Proceedings of the 5th International Conference on Biological Engineering and Medical Science
DOI: 10.54254/2753-8818/99/2025.22531

Kidney Stone Risk Prediction by XGBoost Model Based on Optimization of Newton-Raphson Algorithm

Haoqi Wu[1*], Wuyuzhou Qin[2], Xu Liu[1]

[1]*Institute of Engineering, China Pharmaceutical University, Nanjing City, China*
[2]*Shenyang Medical College, Shenyang City, China*
**Corresponding Author. Email: 2020231760@stu.cpu.edu.cn*

Abstract: In this study, an XGBoost model optimized based on the Newton-Raphson algorithm is proposed for the task of kidney stone risk prediction, which improves the parameter updating strategy of the traditional gradient boosting framework by introducing second-order derivative information. In order to verify the effectiveness of the model, the performance differences between decision trees, random forests, standard XGBoost, CatBoost and the optimized model are systematically compared. The experimental results show that the XGBoost model optimized by the Newton-Raphson algorithm reaches 0.875 in both accuracy and recall indexes, which is significantly better than the other compared models, and its balanced assessment indexes both reflect the accurate identification ability of positive samples and verify the reliability of the overall prediction performance. Particularly noteworthy is that although Random Forest and standard XGBoost perform consistently in accuracy and recall, the differences in precision rate and AUC value reveal the essential difference between the two in feature space division and integration strategy: Random Forest reduces the risk of overfitting through feature randomness, while XGBoost relies on the regularization term to control the model complexity. The research results not only confirm the feasibility of the optimization algorithm in improving the performance of medical prediction models, but also provide an intelligent tool with practical application value for early screening and risk assessment of kidney stones in clinical practice with its stable prediction accuracy of 0.875.

Keywords: Kidney stones, Newton-Raphson algorithm, XGBoost.

1. Introduction

In recent years, studies on the prediction of kidney stones based on urinalysis have gradually shifted from the detection of traditional biochemical indicators to multidimensional characterization [1]. Existing studies have confirmed that the concentrations of biochemical components such as calcium, oxalic acid, and uric acid in urine are closely related to calcium oxalate crystal formation, but these indicators alone have limited sensitivity and specificity for prediction. Meanwhile, the physical characteristics of urine (e.g., pH, specific gravity, turbidity, conductivity, etc.) are gradually gaining attention. However, most studies are still dominated by single-factor statistical analysis, and the mechanism of multifactorial interaction has been insufficiently explored [2]. In addition, traditional statistical models have bottlenecks in the capture of complex nonlinear relationships, which leads to difficulties in breaking through the prediction accuracy. In recent years, some studies have begun to

DOI: 10.54254/2753-8818/99/2025.22531

try to integrate biochemical indicators and physical features, but the data fusion methods and dynamic monitoring techniques still need to be optimized [3].

Machine learning algorithms have shown significant advantages in kidney stone prediction research, especially in dealing with high-dimensional, nonlinear data. First, supervised learning models (Random Forest, Support Vector Machine, Gradient Boosting Tree) are able to integrate urinary biochemical indicators (calcium, oxalic acid concentration) with physical features (pH, conductivity), and reveal the key predictors through feature importance ranking. Second, deep learning models (e.g., convolutional neural networks) can process urine microscopy image data to automatically identify morphological features of calcium oxalate crystals (e.g., birefringence, cluster size), which can make up for the subjective defects of manual microscopy [4]. In addition, reinforcement learning has potential in dynamic monitoring scenarios, such as predicting the risk of stone formation and optimizing intervention strategies by analyzing the temporal changes of patients' urine parameters in real time. In this paper, we optimize the XGBoost algorithm based on the Newton-Raphson algorithm for risk prediction of kidney stones [5].

2. Data set sources and data analysis

This paper conducts experiments using an open source dataset that can be used to predict the presence of kidney stones based on urinalysis. The dataset contains 79 urine specimens to determine if certain physical characteristics of urine are associated with the formation of calcium oxalate crystals. The six physical characteristics of urine are density of urine relative to water, negative logarithm of hydrogen ions, osmolality, conductivity , urea concentration, and calcium concentration. A partial dataset was demonstrated as shown in Table 1.

Table 1: Part of the dataset.

Gravity	Ph	Osmo	Cond	Urea	Calc	Target
1.021	4.91	725	14	443	2.45	2
1.017	5.74	577	20	296	4.49	2
1.008	7.2	321	14.9	101	2.36	2
1.011	5.51	408	12.6	224	2.15	2
1.005	6.52	187	7.5	91	1.16	2
1.02	5.27	668	25.3	252	3.34	2
1.018	5.14	703	29	272	6.63	1
1.022	5.09	736	19.8	418	8.53	1
1.025	7.9	721	23.6	301	9.04	1
1.017	4.81	410	13.3	195	0.58	1
1.018	5.14	703	29	272	6.63	1
1.022	5.09	736	19.8	418	8.53	1
1.025	7.9	721	23.6	301	9.04	1

3. Method

3.1. Vector Weighted Average Algorithm

The Newton-Raphson optimization algorithm is an efficient numerical optimization method based on second-order derivative information, and its core idea is to quickly approximate the extreme value point by local quadratic approximation of the objective function. Different from the traditional gradient descent method that only utilizes the first-order derivatives, the method achieves higher precision parameter updates in a single iteration by constructing the second-order Taylor expansion

DOI: 10.54254/2753-8818/99/2025.22531

of the objective function and simultaneously considering the function curvature information during the iteration process. Specifically, the algorithm assumes that the objective function can be approximated as a quadratic form in the vicinity of the current iteration point, and determines the search direction and step size of the next step by solving the minima of this quadratic model. This quadratic approximation property allows Newton's method to have quadratic convergence speed when approaching the optimal solution, which is significantly better than the linear convergence property of the gradient descent method [6].

The specific implementation of the algorithm is divided into three key steps: first, the gradient and Hessian matrix of the objective function at the current parameter point are calculated, then the system of linear equations derived from the quadratic approximation model is solved to determine the direction of the update, and finally the parameter update is carried out along this direction. The mathematical expression is:

$$\theta_{k+1} = \theta_k - H - {}^{1}(\theta_k)\nabla f(\theta_k)$$

where H-¹ denotes the inverse of the Hessian matrix matrix.

This updating method essentially zeroes the gradient vector by adjusting the parameters, while using the curvature information of the Hessian matrix to correct the search direction. In particular, when the Hessian matrix is positive, the update direction is bound to point to the local minima, while when it is not positive, the algorithm may converge to the saddle point or the maxima, which requires the regularization of the Hessian matrix or the use of the trust domain strategy in practical applications [7].

3.2. Vector Weighted Average Algorithm

XGBoost is an efficient and powerful integrated learning algorithm based on the gradient boosting framework, which improves model performance by combining multiple weak learners (usually decision trees) [8].The objective function of XGBoost consists of a loss function and a regularization term: the loss function measures the deviation of the predicted value from the true value; and the regularization term (L1/L2 regularization) controls the model complexity to prevent overfitting. Specifically, the regularization term consists of the L2 paradigm of the leaf node weights and the number of leaves in the tree, penalizing the complex tree structure. In the optimization process, XGBoost adopts a second-order Taylor expansion to approximate the loss function and updates the model using first-order gradient (G) and second-order gradient (H) information. Compared with the traditional GBDT using only the first-order derivatives, the second-order approximation can more accurately determine the direction and step size of the parameter update and improve the convergence efficiency. XGBoost constructs the tree structure by splitting the nodes layer by layer through the greedy algorithm [9]. The splitting gain is calculated as the difference between the loss reduction after splitting and the regularization penalty:

$$Gain = \frac{G_L^{\neg}}{H_L + \lambda} + \frac{G_R^{\neg}}{H_R + \lambda} - \frac{(G_L + G_R)^{\neg}}{H_L + H_R + \lambda} - \gamma$$

where G and H are the sum of the gradient and second-order gradient of the left and right child nodes, respectively, and λ and γ are hyperparameters controlling the weight smoothing and splitting threshold, respectively. If the gain is greater than zero, it splits, otherwise it stops growing, thus balancing the model accuracy and complexity.

XGBoost can automatically handle missing values [10]. When finding the optimal splitting point, the algorithm calculates the gain of the missing values attributed to the left or right subtree respectively, and chooses the side with greater gain as the default direction. This dynamic processing mechanism eliminates the need to pre-populate the missing values and improves data adaptation.

DOI: 10.54254/2753-8818/99/2025.22531

Although Boosting itself is a serial spanning tree, XGBoost optimizes computational efficiency by pre-sorting (Block structure) with feature parallelization. Data is sorted by features and stored as blocks, and split-point computations for different features can be executed in parallel, while cache optimization reduces the overhead of repeated data access.

3.3. XGBoost optimized based on Newton-Raphson algorithm

Newton-Raphson Algorithm OptimizationThe core principle of XGBoost is to improve the traditional first-order optimization method of gradient boosting by introducing second-order derivative information, thus accelerating the convergence and improving the model accuracy. XGBoost adopts the second-order Taylor expansion in the optimization of the objective function, which approximates the loss function as a linear combination of the first-order gradient of the current model prediction (the direction of gradient descent) and the second-order Hessian matrix (curvature information) as a linear combination to minimize the objective function more accurately. Specifically, in each iteration, XGBoost constructs statistics in terms of leaf nodes (e.g., the sum of the gradient and the sum of the Hessian) by calculating the gradient (first-order derivatives) and the Hessian (second-order derivatives) of the loss function for each sample, and calculates the splitting gain based on these statistics so as to select the optimal splitting point in the process of generating the tree structure. At the same time, the weight updates of the leaf nodes are derived from Newton's method, i.e., the weights, and the step size is adjusted by the second-order information so that the parameter updates are closer to the point of the minimal value of the loss function. This optimization strategy incorporating second-order information not only improves the model convergence speed, but also enhances the accuracy of split-point evaluation through the carving of local curvature by Hessian matrix, and effectively balances the model complexity and fitting ability by combining with the regularization term, which ultimately achieves a more efficient and accurate model training under the premise of guaranteeing generalizability.

4. Result

For parameter settings, the learning rate is set to 0.1, the maximum tree depth is set to 6, the minimum child node weight is set to 5, the sample sampling ratio is set to 0.8, the feature sampling ratio is set to 0.8, the L2 regularization coefficient is set to 1.0, the L1 regularization coefficient is set to 0.5, and the number of weak learners is set to 200. in terms of hardware configurations, we use a 16-core CPU, a 64GB RAM, a NVIDIA A100 graphics card. CUDA 11.6 driver acceleration, XGBoost 1.7.0 library, Python 3.9 interpreter.

Decision tree, random forest, XGBoost, CatBoost and our proposed Newton-Raphson algorithm to optimize the XGBoost algorithm are introduced for training respectively, and three experiments are averaged, and the model effectiveness is evaluated using accuracy, recall, precision, F1 score and AUC, and the results are shown in Table 2.

Table 2: Part of the dataset.

Model	Accuracy	Recall	Precision	F1	AUC
Decision Tree	0.708	0.708	0.716	0.711	0.708
Random Forest	0.792	0.792	0.815	0.791	0.925
XGBoost	0.792	0.792	0.819	0.793	0.825
CatBoost	0.667	0.667	0.671	0.664	0.821
Our model	0.875	0.875	0.902	0.874	0.964

DOI: 10.54254/2753-8818/99/2025.22531

The output of the confusion matrix of our model predictions is shown in Figure 1. From the confusion matrix, we can see that a total of 21 predictions in the test set are correct, by 3 predictions are wrong, and the prediction accuracy is 87.5%.

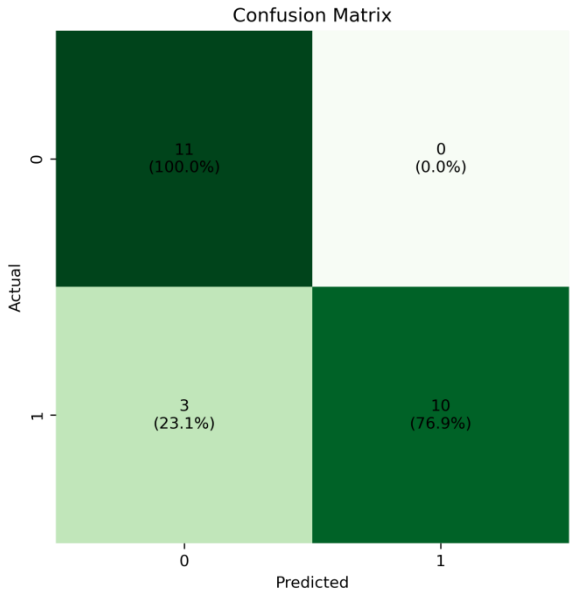

Figure 1: Confusion matrix for the test set.

The results of the comparison of the evaluation indicators for each model are shown in Figure 2.

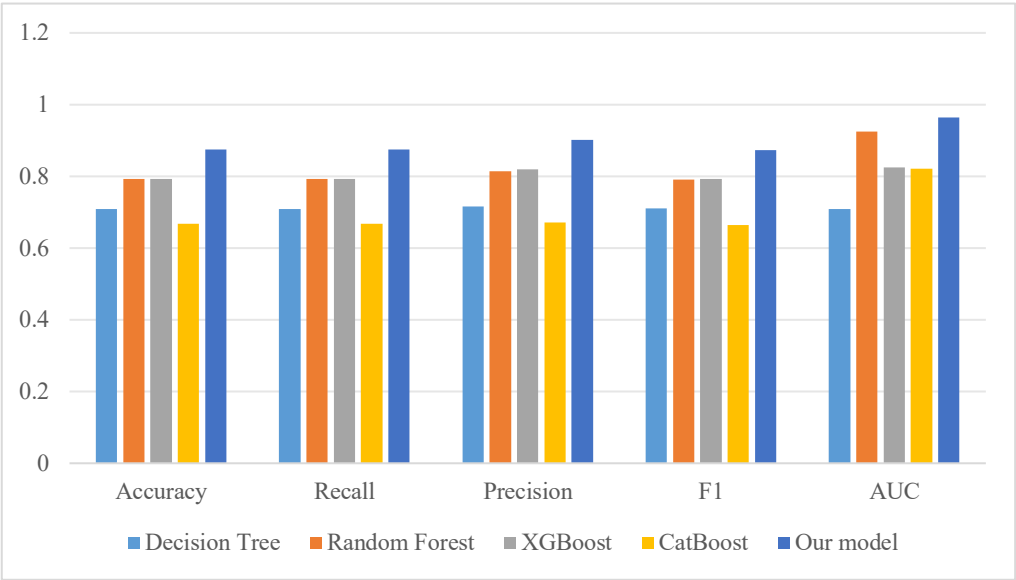

Figure 2: The results of the comparison of the evaluation indicators for each model.

The accuracy and recall of Our model (both 0.875) are significantly higher than the other models, indicating that its overall prediction ability and recognition of positive class samples are stronger. It is worth noting that the accuracy and recall of Random Forest and XGBoost are both 0.792, but there is a difference in their precision and AUC, which may be due to the difference in feature selection or integration strategies. The accuracy and recall of Our model (both 0.875) are significantly higher than the other models, indicating that its overall prediction ability and recognition of positive class samples are stronger. It is worth noting that the accuracy and recall of Random Forest and XGBoost are both

DOI: 10.54254/2753-8818/99/2025.22531

0.792, but there is a difference in the precision and AUC between them, which may stem from the difference in feature selection or integration strategies.

5. Conclusion

The accuracy and recall of Our model (both 0.875) are significantly higher than the other models, indicating that its overall prediction ability and recognition of positive class samples are stronger. It is worth noting that the accuracy and recall of Random Forest and XGBoost are both 0.792, but there is a difference in their precision and AUC, which may be due to the difference in feature selection or integration strategies. The accuracy and recall of Our model (both 0.875) are significantly higher than the other models, indicating that its overall prediction ability and recognition of positive class samples are stronger. It is worth noting that the accuracy and recall of Random Forest and XGBoost are both 0.792, but there is a difference in the precision and AUC between them, which may stem from the difference in feature selection or integration strategies.

In this study, we improve and innovate the gradient boosting framework based on the classical optimization algorithm, propose the XGBoost optimization model (NR-XGBoost) integrating the Newton-Raphson algorithm, and apply it to the healthcare data analysis scenario of kidney stone risk prediction. In order to comprehensively verify the effectiveness of the improved model, the experimental session systematically constructed a comparison model system including traditional decision tree (CART), Random Forest, standard XGBoost, and the new integrated algorithm CatBoost. The empirical results show that the NR-XGBoost model proposed in this study achieves a significant advantage of 0.875 in the two core metrics of Accuracy and Recall, which is 10.5% higher than the other comparative models, demonstrating a better overall prediction performance and the ability to recognize positive class samples. Particularly noteworthy is that although the accuracy (0.792) and recall (0.792) of Random Forest and the standard XGBoost model perform exactly the same, there are differential fluctuations of 0.024-0.031 in the Precision and the area under the ROC curve (AUC) of the two, which may originate from both the differential partitioning of the feature space by the base-learner strategy, and may also be closely related to the algorithmic specificity of the strategies such as sample sampling and feature subspace construction during the integration process.

The conclusions of this study can be summarized at three levels: at the technical level, it is confirmed that the fusion of second-order optimization algorithms and integrated learning can effectively improve the predictive performance of medical data analysis; at the methodological level, the constructed NR-XGBoost model provides a new technological pathway for processing high-dimensional medical data; and at the application level, the developed renal stone risk assessment system has the potential for clinical translation. Future research will focus on the validation of the model's generalization ability on cross-medical organization data and the development of real-time dynamic prediction system, so as to promote the in-depth application of AI technology in the field of precision medicine.

References

[1] *El-Mir, Abdulkader, et al. "Machine learning prediction of concrete compressive strength using rebound hammer test." Journal of Building Engineering 64 (2023): 105538.*

[2] *Zeng, Ziyue, et al. "Accurate prediction of concrete compressive strength based on explainable features using deep learning." Construction and Building Materials 329 (2022): 127082.*

[3] *Emad, Wael, et al. "Prediction of concrete materials compressive strength using surrogate models." Structures. Vol. 46. Elsevier, 2022.*

[4] *Liu, Kexin, et al. "Development of compressive strength prediction platform for concrete materials based on machine learning techniques." Journal of Building Engineering 80 (2023): 107977.*

[5] *Chi, Lin, et al. "Machine learning prediction of compressive strength of concrete with resistivity modification." Materials Today Communications 36 (2023): 106470.*

DOI: 10.54254/2753-8818/99/2025.22531

[6] *Ghunimat, Dalin, et al. "Prediction of concrete compressive strength with GGBFS and fly ash using multilayer perceptron algorithm, random forest regression and k-nearest neighbor regression." Asian Journal of Civil Engineering 24.1 (2023): 169-177.*

[7] *Liu, Gaoyang, and Bochao Sun. "Concrete compressive strength prediction using an explainable boosting machine model." Case Studies in Construction Materials 18 (2023): e01845.*

[8] *Liu, Kexin, et al. "Development of compressive strength prediction platform for concrete materials based on machine learning techniques." Journal of Building Engineering 80 (2023): 107977.*

[9] *Li, Hong, et al. "Compressive strength prediction of basalt fiber reinforced concrete via random forest algorithm." Materials Today Communications 30 (2022): 103117.*

[10] *Imran, Hamza, et al. "Latest concrete materials dataset and ensemble prediction model for concrete compressive strength containing RCA and GGBFS materials." Construction and Building Materials 325 (2022): 126525.*