, where one class has a noticeably larger number of samples compared to the others. In our study, for instance, the death rate due to covid-19 was 21%, while the censoring rate was 79%. This imbalance, known as data imbalance, can impact the model's performance and yield unreliable outcomes. To address this, researchers typically employ techniques such as oversampling (repeating random records from the minority class), undersampling (removing random records from the majority class), or a combination of both with a ratio of 0.5.

In our study, we compared the models using imbalanced and balanced data by employing the combined method with a ratio of 0.5. This was accomplished through the ROSE package and the ovun.sample function.[14]
Next, we utilized the k-fold cross-validation method to create subsets within the training dataset, with k=5. We compared the accuracy and calibration of the models in predicting survival on the experimental dataset using the Concordance index (C-Index) and calibration. The mlr3tuning package was employed to optimize the hyperparameters.[15] All statistical analyses were conducted using R software.[16]
Cross-validation is an effective measure to prevent overfitting as it allows the model to be trained on multiple datasets instead of relying on a single test and train dataset. This ensures the model's performance on unseen data and its generalizability. One commonly used cross-validation method is the k-fold, which randomly divides the training dataset into k subsamples of equal size. Each stage of the cross-validation process involves using k-1 of these subsamples as the training dataset and one as the validation dataset to calculate performance evaluation indices such as the C-Index. The average C-Index or calibration obtained from these k steps represents the final Concordance index or final calibration for evaluating the model's performance.[17]

Concordance Index (C-Index)

The concordance index (C-index) or Harrell’s CH is one of tools used for evaluating the performance of a survival model. Intuitively, it is the fraction of all pairs of patients whose predictions have correct orders over the pairs that can be ordered. Formally, the C-index is

$$CI=\frac{1}{|\rho|}\sum_{(i,j)\epsilon\rho} I(F(x_i)<F(x_j))=\frac{1}{|\rho|}\sum_{i\epsilon E}\sum_{j:\,t_j>t_i} I(F(x_i)<F(x_j)) \quad (7)$$

ρ is the set of validly orderable pairs, where $t_i < t_j$; |ρ| is the number of pairs in ρ; *F(x)* is the prediction of survival time; *I* is the indicator function of whether the condition in parentheses is satisfied or not. In the PH setting, the predicted survival time can be equivalently represented by the negative log relative hazard. The C-index estimates the probability that the order of the predictions of a pair of comparable patients is consistent with their observed survival information.[8]

surv.calib_beta

Calibration is an important statistical measure that evaluates the effectiveness of predictive models. The C-Index and similar criteria assess the agreement between predicted and actual values, indicating how accurately the model distinguishes between patients.[18]

Calibration, on the other hand, measures the proximity of observed and predicted absolute risks.[7] The surv.calib_beta method involves fitting the predicted linear predictor from a Cox PH model as the sole predictor in a new Cox PH model with the test data as the response variable.

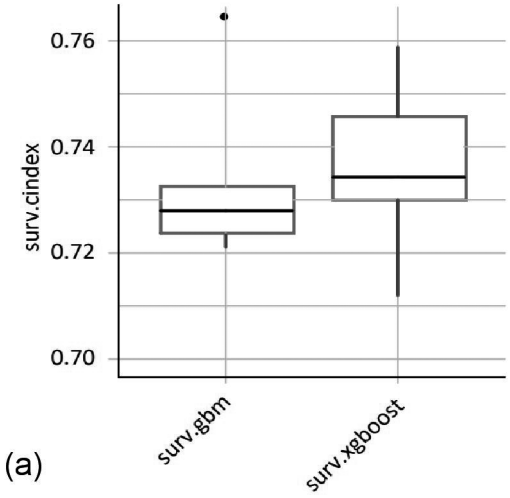
$$h(t|x)=h_0(t)\exp(l\beta) \quad (8)$$

Where l is the predicted linear predictor. The model is well calibrated if β the estimated coefficient is equal to 1.[19] A slope < 1 suggests that estimated risks are too extreme, i.e., too high for patients who are at high risk and too low for patients who are at low risk. A slope > 1 suggests the opposite, i.e., that risk estimates are too moderate.[20]

Quality of analysis: we support that our analysis supports following the Standards for Reporting of Diagnostic Accuracy Studies (STARD) guidelines by Cohen et al.,[21] or the proposed STARD-AI guidelines by Sounderajah et al.[22]

Results

In the case of imbalanced data, the C-index values for the gradient boosting and XGBoost models are 0.734 and 0.746, respectively. In balanced data, the C-index values are 0.893 for gradient boosting and 0.894 for XGBoost (Table 1). Based on Figure 1(a,b), it seems that there is no large difference between the two models in terms of C-index value. However, balancing the data has led to an increase in model accuracy by approximately 0.15, which is a relatively substantial improvement.



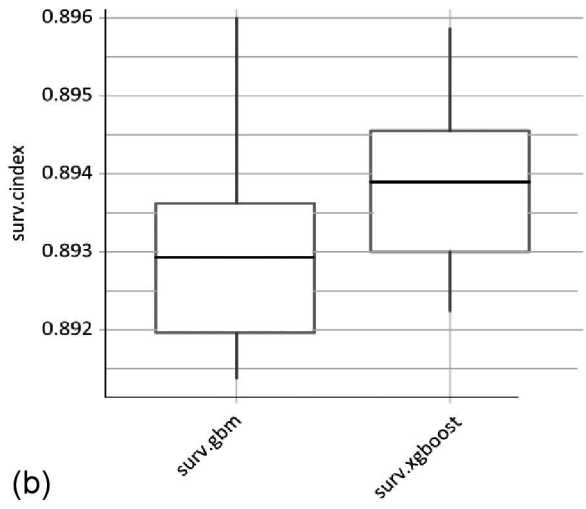


Figure 1. Box plot of surv.index index for XGBoost and Gradient Boosting model in imbalanced data (a) and balanced data (b)

Regarding the surv.calib_beta index, the values for the imbalanced data are 0.593 for gradient boosting and 0.86 for XGBoost. In balanced data, the values are 0.66 for gradient boosting and 0.853 for XGBoost (Table 1). Figure 2(a,b) clearly demonstrate a noticeable distinction between the two models, with the XGBoost model exhibiting better calibration.

In terms of training time, the gradient boosting model takes approximately 121.10 seconds, whereas the XGBoost model only requires 2.59 seconds. This indicates that the XGBoost model undergoes the learning process around 47 times faster than the gradient boosting model.

Discussion

Table 1. Comparison of Performance of XGBoost and Gradient Boosting methods in imbalanced and balanced data

Model	surv.C-index imbalanced data	surv.C-index balanced data	surv.calib_beta imbalanced data	surv.calib_beta balanced data
Surv.XGBoost	0.746	0.894	0.860	0.853
Surv.GBM	0.734	0.893	0.593	0.660

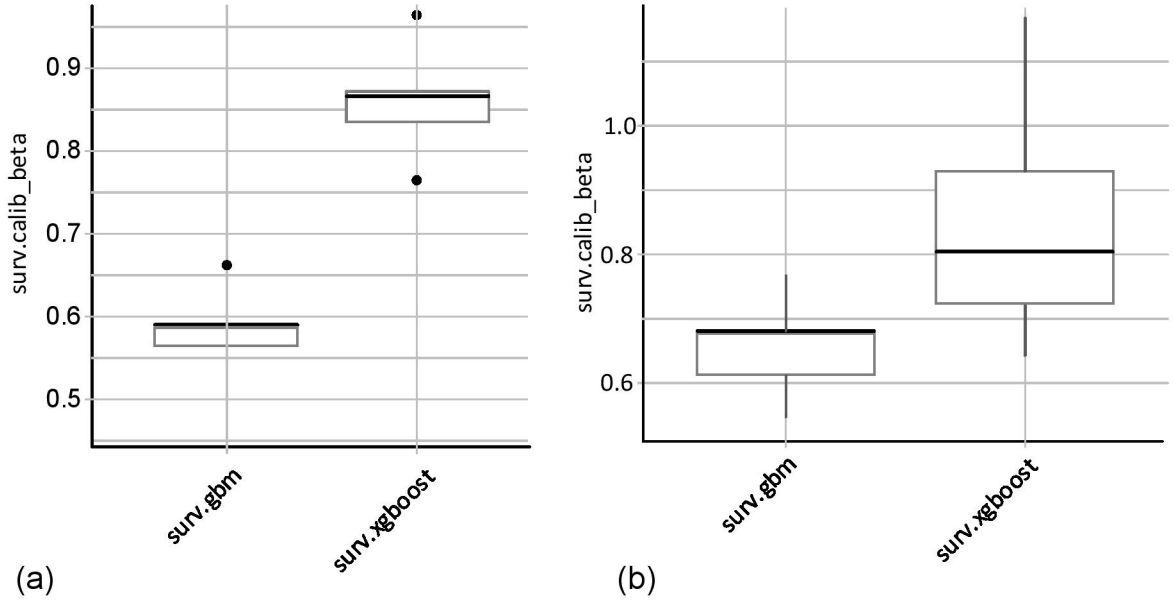


Figure 2. Box plot of surv.calib_beta index for XGBoost and Gradient Boosting model in imbalanced data (a) and balanced data (b)

Survival analysis is a statistical approach used to model the time until a specific event occurs, such as death, hospital discharge, or ICU admission.[23] In this type of analysis, accurately predicting patient prognosis, identifying factors related to survival, and predicting outcomes like death or disease recurrence are crucial for making informed decisions regarding treatment, monitoring, and disease prevention. However, even if a model has high accuracy, its results cannot be considered reliable unless the model is well calibrated. Failure to calibrate the model can introduce bias in risk estimation for patients at high or less common risk of the desired outcome, ultimately impacting clinical decisions.[7] Therefore, striking a balance between predictive power and calibration is always emphasized.[24]

In our study, both the Gradient boosting and XGBoost models exhibited similar prediction accuracy, with both algorithms demonstrating relatively good accuracy and discrimination power. Previous studies comparing Gradient boosting and XGBoost methods with traditional methods like logistic regression (LR) and other machine learning techniques such as Support Vector Machine (SVM) and Naïve Bayes (NB) have consistently shown higher accuracy for Gradient boosting and XGBoost methods.[25,26] Nevertheless, it seems that the gradient boosting model was not well calibrated for both balanced and imbalanced data, as suggested by the surv.calib_beta value. On the other hand, the XGBoost model demonstrated good calibration with a surv. calib_beta index close to one. In a study by Hu et al., which aimed to identify pregnant women at risk of gestational diabetes, the XGBoost model not only exhibited higher accuracy compared to the traditional LR model but also showed desirable calibration.[27] Additionally, it was observed that data balancing only affected the C-index and consequently the model's accuracy. In our study, it appears that while data balancing can improve the model's accuracy to some extent, calibration heavily depends on the specific model employed.

Regarding training time, it is worth mentioning that both Gradient boosting and XGBoost algorithms are based on decision trees. Therefore, parallel execution of multiple trees simultaneously is practically unfeasible due to the need for predictions after each tree to update gradients. However, the XGBoost method employs parallelization within a tree to create branches independently using openMP, resulting in a shortened learning process.[12] As the number of data points and features examined in a study increases, the disparity in training time between the Gradient boosting and XGBoost algorithms will likely grow exponentially.

Conclusion

Both the Gradient boosting and XGBoost models exhibited similar prediction accuracy and discrimination power, but the XGBoost model demonstrated relatively good calibration compare to Gradient boosting model. Balancing the data can enhance the accuracy of the model to some extent, but the calibration performance is influenced by the specific type of model used. Also the XGBoost model undergoes the learning process faster than the gradient boosting model.

Acknowledgments

The researchers express their gratitude and

appreciation to the staff of Statistics and Information Technology Management at Mashhad University of Medical Sciences for their assistance with researchers in data gathering. This paper is a part of a thesis in department of biostatistics in school of health of Mashhad University of Medical Sciences and was financially supported by the Vice-chancellor for Research and Technology, Mashhad University of Medical Sciences, Iran (project No. 4000853).

Conflict of interest

The authors declare no conflict of interest.

References

1. Aljameel SS, Khan IU, Aslam N, Aljabri M, Alsulmi ES. Machine Learning-Based Model to Predict the Disease Severity and Outcome in COVID-19 Patients. Scientific Programming. 2021;2021.

2. Kim DW, Lee S, Kwon S, Nam W, Cha I-H, Kim HJ. Deep learning-based survival prediction of oral cancer patients. Scientific Reports. 2019;9(1):6994 .

3. Kantidakis G, Putter H, Lancia C, Boer Jd, Braat AE, Fiocco M. Survival prediction models since liver transplantation - comparisons between Cox models and machine learning techniques. BMC Medical Research Methodology. 2020;20(1):277.

4. Kirişci M. Comparison of artificial neural network and logistic regression model for factors affecting birth weight. SN Applied Sciences. 2019;1(4):378.

5. Spooner A, Chen E, Sowmya A, Sachdev P, Kochan NA, Trollor J, et al. A comparison of machine learning methods for survival analysis of high-dimensional clinical data for dementia prediction. Scientific Reports. 2020;10(1):20410.

6. Uddin S, Khan A, Hossain ME, Moni MA. Comparing different supervised machine learning algorithms for disease prediction. BMC Medical Informatics and Decision Making. 2019;19(1):281.

7. Soave DM, Strug LJ. Testing Calibration of Cox Survival Models at Extremes of Event Risk. Frontiers in Genetics. 2018;9.

8. Chen Y, Jia Z, Mercola D, Xie X. A Gradient Boosting Algorithm for Survival Analysis via Direct Optimization of Concordance Index. Computational and Mathematical Methods in Medicine. 2013;2013:873595.

9. Brandon Greenwell BB, Jay Cunningham, GBM Developers gbm: Generalized Boosted Regression Models [Available from: https://cran.r-project.org/web/packages/gbm/index.html.

10. Khandelwal N. A Brief Introduction to XGBoost: Towards Data Science; 2020 [updated Jul 7, 2020. Available from: https://towardsdatascience.com/a-brief-introduction-to-xgboost-3eaee2e3e5d6.

11. Tewarisv U. Understanding L1 and L2 regularization for Deep Learning: Analytics Vidhya; 2021 [Available from: https://

medium.com/analytics-vidhya/regularization-understanding-l1-and-l2-regularization-for-deep-learning-a7b9e4a409bf.

12. Tianqi Chen TH, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, Rory Mitchell, Ignacio Cano, Tianyi Zhou, Mu Li, Junyuan Xie, Min Lin, Yifeng Geng, Yutian Li, Jiaming Yuan. xgboost: Extreme Gradient Boosting [Available from: https://cran.r-project.org/web/packages/xgboost/index.html.

13. GitHub. Full survival curve estimation using xgboost.[Available from: https://rdrr.io/github/IyarLin/survXgboost/.

14. Nicola Lunardon GM, Nicola Torelli. ROSE-package: ROSE: Random Over-Sampling Examples [Available from: https://rdrr.io/cran/ROSE/man/ROSE-package.html.

15. Marc Becker ML, Jakob Richter, Bernd Bischl, Daniel Schalk. mlr3tuning: Hyperparameter Optimization for 'mlr3' [Available from: https://cran.r-project.org/web/packages/mlr3tuning/index.html.

16. Team, R. D. C. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. Retrieved from http://www.R-project.org.

17. Hu Y-J, Ku T-H, Jan R-H, Wang K, Tseng Y-C, Yang S-F. Decision tree-based learning to predict patient controlled analgesia consumption and readjustment. BMC Medical Informatics and Decision Making. 2012;12(1):131.

18. Vickers AJ, Cronin AM. Everything you always wanted to know about evaluating prediction models (but were too afraid to ask). Urology. 2010;76(6):1298-301.

19. Van Houwelingen, C. H (2000). "Validation, calibration, revision and combination of prognostic survival models." Statistics in Medicine, 19(24), 3401-3415.

20. Van Calster B, McLernon DJ, van Smeden M, Wynants L, Steyerberg EW, Bossuyt P, et al. Calibration: the Achilles heel of predictive analytics. BMC Medicine. 2019;17(1):230.

21. Cohen JF, Korevaar DA, Altman DG, Bruns DE, Gatsonis CA, Hooft L, et al. STARD 2015 guidelines for reporting diagnostic accuracy studies: explanation and elaboration. BMJ Open. 2016;6: e012799. doi: 10.1136/bmjopen-2016-012799

22. 43. Sounderajah V, Ashrafian H, Golub RM, Shetty S, De Fauw J, Hooft L, et al. Developing a reporting guideline for artificial intelligence-centred diagnostic test accuracy studies: the STARD-AI protocol.

23. Kleinbaum, D.G. and Klein, M. (2012) Survival Analysis: A Self-Learning Text. 3rd Edition, Springer, NewYork.

24. Goldstein M, Han X, Puli A, Perotte AJ, Ranganath R. X-CAL: Explicit Calibration for Survival Analysis. Adv Neural Inf Process Syst. 2020;33:18296-307.

25. Ayumi V, editor Pose-based human action recognition with Extreme Gradient Boosting. 2016 IEEE Student Conference on Research

and Development (SCOReD); 2016 13-14 Dec. 2016.

26. Seto H, Oyama A, Kitora S, Toki H, Yamamoto R, Kotoku Ji, et al. Gradient boosting decision tree becomes more reliable than logistic regression in predicting probability for diabetes with big data. Scientific Reports. 2022;12(1):15889.

27. Hu X, Hu X, Yu Y, Wang J. Prediction model for gestational diabetes mellitus using the XGBoost machine learning algorithm. Front Endocrinol (Lausanne). 2023;14:1105062.