Progress in Medical Devices
ISSN: 2957-5478

Zentime

Research Article https://doi.org/10.61189/846307fkxccq

Analysis of urinary non-formed components at home based on machine learning algorithms

Yifei Bai, Rongguo Yan, Yuqing Yang, Chengang Mao

School of Health Science and Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China.

Corresponding author: Rongguo Yan.

Declaration of conflict of interest: None.

Received April 12, 2024; Accepted July 11, 2024; Published September 30, 2024

Highlights

- The study evaluated five machine learning algorithms in analyzing urinary non-formed components. Among them, the Random Forests model demonstrated the highest accuracy, precision, recall, and F1 score, suggesting its effectiveness in analyzing urinary non-formed components.
- A technological innovation is introduced for home urinalysis, offering the potential to enhance medical efficiency and patient experience.

Abstract

Objective: Machine learning can automatically extract valuable insights from vast datasets, predict and classify diseases, and evaluate drug efficacy. To assess the effectiveness of machine learning algorithms in analyzing non-formed components in urine, real medical data were processed and annotated. **Methods:** Five models, including K-Nearest Neighbors, Decision Trees, Random Forests, Support Vector Machines, and Gaussian distributions, were constructed to quantitatively analyze 12 non-formed urine components, such as vitamin C, white blood cells, and urinary bilirubin. The efficacy of these models was then compared. **Results:** It was found that the Random Forest model outperformed others, achieving the lowest mean squared error, high recall rate, accuracy, and area under the curve. **Conclusions:** These findings indicate that machine learning offers significant potential for studying non-formed urine components, potentially enhancing the precision and effectiveness of disease detection and providing valuable support for clinical decision-making.

Keywords: Home urine component analysis, machine learning models, quantitative results

Introduction

Urine composition analysis is a widely utilized medical diagnostic technique that accurately reflects various physiological characteristics, such as bilirubin, protein, and glucose levels [1]. Beyond its application in diagnosing and monitoring diseases of the kidneys and urinary tract, urine dry chemical analysis is crucial for assessing individual health, drug usage, auxiliary diagnosis, and monitoring occupational diseases. It also plays a significant role in diagnosing conditions affecting other organs. However, traditional urine analysis often proves costly and impractical for home use due to the need for specific tools and laboratory settings [2].

The medical industry has extensively leveraged machine learning, particularly in areas such as wearable sensors, medicinal chemistry, brain and cancer research, and medical imaging. The unique strength of machine learning algorithms lies in their ability to analyze large datasets to uncover patterns, elucidate underlying phenomena, and identify correlations, which can significantly aid medical professionals in better understanding and diagnosing patients [3]. For many diseases, machine learning-based approaches have shown potential to enhance the performance, accuracy, predictability, and reliability of diagnostic systems.

Address correspondence to: Rongguo Yan, School of Health Sciences and Engineering, University of Shanghai for Science and Technology, No.516 Jungong Road, Shanghai 200093, China. E-mail: yanrongguo@usst.edu.cn.

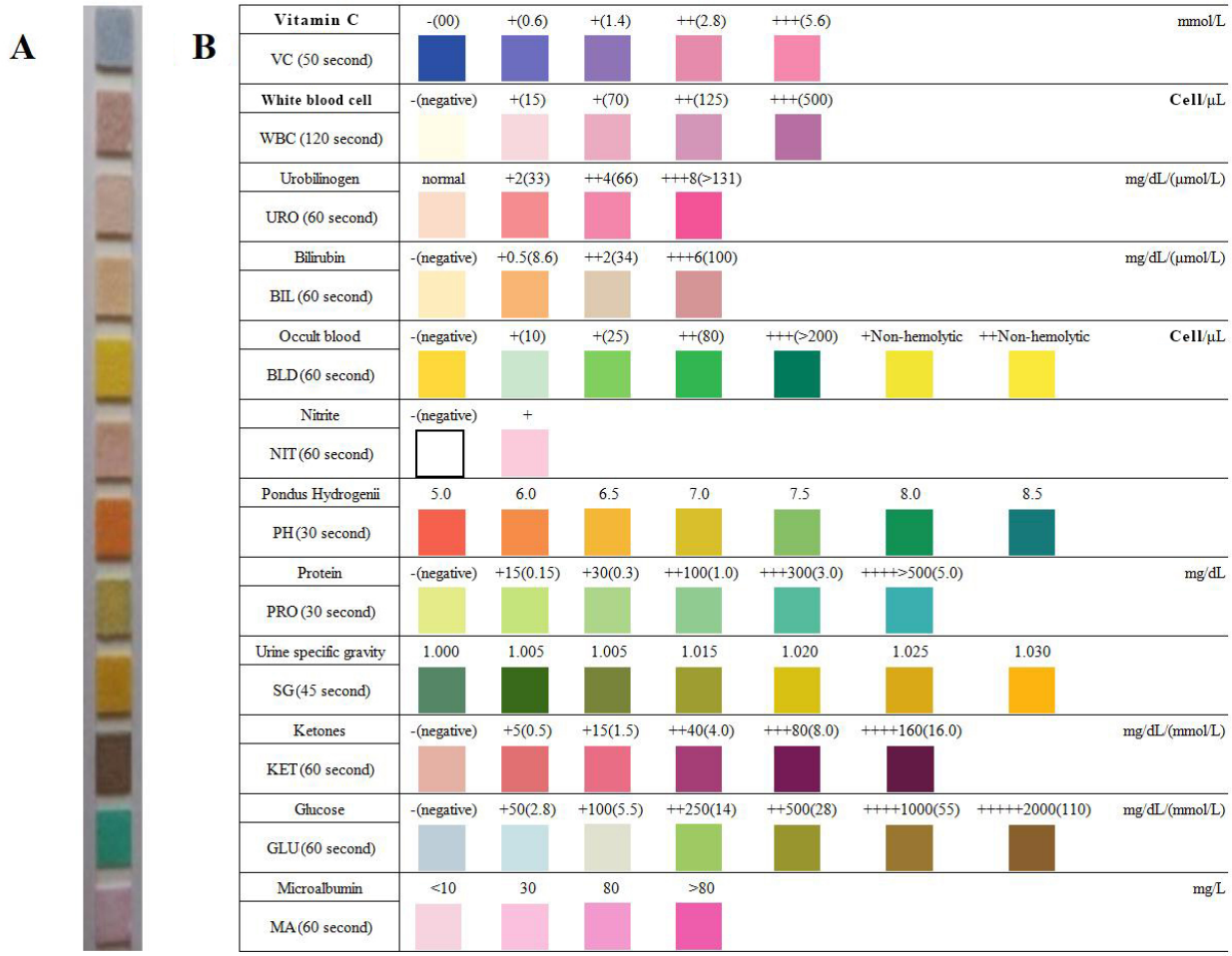

Figure 1. URIT 12-pads Test strips (A) and its color charts (B).

The field of urine analysis and related domains offer a broad array of applications for machine learning. Technological advancements in color recognition, sensor technology, and related fields have significantly enhanced the analysis of urine components. Moreover, research efforts aimed at improving user experience, portability, and the development of intelligent analytical systems hold potential to make urine detection technology more practical and widely applicable [4-6].

This study processed and annotated real medical data to develop models based on K-Nearest Neighbors (KNN), Decision Trees, Random Forests, Support Vector Machines (SVM), and Gaussian distributions. These models were used for the quantitative analysis of 12 non-formed components in urine, including vitamin C, white blood cells, urobilinogen, etc. The primary objective was to investigate the efficacy of machine learning algorithms in analyzing urinary non-formed components. Additionally, the study compared the recognition capabilities of the five models.

Material and methods

Principles of urine dry chemistry test

Urine analysis using the dry-chemistry strip method is the most prevalent detection technique, as depicted in **Figure 1**. In this method, a urine sample is directly applied to a dry test strip [7, 8]. The water content of the urine acts as the solvent, causing specific elements in the urine to chemically react with materials in the test strip pad, resulting in a color change. Darker shades indicate higher concentrations of the tested components, establishing a correlation between color intensity and component concentration [9]. Quantitative analysis of these non-formed components was conducted by comparing the color-changed test strip with a urine analysis color chart, as illustrated in **Figure 2**. The simplicity and rapidity of dry chemistry urine analysis make it a favored option for home use, as well as in clinics and other medical settings.

Principles of machine learning algorithm

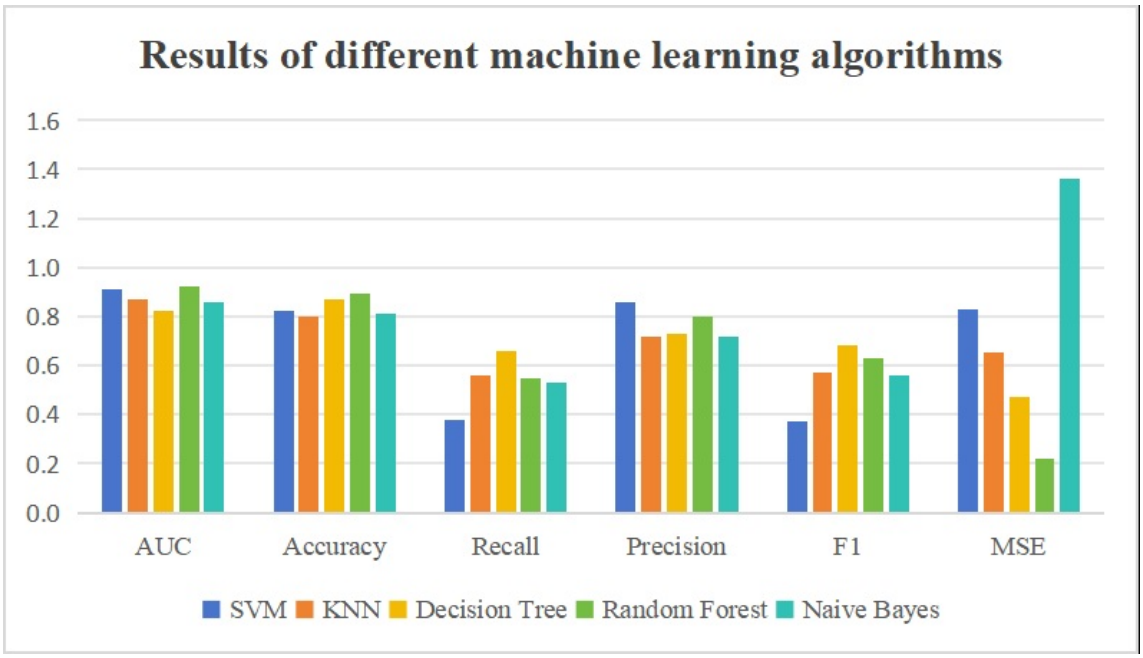

Figure 2. Comparison of the results of different Machine Learning algorithm. SVM, Support Vector Machines; KNN, K-Nearest Neighbors; AUC, Area Under the Curve; MSE, Mean Squared Error.

KNN

The KNN technique is a widely employed method for both regression and classification tasks. Central to KNN is the premise that a sample is likely to belong to a certain class if the majority of its KNN in the feature space are members of that same class [10].

The KNN algorithm operates through several steps. Initially, it constructs a feature space model using labeled training examples. Then, it identifies the k training samples closest to the new, unlabeled sample within this feature space. Based on the labels of these k neighbors, the algorithm predicts the classification or regression outcome for the new sample [11].

The KNN algorithm is valued for its simplicity and ease of understanding, along with its effectiveness in handling non-linear problems. However, it has some drawbacks, particularly when applied to large datasets. The algorithm is sensitive to noise and outliers in the feature space, which can adversely affect its accuracy. Additionally, it is computationally demanding, as it involves calculating the distance between samples for each query, which increases the computational complexity.

Decision tree

A popular method for constructing decision trees is known as Classification and Regression Trees. This approach uses a greedy algorithm to split data at each node, aiming to maximize information gain or minimize Gini impurity [12]. The algorithm continues to divide the data into two subsets until a stopping condition is met, such as reaching the maximum tree depth or having only a minimal number of samples at a node. The result of this process is a binary tree structure, where each leaf node represents a class label or prediction value, and each internal node corresponds to a feature.

Classification and Regression Trees algorithms are highly effective due to their ability to handle both numerical and categorical data, and they provide a straightforward method for visualizing and interpreting the correlations between features and outcomes. However, they are prone to overfitting. This issue can be mitigated with techniques such as pruning, which helps to simplify the model and improve its generalizability by removing sections of the tree that provide little predictive power.

SVM

SVM is a robust supervised learning method effectively used for both binary and multiclass classification challenges. Based on the principles of the perceptron model, SVM seeks to find a line (in two-dimensional space) or a hyperplane (in three-dimensional or higher-dimensional spaces) that optimally separates data points of different classes [13]. The ideal hyperplane is the one that maximizes the margin between these classes, with the most significant data points, known as support vectors, defining this decision boundary.

For multi-class classification tasks, SVM can be implemented using two primary approaches: the "one-versus-rest" (also known as "one-versus-all") and the "one-versus-one" (also referred to as "pairwise") methods [14, 15]. In the "one-versus-rest" approach, an SVM model is trained for each class, where the specific class is treated as the positive class and all others are combined into a negative class. This requires training multiple SVM models—one for each class. On the other hand, the "one-versus-one" method involves training an SVM model for every possible pair of classes, resulting in n(n-1)/2 models for n classes. To ascertain the final classification, a voting or scoring mechanism is used, where the class that receives the highest score from the SVM models is selected.

In this study, we utilized 8,036 sets of color data from urinalysis multi-test strips to train an SVM-based color classification algorithm employing the "one-versus-rest" approach. We designated 1,004 sets of color data for testing and an additional 1,004 sets for validation of the method. The polynomial kernel function was chosen for the SVM model to optimize the classification process.

Random forest

Random Forest is an ensemble learning technique that constructs multiple decision trees and aggregates their predictions through voting or averaging. Each tree is developed using bootstrap samples from the original dataset, involving random sampling with replacement [16]. Additionally, during the construction of each tree, a random subset of features is selected for node splitting. This means that each node decision is made based on a limited subset of features, rather than the entire feature set. This strategy not only increases the diversity among the trees but also helps reduce overfitting, thereby enhancing the robustness and overall performance of the model. Random Forest is particularly effective for both classification and regression tasks, capable of managing high-dimensional data and providing valuable insights into feature importance [17, 18].

Gaussian naive bayes

Gaussian Naive Bayes is a specialized form of the Naive Bayes classifier designed to handle continuous data, assuming that the distribution of features follows a Gaussian (normal) distribution. This forms part of the broader Naive Bayes algorithmic framework, which posits that each feature is independent given the class label [19, 20]. Although this independence assumption may not always align with real-world data complexities, it simplifies the computational process significantly. Despite its simplicity, Gaussian Naive Bayes can perform effectively across a variety of practical applications.

Analysis of urinary non-formed components based on machine learning algorithms

Data collection

In this study, the standard urine poly-reagent band was photographed, and its color was extracted using a smartphone. Due to the limited amount of standard data available, the color data of the collected standard urine poly-reagent band was expanded using the interpolation method, resulting in a comprehensive standard color dataset consisting of 1908 entries. Additionally, actual urine color information from 935 patients was obtained from the hospital to further enhance the model's functionality.

Data cleaning

In the data cleaning process, several key steps need to be considered:

• Handling Missing Values: Various techniques can be employed to address missing values, depending on the rate of missing data and its significance. These techniques include eliminating fields, imputing missing values, or recollecting data [21]. Approaches for filling in missing information can utilize the mean, median, quantiles, mode, random values, interpolation, and other methods. Additionally, business knowledge or experience can guide this process. Virtual variables that map to high-dimensional space can also be introduced, or a model for missing data prediction can be constructed.

• Format and Content Processing: Ensuring the correctness and integrity of data formats and content is crucial for data processing. This involves transforming, purifying, and standardizing the data in an appropriate manner to facilitate further analysis and modeling.

• Duplicate Data Removal: Removing redundant entries helps prevent their influence on the model. This can be achieved by identifying and eliminating duplicate data from the dataset through the comparison of content or record identifiers.

• Noise Data Handling: Anomalous or erroneous data points are referred to as noise data. An excessive amount of noisy data can hinder

the model's ability to generalize, while an appropriate amount of noise data can help avoid overfitting. When working with noisy data, it is important to assess the situation and handle noise in data by data cleaning, outlier detection, and smoothing.

Feature extraction

This study selected 11 features, including brightness, contrast, saturation, color distance etc., and performed feature extraction on the dataset [22-25]. Some of the formulas used are as follows:

$$\text{brightness} = \frac{R+G+B}{3} \quad (1)$$

Relative brightness:

$$l1 = 0.2126 * \mathrm{R} + 0.7152 * \mathrm{G} + 0.0722 * \mathrm{B} \quad (2)$$

$$l2 = 0.2126 + 0.7152 + 0.0722 \quad (3)$$

$$\text{contrast} = \frac{l1+0.05}{l2+0.05} \quad (4)$$

$$\text{saturation} = \frac{X}{R+B+G} \quad (5)$$

X is the maximum value among R, G, and B.

$$\text{color temperature} = \frac{R}{B}(B \neq 0) \quad (6)$$

$$\text{color distance} = \sqrt{(1-R)^2 + (1-G)^2 + (1-B)^2} \quad (7)$$

Labeling

The data were labeled based on the detection outcomes corresponding to the input data using Arabic numerals to ensure compliance with the required data format for machine learning model training. The correlation between the label numbers and the respective detection outcomes was recorded for future reference.

Training, validation, and testing sets

This study employed a random sampling approach to allocate the data into training, validation, and testing sets. Specifically, 80% of the data are designated as training samples, 10% as validation samples, and the remaining 10% as test samples.

Results

Assessment methods

Seven metrics were calculated using the validation set to assess the models' performance:

• Area Under the Curve (AUC): The accuracy of the classifier in multi-class classification issues was indicated by the area under the ROC curve. Higher AUC values indicate greater classifier performance [26].

• Accuracy: The ratio of the number of samples correctly predicted by the model to the total number of samples. The accuracy was calculated as follows [27]:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (8)$$

The number of positive examples that the model correctly predicted is represented by TP (True Positive), the number of negative examples that the model correctly predicted is represented by TN (True Negative), the number of negative examples that the model incorrectly predicted as positive examples is represented by FP (False Positive), and the number of positive examples that the model incorrectly predicted as negative examples is represented by FN (False Negative).

• Mean Squared Error (MSE): The average of the squared differences between the model's predictions and the true values [28].

• Confusion Matrix: A matrix used to evaluate the accuracy of a classification model, containing the metrics of TP, TN, FP, and FN [29].

• Recall Rate (R): The proportion of positive examples correctly predicted by the model among all positive examples. The Recall Rate was calculated as follows [30]:

$$R = \frac{TP}{TP+FN} \quad (9)$$

• Precision Rate (P): The proportion of true positive examples among all samples predicted as positive by the model. The Precision Rate was calculated as follows [31]:

$$P = \frac{TP}{TP+FP} \quad (10)$$

• F1 Score: The harmonic average of the recall rate R and the precision rate P, providing a weighted average that combines the accuracy of the model. The F1 Score was calculated as follows [32, 33]:

Table 1. Results of different Machine Learning algorithms

	AUC	Accuracy	Recall	Precision	F1	MSE
SVM	0.91	0.82	0.38	0.86	0.37	0.83
KNN	0.87	0.80	0.56	0.72	0.57	0.65
Decision Tree	0.82	0.87	0.66	0.73	0.68	0.47
Random Forest	0.92	0.89	0.55	0.80	0.63	0.22
Naive Bayes	0.86	0.81	0.53	0.72	0.56	1.36

Note: SVM, Support Vector Machines; KNN, K-Nearest Neighbors; AUC, Area Under the Curve; MSE, Mean Squared Error.

$$F1 = \frac{2 \times P \times R}{P + R} \quad (11)$$

Discussion

Comparative Analysis of Results

The performance of the five Machine Learning algorithms employed in this study are compared based on the results presented in **Table 1**.

• AUC: The Random Forest and SVM models exhibit superior AUC scores of 0.92 and 0.91, respectively, outperforming the KNN and Gaussian Naive Bayes models, which scored 0.87 and 0.86, respectively. The Decision Tree model has the lowest AUC score of 0.82.
• Accuracy Rate: The Random Forest model achieves the highest accuracy of 0.89, followed by the Decision Tree and SVM models with 0.87 and 0.82 accuracy, respectively. The Gaussian Naive Bayes and KNN models have lower accuracy rates, both around 0.80.

• Recall Rate: The Decision Tree model has the highest recall rate of 0.66, followed by the KNN and Random Forest models with 0.56 and 0.55, respectively. The Gaussian Naive Bayes and SVM models have recall rates of 0.53 and 0.38, respectively.

• Precision Rate: The SVM model attains the highest precision score of 0.86, followed by the Decision Tree and Random Forest models with 0.80 and 0.73, respectively. The Gaussian Naive Bayes and KNN models have relatively lower precision scores, both around 0.72.

• F1 Score: The Decision Tree and Random Forest models demonstrate better F1 scores of 0.68 and 0.63, respectively, while the KNN and Gaussian Naive Bayes models have lower F1 scores of 0.57 and 0.56, respectively. The SVM model has the lowest F1 score of 0.37.

• MSE: The Random Forest model has the lowest MSE of 0.22, indicating the best performance. The Decision Tree and KNN models have MSEs of 0.47 and 0.65, respectively, while the SVM and Gaussian Naive Bayes models have higher MSEs of 0.83 and 1.36, respectively.

Error analysis of the test results

The Random Forest model exhibited the strongest performance on the validation set, achieving high accuracy and reliability in detecting non-formed components in urine samples. This success can be attributed to the model's ability to handle complex relationships within the data and reduce variance through ensemble learning techniques [34].

The SVM model demonstrated good precision and AUC scores but had relatively lower recall and accuracy. This could be due to the sensitivity of SVM performance to the choice of kernel function and the optimization of hyperparameters, which can be challenging when dealing with nonlinear data [35].

The KNN model performed comparably to the SVM model, but offered a slight advantage in terms of recall and F1 Score. The number of neighbors and the distance measure used in the KNN algorithm can significantly influence its performance, depending on the characteristics of the dataset [36].

The Decision Tree model outperformed the Random Forest and SVM models in terms of recall and F1 Score, while also demonstrating favorable accuracy and MSE. The performance of the Decision Tree model can be influenced by factors such as feature selection and the presence of noise in the dataset [37].

The Gaussian Naive Bayes model showed the poorest performance, particularly in terms of recall and precision. This can be attributed to the model's underlying assumption of feature independence, which may not hold true for the complex relationships present in the urine data [38].

Conclusions

To summarize the findings, in conjunction with the insights from **Figure 2**, the following conclusions can be drawn:

• The Random Forest model demonstrated distinct advantages in terms of prediction performance and generalization capacity. It achieved the best results across several key metrics, including AUC, accuracy, and MSE, indicating its strong overall predictive ability.

• The Decision Tree model exhibited strong performance in terms of recall rate and F1 score, suggesting that it is particularly effective at capturing and leveraging important information within the data.

• The SVM model achieved the highest precision among the tested algorithms. However, it performed relatively poorly in terms of recall and F1 score, indicating that further parameter tuning and optimization may be necessary to enhance its overall effectiveness.

• While the Gaussian Naive Bayes and KNN models did not demonstrate exceptional performance, they exhibited relatively steady and consistent results across a range of evaluation metrics.

• These observed differences in model performance can be primarily attributed to the varying adaptability of the different algorithms to the specific characteristics and complexity of the data.

Author Contributions: Yifei Bai performed data acquisition, processing, analysis and model building and wrote the manuscript. Yifei Bai, Rongguo Yan, Yuqing Yang, and Chengang Mao made substantial contributions to data analysis and manuscript preparation.

References

[1] Yan RG, Wei RF, Shen JK, et al. Research progress on color determination methods of reagent strips in routine urinalysis. Int J Biomed Eng 2019;42:532-536.

[2] Ryan D, Robards K, Prenzler PD, et al. Recent and potential developments in the analysis of urine: a review. Anal Chim Acta 2011;684(1-2):8-20.

[3] Shehab M, Abualigah L, Shambour Q, et al. Machine learning in medical applications: A review of state-of-the-art methods. Comput Biol Med 2022;145:105458.

[4] Wang JW. Design of a portable urine analyzer. Chongqing University 2020.

[5] Wang F. Research and design of urine analyzer based on color sensor. Jilin University 2014.

[6] Liu G, Hu N, Ma Z, et al. A portable analyzer based on a novel optical structure for urine dry-chemistry analysis. J Instrum 2018;13(07):T07002.

[7] Cui LS. Research on intelligent analysis system for home urine detection. Southeast University 2022.

[8] Li XY. Clinical value of combined use of urine dry chemistry test method and urine sediment test method for routine urine testing. Chin Mod Drug Appl 2019;13(15):78-80.

[9] Li YY, Yan GR, Qiu XY, et al. Application of FIA algorithm in urine component detection. Software 2018;39(06):40-45.

[10] Liu XN, Wang K, Wang CD, et al. Research on identification of college students with difficulties based on KNN algorithm. Electron. Compon Inf Technol 2023;7(08):88-92.

[11] Jawthari M, Stoffova V. Predicting students' academic performance using a modified kNN algorithm. Pollack Periodica 2021;16(3):20-26.

[12] Ju MX, Zhou YX, Tang WN, et al. CART Decision Tree power consumption anomaly detection method based on data imbalance. Jilin Electr Power 2024;52(01):53-56.

[13] Li ZT. Development history and application of Support Vector Machines. Inf Syst Eng 2024;(03):124-126.

[14] Xu JH. An extended one-versus-rest support vector machine for multi-label classification. Neurocomputing 2011;74(17):3114-3124.

[15] Guo RN. Ceramic raw material component identification based on Support Vector Machine. Sci Technol Innovation 2024;(02):32-35.

[16] Guo ZY, Lao ZK, Chen YF, et al. Application of Random Forest algorithm in prediction of droplet diameter in electrospray printing. Mech Des Manuf 2024;(03):153-157.

[17] Liu HX, Li X, Ding JG, et al. Strip thickness prediction based on Random Forest feature selection and genetic algorithm optimization. Steel Rolling 1-16.

[18] Tin Kam H, Hull JJ, Srihari SN. Decision combination in multiple classifier systems. IEEE Trans Pattern Anal Mach Intell 1994;16(1):66-

75.
[19] Wang SH, Gao R, Du RJ. Dependency extension of naive Bayesian classifiers based on Gaussian kernel function. Control Decis 2015;30(12):2280-2284.
[20] Bao QQ, Sun CR. An improved naive Bayes method applied to intelligent triage. Mod Hosp 2024;24(03):424-427.
[21] Li JF, Zhang JY, Li LK, et al. Discussion on processing methods of missing values in clinical trials. J Peking Union Med Coll 1-15.
[22] Shi KW. Neural basis of color brightness at different time frequencies. Shanghai University of Sport 2023.
[23] Chen LP, Zhou H, Zhang NY, et al. Color image segmentation method based on improved convex hull and color contrast. J Hebei Univ, Nat Sci Ed 2018;38(05):543-548.
[24] Ibraheem NA, Hasan MM, Khan RZ, et al. Understanding Color Models: A Review. ARPN J Sci Technol 2012.
[25] Guo JJ, Li QW, Cheng HL, et al. Tree image segmentation algorithm based on Lab color distance and GMM. Inf Technol 2016;(02):1-4+9.
[26] Liu BW, Bu Y, Zou DH, et al. An improved tooth color classification model based on residual network. Software Eng 2024;27(03):52-57.
[27] Liu JQ, Zhong SH, Li SZ, et al. patite source rock identification method based on Machine Learning. Geotectonics Miner 1-14.
[28] Liu AL. Research on obesity level estimation based on Machine Learning. Shaoxing University 2023.
[29] Jin M. Research on pig behavior identification and classification methods based on Machine Learning. Shaoxing University of Arts and Sciences 2021.
[30] Zhang DS. Image classification algorithm based on parameterized quantum circuits and Machine Learning. Nanchang University 2023.
[31] Hu T, Wu MT, Hu W, et al. Classification prediction of fly ash activity based on Machine Learning. J. Cent South Univ (Sci Technol) 2023;54(10):3829-3839.
[32] Huang KL. Research on voice fraud-related behavior based on Machine Learning. University of International Business and Economics 2017.
[33] Wang ZG, Zhang HY, Miao DQ. Automatic selection method of non-maximum suppression threshold based on F1 valu. J Intell Syst 2020;15(05):1006-1012.
[34] Cheng L, Chen XW, De Vos J, et al. Applying a random forest method approach to model travel mode choice behavior. Travel Behav Soc 2019;14:1-10.
[35] Azadeh A, Saberi M, Kazem A, et al. A flexible algorithm for fault diagnosis in a centrifugal pump with corrupted data and noise based on ANN and support vector machine with hyper-parameters optimization. Applied Soft Computing 2013;13(3):1478-1485.
[36] Abu Alfeilat HA, Hassanat ABA, Lasassmeh O, et al. Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review. Big Data 2019;7(4):221-248.
[37] Galathiya A, Ganatra A, Bhensdadia C. Improved Decision Tree Induction Algorithm with Feature Selection, Cross Validation, Model Complexity and Reduced Error Pruning. Int J Comput Sci Inf Technol 2012;3(2):3427-3431.
[38] Khalid H, Khan A, Zahid Khan M, et al. Machine Learning Hybrid Model for the Prediction of Chronic Kidney Disease. Comput Intell Neurosci 2023;2023:9266889.