Neural Computing and Applications (2023) 35:14487–14497
https://doi.org/10.1007/s00521-021-06060-0

S.I: ML4BD_SHS

A hybrid machine learning approach for hypertension risk prediction

Min Fang[1] · Yingru Chen[2] · Rui Xue[1] · Huihui Wang[3] · Nilesh Chakraborty[4] · Ting Su[1] · Yuyan Dai[4]

Received: 6 December 2020 / Accepted: 19 April 2021 / Published online: 20 May 2021

Abstract
Hypertension is a primary or contributing cause for premature death in the entire world. As a matter of fact, there is a high prevalence and low control rates in low- and middle-income countries, such that the prevention and treatment of hypertension should remain a top priority in global health. In the recent years, the awareness, treatment, and control rates of hypertension patients in China have been significantly improved to 51.6%, 45.8%, and 16.8%, respectively. However, those rates are still far from a satisfactory level. Clinical studies suggest that for people in the pre-clinical stage of hypertension or having the risk of hypertension, the progression of the disease may be significanly reduced through a change in lifestyle, or by an effective drug therapy. In this paper, we address risk prediction for hypertension in the next five years, and put forward a model merging KNN and LightGBM. Our approach allows us to predict the hypertension risk for a specific individual using features such as the age of the subject and blood indicators. Results shows that our model is reliable and achieves accuracy and recall rate over 86% and 92%, respectively.

Keywords Hypertension · Boosting · Predictive models · Data analysis · Hybrid models

1 Introduction

Hypertension is a relevant global health challenge due to its high prevalence, and to the corresponding cardiovascular disease and chronic kidney disease [1].

Tackling this disease involves the use of a great amount of medical resources, which influences and are influenced by the economic status of a family or country. The awareness, treatment, and control rates (crude rates) of hypertensive patients are increased in China in the recent years, reaching 51.6%, 45.8% and 16.8%, respectively [2]. However, the current status is still far from a satisfactory level.

✉ Min Fang
fangmin@hit.edu.cn

Yingru Chen
2172272004@email.edu.szu.cn

Rui Xue
xuerui@hit.edu.cn

Huihui Wang
h.wang@ieee.org

Nilesh Chakraborty
nilesh@szu.edu.cn

Ting Su
suting@hit.edu.cn

Yuyan Dai
2070276029@email.szu.edu.cn

[1] Education Center of Experiments and Innovations, Harbin Institute of Technology (ShenZhen), Shenzhen 518055, China

[2] Huazhong University of Science and Technology Union Shenzhen Hospital (Nanshan Hospital), Shenzhen 518051, China

[3] Cybersecurity Program, St. Bonaventure University, New York, NY 14778, USA

[4] College of Computer Science and Engineering, Shenzhen University, Shenzhen 518060, China

Hypertension may be prevented and may be well controlled. Epidemiological research has firmly established that there are four main risk factors for hypertension, i.e. improper diet, addiction to tobacco and/or alcohol, lack of activity, and emotional stress. Therefore, correcting bad habits and behaviors related to these four aspects, would naturally reduce the occurrence of hypertension, also achieving the task of preventing hypertension [2–4]. In turn, it would of great help to assess the risk of hypertension earlier than the clinical developments, so that healthcare/counselling facilities may be provided to individuals in order to correct the bad habits and behaviors.

In the recent years, machine learning (ML) found several applications in diverse fields, including finance, materials science, environment, and so on. ML has proved very effective in solving the difficult problems in various domains [5–16]. In the medical field, researchers are using ML techniques for analyzing neuroimaging (neuroimaging data analysis) [17], detecting mycobacterium tuberculosis (MTB) resistent to the existing TB drug [18], understanding the early stage of Parkinson's disease from voice data [19], and many others.

Inspired by the success of ML techniques in the medical field, we address ML as a tool to predict the risk of hypertension. Our prediction is based on features such as age and blood indicators. We ave conducted experiments on a dataset of more than 30000 records provided by a local hospital. Experimental results reveal that our model, which is based on a combination of KNN and LightGBM algortihms, achieves high accuracy and recall rate (at least 86% and 92%, respectively).

2 Related work

Researchers worlwide are committed to establish hypertension risk prediction models using different data and different machine learning algorithms.

Simple anthropometric data, e.g those related to obesity in Korean adults, have been used in connection with ML models to predict hypertension. Results show that waist circumference (WC) is inded a useful tool to predict the incidence of hypertension. WC is especially useful for young populations, where it represents a more sensitive predictor of hypertension compared to the elderly [20].

Body mass index (BMI), WC, hip circumference (HC), waist-to-hip ratio (WHR) and other data of male subjects have been also used to establish a random forest model to predict hypertension, achieving an accuracy rate of about 76% [21].

Data collected by a behavioral risk factor Monitoring system (BRFSS) have been analyzed using binary Logistic regression model to select the factors having a statistical significance for hypertension, according to a pre-defined P value. A MLP neural network model based on the BP algorithm has been then designed and trained with the selected risk factors to predict hypertension. The authors declare that the model is able to achieve 72% accuracy in predicting the diagnosis of hypertension [22].

Cardio-ankle vascular index (CAVI) has been considered as an indicator and models have been built by providing data to models using machine learning methods (XGBoost and ensemble) or traditional statistical methods (logistic regression). If applied to verification data, the Area Under Curve (AUC) quantifier of the above three models are 0.877, 0.881 and 0.859, respectively [23].

More detailed studies also consider additional factors in building a prediction model, such as family history, smoking, high-salt diet, diabetes, dyslipidemia and physical exercise [24].

Our approach differs from most of the current ones in two aspects. At first, we employ age and blood indicators as the relevant features to consider. Then, we suggest a hybrid model, which combines KNN and LightGBM to improve prediction accuracy.

3 Proposed methodology

3.1 Dataset description

The hypertension data have been collected from the Electronic Health Record (EHR) database of the Huazhong University of Science and Technology, Union Shenzhen Hospital. There are 33,289 records in this dataset, which have been anonymized by removing attributes as the Name, ID, and so on. The anonymized dataset has 23 attributes (e.g. age, blood indicators, see Table 1), which we use as input features. The dataset includes patients data collected from Jan. 2012 to Dec. 2014.

Remarkably, none of the participants had hypertension during the collection of data. In the following phase of the study (conducted from Jan. 2015 to Dec. 2017), the dataset evolved as follows: 17,074 participants (51.3%) reported no-hypertension, marked with label 0, and 16,215 participants (48.7%) reported hypertension, marked with label 1. Fig. 1a shows the distribution of the labelled data, which is roughly balanced. The number of positive (label 1) and negative (label 0) samples in each age group is illustrated in Fig. 1b. The number of young (aged 0–20) and old people (over 80) in the sample is small.

Clinical trials have shown that for subject with no hypertension, but showing pre-hypertension or with risk factors, early prevention by drug therapy of changes in the lifestyle may reduce risk both in the short and medium to long term [25–27]. In our work, using the above dataset, we

Table 1 The features after preprocessing

Feature	Mean	Standard deviation
Age	51.4	16.026
Number of neutrophils	4.652	2.184
Number of lymphocytes	2.029	0.685
Number of eosinophils	0.16	0.14
Number of basophils	0.013	0.019
Total protein	69.151	7.421
Albumin	42.608	6.296
Globulin	26.543	3.98
Total bilirubin	11.179	6.559
Direct bilirubin	3.128	2.458
Potassium	4.056	0.387
Sodium	141.334	2.367
Calcium	2.309	0.145
Urea nitrogen	5.273	3.092
AST/ALT	1.138	0.634
Triglyceride	1.977	1.57
High density lipoprotein	1.302	0.326
Low density lipoprotein	2.947	0.859
Average red blood cell	92.165	6.295
Thrombin time	12.735	1.372
International normalized ratio	0.999	0.132
Activated partial thromboplastin time	32.961	3.934
Fibrinogen	3.653	0.642

focus on developing a model for predicting hypertension in the next 5 years.

3.2 Data preprocessing

3.2.1 Remove outliers

There are no missing values in our data, so there is no need for padding. However, we notice that there are negative values for the number of basophils, urea nitrogen and triglycerides (as shown in the Table 2), which are not admissible. Since there are only few records with negative values, we just drop those subjects with inconsistent data.

In addition, we notice that there are some records where age, number of eosinophils, number of basophils or activated partial thromboplastin time are zero (see Table 3). For the number of eosinophils or the number of basophils zero is an acceptable values, whereas this is of course not the case for age, and also for activated partial thromboplastin time. Also in this case, there are only few records with these anomalies and we just delete those records.

After the above data processing, we are left with 33255 records in the dataset. There are 16205 positive samples, accounting for 48.7% of the total samples, and 17050 negative samples, corresponding to 51.3%. The dataset is thus roughly balanced (see Fig. 2).

3.2.2 Data normalization

Data normalization may increase the speed of gradient descent methods in seeking for the optimal solution, and make results obtained for different dimensions more comparable each other. In other words, data normalization improves accuracy of classifiers. In our experiments, we use Standard Scaler to preprocess the dataset. The mean and the standard deviation of each feature are evaluated separately, and then the data are normalized The score of a sample x may be obtained as

$$z = (x - u)/s, \tag{1}$$

where u is the mean and s is the standard deviation of the training samples, respectively.

Figure 3 reports the original data distribution for age and low-density lipoprotein. In the original data, age lies in the range (0,100), while the low-density lipoprotein is in the interval (0.01,15.53). The numerical range of these two features differs greatly.

Figure 4 shows the data distribution of the two features after normalization. As it is apparent from the plot, the ranges of the two features are roughly the same, normalization may indeed make different features more comparable.

3.3 Experiment and evaluation

3.3.1 Data preparation and performance metric

We split the dataset into training and test data using a ratio 8:2. Training data are used for training and for validation, while test data are used for final evaluation.

In order to assess the performance of the model, we employ Accuracy, Precision, Recall, and F1 score. They are defined as follows

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{FN} + \text{FP} + \text{TN}), \tag{2}$$

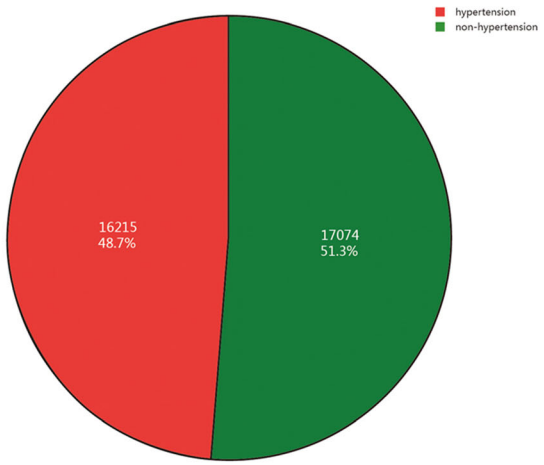
$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}), \tag{3}$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}), \tag{4}$$

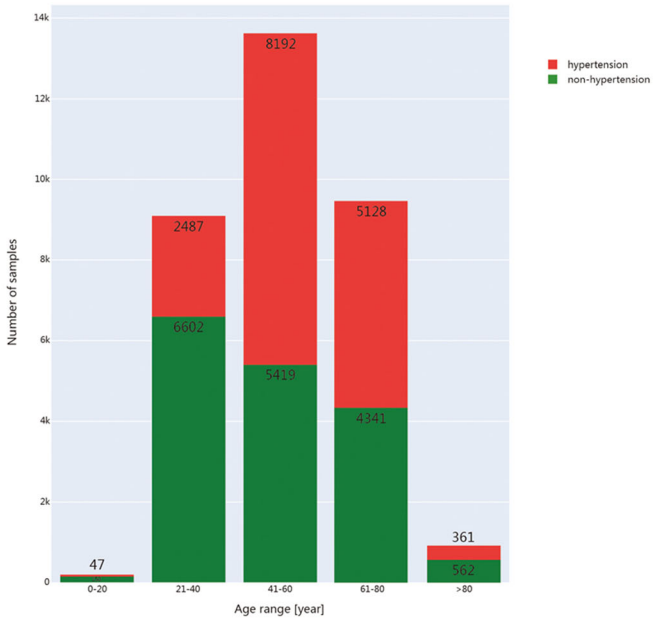
$$\text{F1} = 2\text{PR}/(P + R), \tag{5}$$

where TP = True Positive, FP = False Positive, TN = True Negative, FN = False Negative. P and R denote Precision and Recall, respectively.

Accuracy quantifies the number of correct predictions over the total number of samples. Precision refers instead to the number of true positives compared to the total



(a) The ratio of positive sample to negative sample is 48.7:51.3



(b) Number of positive and negative samples per age group

Fig. 1 Distribution of positive and negative samples

Table 2 Features containing negative values

Feature	Count	Rate (%)
Number of basophils	15	0.045
Urea nitrogen	1	0.003
Triglyceride	10	0.03

Table 3 Vanishing features (zero)

Feature	Count	Rate (%)
Age	666	0.02
Number of eosinophils	370	1.11
Number of basophils	8236	24.74
Activated partial thromboplastin time	370	0.01

number of positive samples. Recall assesses the sensitivity of the method using the ratio between the number of true positives and the sum (TP+FN). Finally, F1 score represents the harmonic mean of Precision and Recall, which provides an overall assessment of the model performance. We also employ AUC (Area Under Curve), i.e. the area under the ROC curve, as an additional figure of merit. AUC lies in the range [0,1] and the larger is the AUC, the more accurate is the classifier.

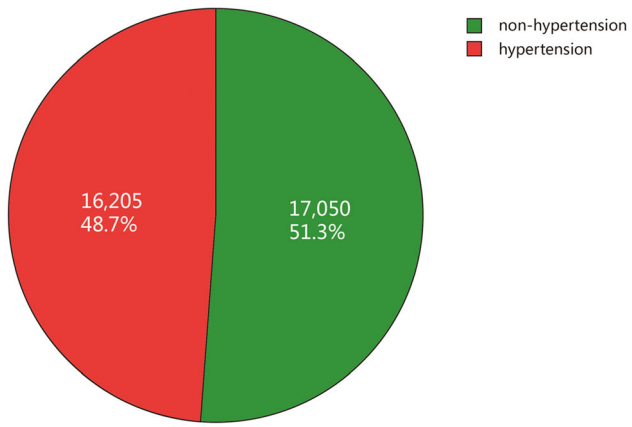


Fig. 2 The ratio of positive and negative samples after deleting outliers is 48.7:51.3

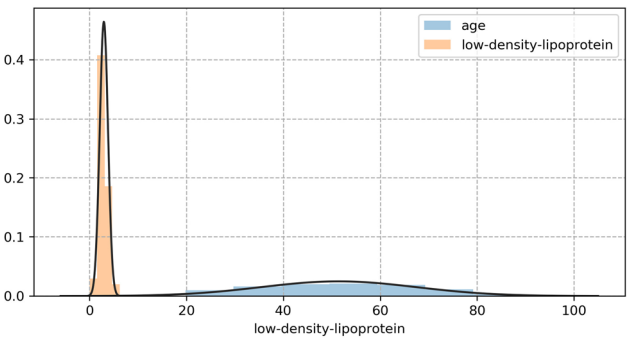


Fig. 3 The original data distribution of age and low density lipoprotein

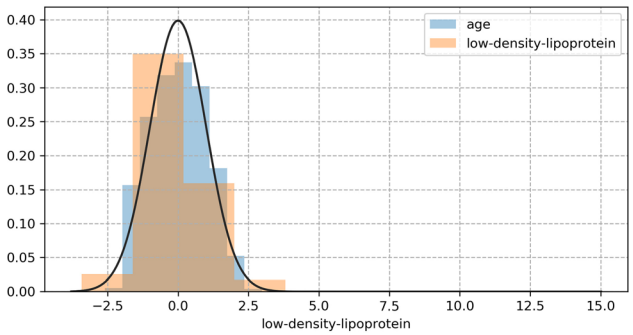


Fig. 4 Distribution of age and low-density lipoprotein after normalization

3.3.2 K-nearest neighbor (KNN) model

KNN algorithm [28] is a statistical approach proposed by Cover and Hart in 1968. There is no training stage in KNN, which is based on assigning to any unclassified sample point the classification of the nearest point in a set of previously classified points. Upon choosing a distance in data space, the k patterns closest to the pattern X are selected. Then, the most frequent class in those k patterns is take as the class of X pattern [29] (k is usually a small number).

The distances employed in the above processing are usually the Euclidean, Manhattan, or Minkwoski ones, shown in Eqs. (6), (7), and (8) respectively.

$$\mathrm{d}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{6}$$

$$\mathrm{d}(x,y) = \sum_{i=1}^{n}|x_i - y_i|, \tag{7}$$

$$\mathrm{d}(x,y) = \sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p}, \tag{8}$$

where x is the vector of features in the pattern under recognition, y is a pattern in the training database, n is the size of the feature vector size, and p is a free parameter, which is determined empirically in order to optimize performance.

We use Grid Search with Cross Validation to select the hyperparameters for KNN estimator. Cross-validation means that the incoming training data is first divided into 5 equal smaller sets (set CV = 5 to get 5 equal smaller sets), and each subset is respectively validated once, considering the remaining 4 subsets as the training subset (See Fig. 5).

In this way, every hyperparameter combination is trained on the training subset, and the trained estimators are evaluated on the validation set to calculate F1 score (in this phase, we use F1 as a figure of merit to seek for the best combination of hyperparameters). The F1 scores obtained for different verification subsets are averaged and this value is used to assess that group of hyperparameter combinations. For all candidate sets of hyperparameters, all the possible combinations are considered by a loop traversal, and by comparison we select the optimal set of hyperparameter combinations.

We search for the best combination of hyperparameters in the range shown in Table 4. KNN is then trained on the whole training set using the best hyperparameter combination and performance is evaluated on the test set. We obtain Accuracy = 0.8351, Precision = 0.7725, Recall = 0.9408 and F1 score = 0.8484 (See Fig. 6a). The AUC of this model is 0.9456 (See Fig. 6b). Results show that the achieved Recall is rather good, whereas the obtained Precision is not satisfactory enough.

3.3.3 Light gradient boosting machine (LightGBM) model

LightGBM is a Gradient Boosting framework put forward by Microsoft Research in Jan 2017. It's the new member of boosting models. LightGBM is based on GBDT. GBDT is an ensemble model of decision trees, which are trained in sequence. At each iteration, the trees learning is obtained by fitting the negative gradients. Learning the trees thus represents the main cost in GBDT. In turn, the most time-consuming step in learning a decision tree is the search of the best split points. LigthGBM is designed to solve the poor performance of GBDT when dealing with multiple features and large data size. In particular, LigthGBM exploits two techinques: gradient-based one-side Sampling (GOSS) and Exclusive Feature Bundling (EFB) [30].

1. Gradient-based One-Side Sampling (GOSS)

   Gradient-based One-Side Sampling is based on keeping large-gradient instances, and sampling randomly small-gradient instances. The reason behind this choice lies in the fact that samples with small gradient are well-trained, i.e. the training error is very small. If some data are discarded, the data distribution is changed and the accuracy of the model may be lost. GOSS thus amplifies small-gradient sampled data by a (constant) factor (1-a)/b in calculating the information gain. In this way, it reduces the amount of data and the computational burden, overall achieving faster speed.

   The calculation steps of GOSS are the following:

   (a) Sort data according to the absolute value of their gradients;
   (b) Select the top a×100% instances with the largest gradients to build the subset A;
   (c) From the complementary set A^C, randomly sample b×100% instances to form the subset B;

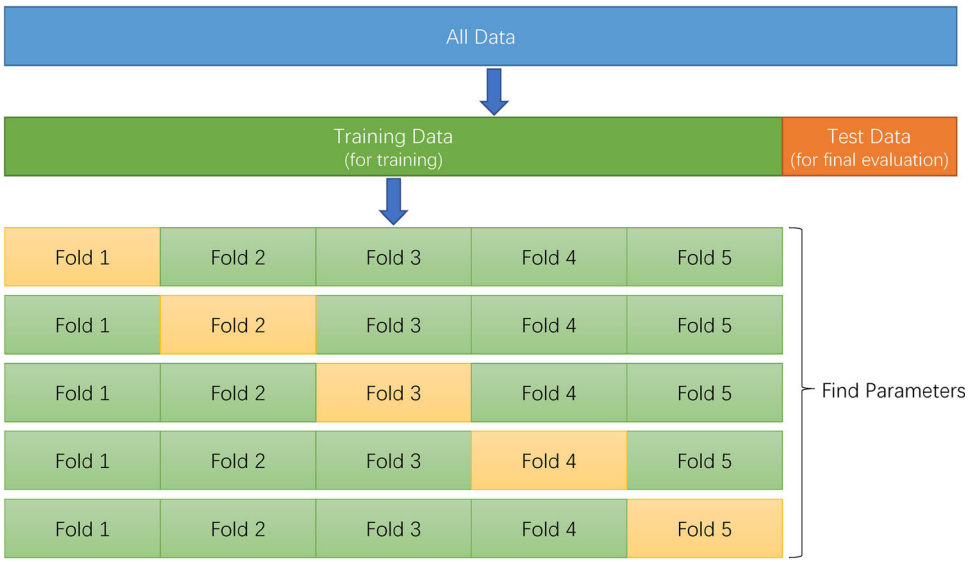


Fig. 5 Searching for the best combination of hyperparameters by cross validation

Table 4 The hyperparameters of KNN

Hyperparameter	Options/Range	Selected value
Weights	Uniform, distance	Distance
*n*_neighbors	1–20	18
P(available only when weights="distance")	1–20	1

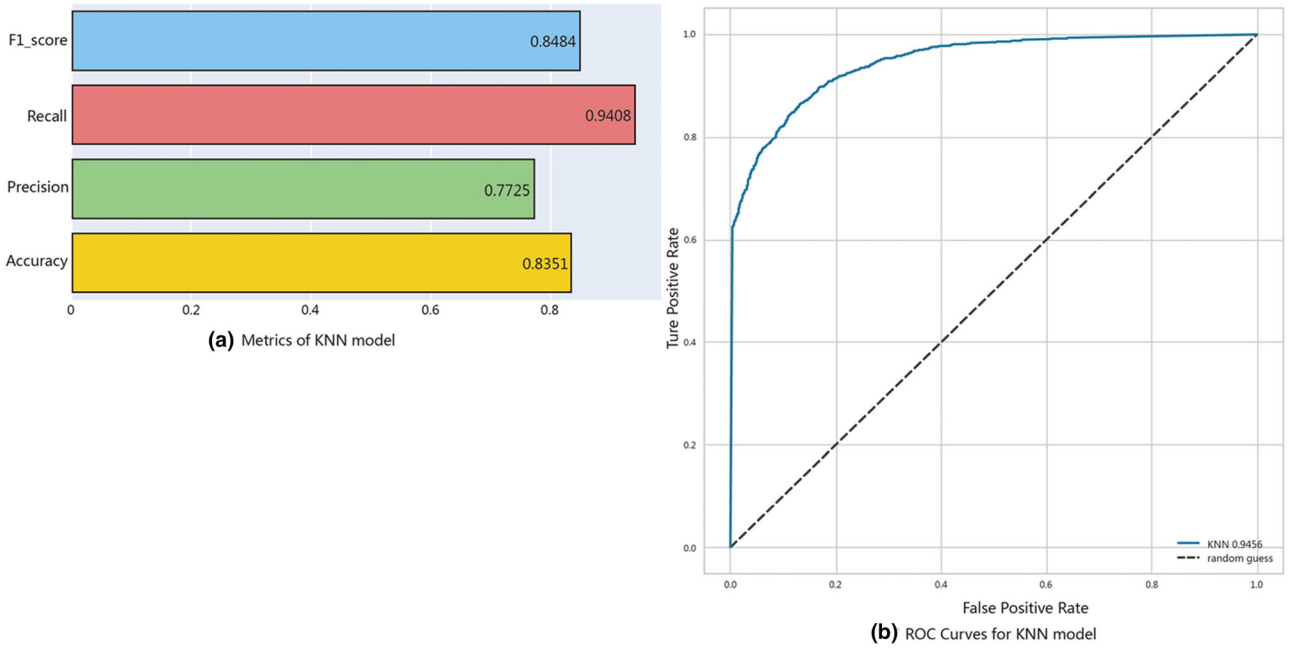


Fig. 6 Performance of KNN model

(d) Split data according to the estimated variance gain $\tilde{V}_j(d)$ over the subset $A \cup B$ i.e.

$$\tilde{V}_j(d) = \frac{1}{n}\left(\frac{\left(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i\right)^2}{n_l^j(d)} + \frac{\left(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i\right)^2}{n_r^j(d)}\right), \tag{9}$$

where the different sets in the above formula are defined as follows: $A_l = \{x_i \in A, x_{ij} \leq d\}$, $A_r = \{x_i \in A, x_{ij} > d\}$, $B_l = \{x_i \in B, x_{ij} \leq d\}$, $B_r = \{x_i \in B, x_{ij} > d\}$. The factor $\frac{1-a}{b}$ is used to renormalise the gradients in B to the size of A^C.

2. EFB (Exclusive feature bundling)

Exclusive Feature Bundling (independent feature combination) reduces the number of features and improves computing efficiency. If the features that are bundled are mutually exclusive, there is no loss of information. On the other hand, if two or more features are not fully mutually exclusive (they may be all nonzero), we should use a conflict rate to measure the degree of non-mutual exclusion of those features, and assess whether to bundle them together or not in order to get a suitable balance between the conflicting needs of accuracy and efficiency.

EFB uses the Greedy Bundling algorithm to determine which features may be bundled together. Then, EFB rebuilds newBin and binRanges by using the Merge Exclusive Features algorithm.

LightGBM model is characterized by many hyperparameters, so we use Random Grid Search with Cross Validation to look for the optimal hyperparameters in a given range. Random Grid Search means sampling randomly in the hyperparameter space instead of traversing all possible hyperparameter combinations. Rather obviously, when the number of hyperparameters is large, Random Grid Search is much faster than Grid Search.

We search for the best combination of hyperparameters in the ranges reported in Table 5, by using Random Grid Search with Cross Validation. We then train the LightGBM estimator with using the best combination of hyperparameters on the entire training set, and finally evaluate it on the test set. Overall, we obtain Accuracy = 0.8654, Precision = 0.8373, Recall = 0.9007 and F1 score = 0.8678 (See Fig. 7a). The AUC of the model is 0.9284 (See Fig. 7b). Accuracy, Precision and F1 score are better than those of KNN model, whereas Recall and AUC are lower.

3.3.4 Model integration

Upon comparing the performance of KNN model to that of LightGBM model, we notice that Accuracy, Precision and F1 score of LightGBM are larger than KNN model. In particular, Precision of LightGBM is 6.48% higher than that of KNN model. However, Recall of KNN model is quite satisfactory, reaching 94.08%, while Recall of LightGBM model is only 90.07%.

The confusion matrices of the two models on the test set are shown in Fig. 8. We can see that TP and FN of KNN model are better than LightGBM model, whereas the situation is reversed for TN and FP.

Since our model is being developed to predict the risk of hypertension, Recall is a relevant figure of merit. We indeed aim to avoid missing a sample that may develop into hypertension within 5 years. On the other hand, Precision is also important since we should avoid false alarms, which would cause unnecessary anxiety due to wrong classification.

In order to satisfy those conflicting requirements, we propose a hybrid prediction model based on KNN and LightGBM (See Fig. 9). Our KNN-LightGBM hybrid model may be summarized in the following steps:

(a) Split the dataset into a training and a test sets, using the ratio 8:2;
(b) Train the best KNN model on training dataset using grid search and cross validation;
(c) Train the best LightGBM model on training dataset using random grid search and cross validation;
(d) Exploit KNN and LightGBM models obtained from steps (b) and (c) to predict the test data, and gets the probabilities of the 2 categories;
(e) Extract the final classification results by a weighted average of the classification probabilities of KNN and LightGBM models (according to the weight of 1:0.8).

We evaluate the KNN-LightGBM model on the test set. Fig. 10 shows the performance of the hybrid model. We obtain Accuracy = 0.8606, Precision = 0.8168, Recall = 0.9227 and F1 score = 0.8666. The red line is the ROC curve of the KNN-LightGBM hybrid model, which is significantly better than the ROC curve of the KNN (blue line) and LightGBM (green line) models. The AUC on the hybrid model is 0.9505 (See Fig. 10b).

4 Results

In addition to KNN, LightGBM and KNN-LightGBM hybrid model, we process data also using SVM, Random Forest and shallow neural network models. Table 6 shows a comparison

Table 5 The hyperparameters of LightGBM

Hyperparameter	Options/Range	Selected value
learning_rate	[0.01, 0.02, 0.03, 0.04, 0.05, 0.08, 0.1, 0.2, 0.3, 0.4]	0.1
n_estimators	Range (100, 2001, 100)	1600
max_depth	[7, 9, 11, 13, 15, 17, 19, 21]	15
num_leaves	sp_randint (6, 50)	48
min_child_samples	sp_randint (100, 500)	299
min_child_weight	[1.e-5, 1.e-3, 1.e-2, 1.e-1, 1., 1.e1, 1.e2, 1.e3, 1.e4]	10
subsample	sp_uniform(loc=0.2, scale=0.8)	0.5981423828285206
colsample_bytree	sp_uniform(loc=0.4, scale=0.6)	0.6208614411889205
reg_alpha	[0, 1e-1, 1, 2, 5, 7, 10, 50, 100]	2
reg_lambda	[0, 1e-1, 1, 5, 10, 20, 50, 100]	1

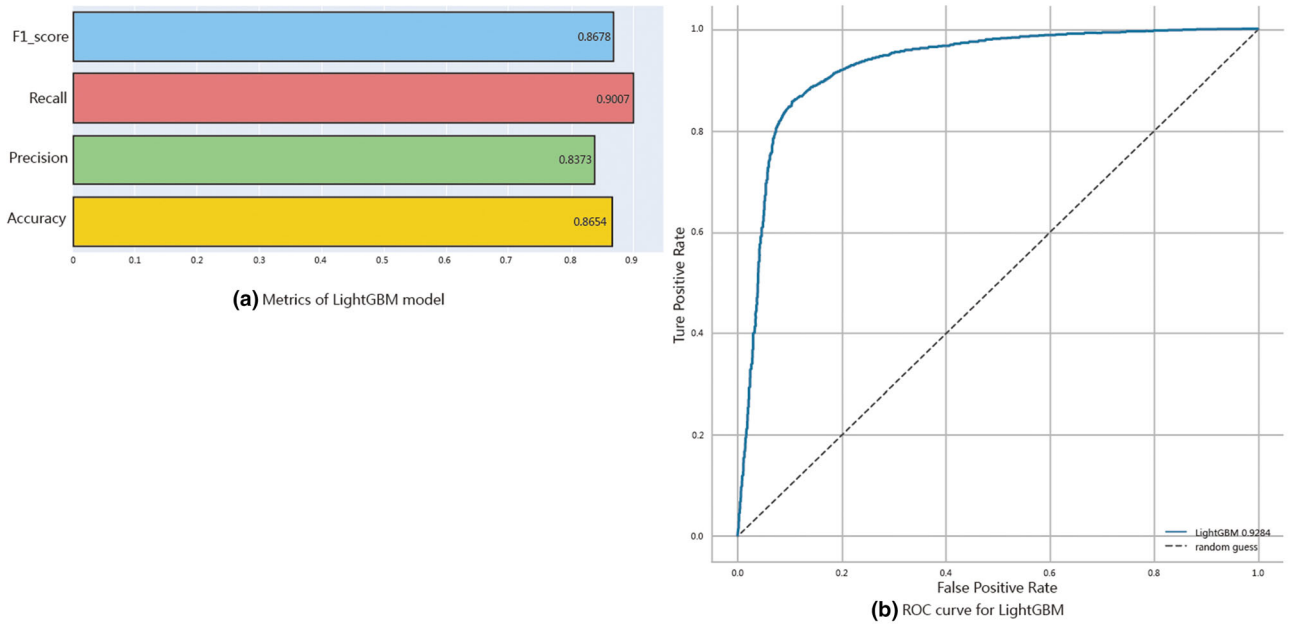


Fig. 7 Performance of LightGBM model

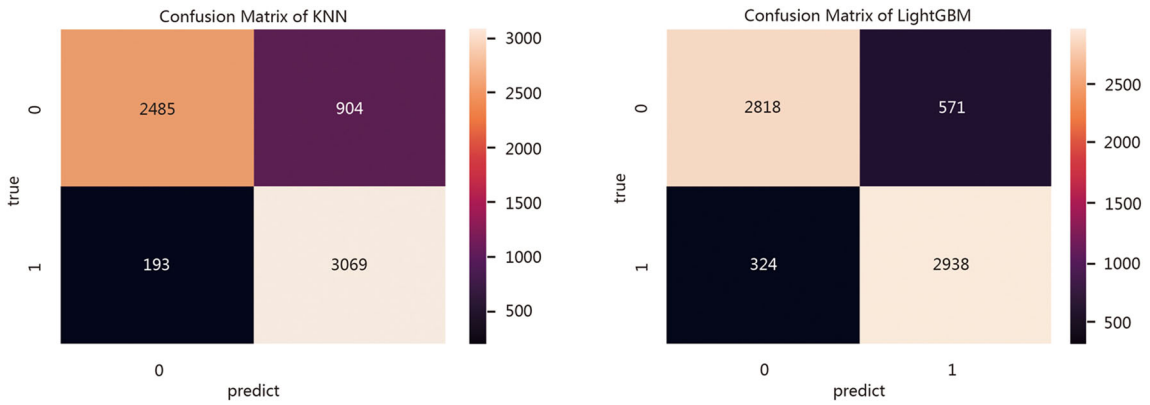


Fig. 8 Confusion matrices of KNN model and LightGBM models

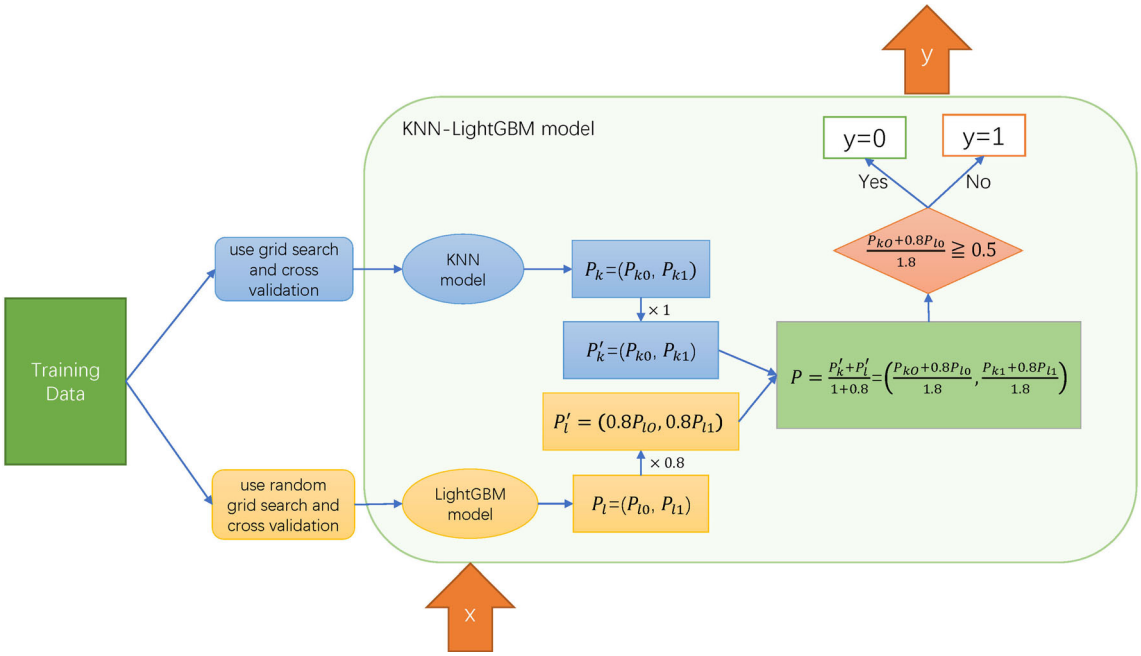


Fig. 9 Build a KNN-LightGBM hybrid model

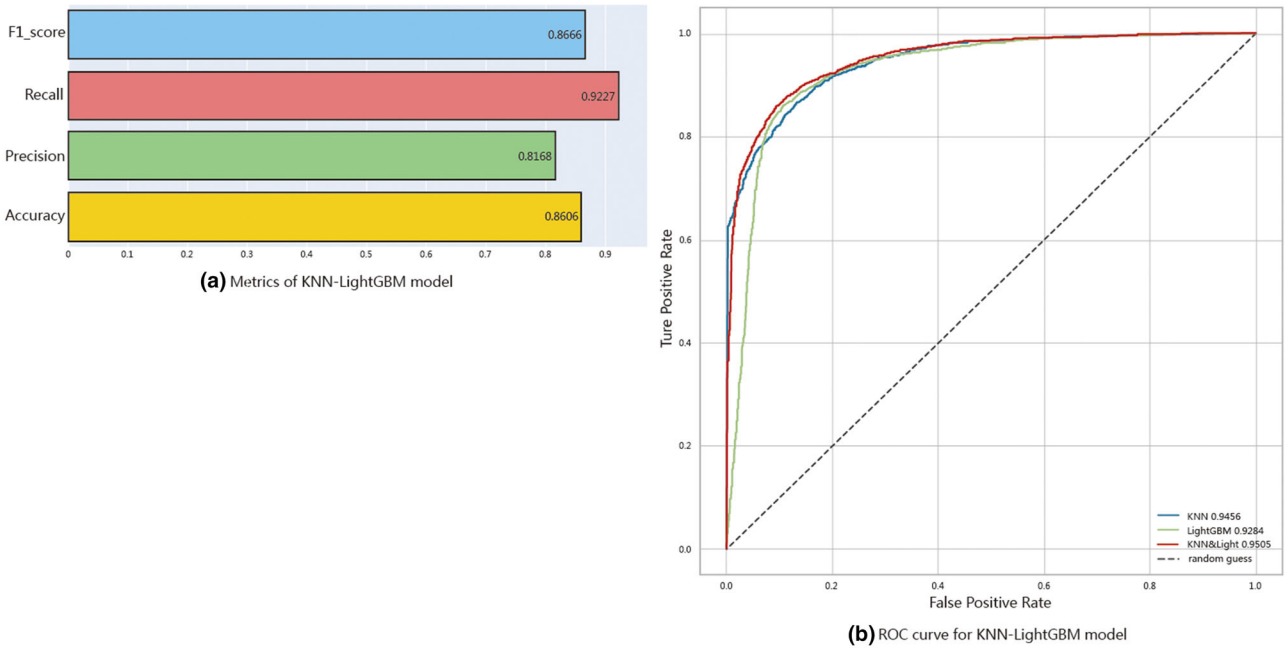


Fig. 10 Performance of KNN-LightGBM hybrid model

among the performance of the six models on this dataset. It can be seen that KNN-LightGBM hybrid model performs significantly better than SVM, Random Forest and 6-layer neural network model in Accuracy, Precision and F1 score. Also, KNN-LightGBM hybrid model is more balanced than KNN and LightGBM. Finally, it is superior to the other models in terms of the AUC value.

Table 6 Performance of six different models

Model	Accuracy	Precision	Recall	F1 score	AUC
KNN	0.835	0.773	0.941	0.848	0.946
LightGBM	0.865	0.837	0.901	0.868	0.928
KNN-LightGBM	0.861	0.817	0.923	0.867	0.951
SVM	0.823	0.78	0.889	0.831	0.889
Random Forest	0.845	0.814	0.887	0.849	0.913
Neural Network (with 6 hidden layers)	0.840	0.798	0.902	0.847	0.895

5 Conclusion

In this paper, a risk prediction model for hypertension within 5 years has been established. KNN and LightGBM have been combined to exploit the best features of both, and then soft voting is adopted to obtain the final classification results. The performance of the hybrid model have been compared with those of SVM, Random Forest and 6-layer neural network models. Experimental results show that KNN-LightGBM hybrid model has a better classification performance in this dataset. We expect out hybrid modelto better assist doctors in the prediction and diagnosis of hypertension, improving the accuracy and reducing the misdiagnosis rate.

We also plan to study more factors influencing hypertension. In fact, according to literature, a family history and risk factors such as smoking and alcohol abuse are directly associated with hypertension. We plan to use resident health data to mine this information and incorporate it into our prediction model.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Mills KT, Bundy JD, Kelly TN, Reed JE, Kearney PM (2016) Global disparities of hypertension prevalence and control: a systematic analysis of population-based studies from 90 countries. Circulation 134(6):441–450
2. Liu L (2019) 2018 Chinese guidelines for the management of hypertension. Chin J Cardiovasc Med 24:24–56
3. Liu L (2011) Writing group of Chinese guidelines for the management of H: 2010 Chinese guidelines for the management of hypertension [Article in Chinese]. Zhonghua xin xue guan bing za zhi 39:579–615. https://doi.org/10.1038/cdd.2010.68
4. Chen ShuTang (1999) Prevention of hypertension [Article in Chinese]. China Healthcare & Nutrition 10:26–27. https://doi.org/CNKI:SUN:ZHBJ.0.1999-10-018
5. Boudabsa L, Filipovic D (2020) Machine learning with kernels for portfolio valuation and risk management, Papers, arXiv.org. https://EconPapers.repec.org/RePEc:arx:papers:1906.03726
6. Pankajakshan P, Sanyal S, de Noord OE, Bhattacharya I, Bhattacharyya A, Waghmare U (2017) Machine learning and statistical analysis for materials science: stability and transferability of fingerprint descriptors and chemical insights. Chem Mater 29(10):4190–4201
7. Raza A, Bardhan S, Xu L, Yamijala SS, Lian C, Kwon H, Wong BM (2019) A machine learning approach for predicting defluorination of per-and Polyfluoroalkyl Substances (PFAS) for their efficient treatment and removal. Environ Sci Technol Lett 6(10):624–629
8. Lotfi B, Damir F (2019) Machine learning with kernels for portfolio valuation and risk management. SSRN Electron J 01:2019
9. Song H, Srinivasan R, Sookoor T, Jeschke S (2017) Smart cities: foundations, principles and applications. Wiley, Hoboken, NJ, pp 1–906. ISBN: 978-1-119-22639-0
10. Sabina J, Christian B, Houbing S, Rawat DB (2017) Industrial internet of things. Springer International Publishing, New York
11. Yunchuan S, Houbing S, Jara AJ, Rongfang B (2017) Internet of things and big data analytics for smart and connected communities. IEEE Access 4:766–773
12. Yuan Z, Limin S, Houbing S, Xiaojun C (2014) Ubiquitous WSN for healthcare: recent advances and future prospects. IEEE Internet Things J 1(4):311–318
13. Lo'Ai AT, Mehmood R, Benkhelifa E, Song H (2017) Mobile cloud computing model and big data analysis for healthcare applications. IEEE Access 4(99):6171–6180. https://doi.org/10.1109/ACCESS.2016.2613278
14. Tawalbeh LA, Bakhader W, Mehmood R, Song H (2016) Cloudlet-based mobile cloud computing for healthcare applications. 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, pp 1–6. https://doi.org/10.1109/GLOCOM.2016.7841665
15. Jiang B, Yang J, Lv Z, Song H (2018) Wearable vision assistance system based on binocular sensors for visually impaired users. IEEE Internet Things J 6(2):1375–1383
16. Jiang Y, Song H, Wang R, Gu M, Sun J, Sha L (2016) Data-centered runtime verification of wireless medical cyber-physical system. IEEE Trans Ind Inform 13(4):1900–1909
17. Abraham A, Pedregosa F, Eickenberg M, Gervais P, Mueller A, Kossaifi J, Varoquaux G (2014) Machine learning for neuroimaging with scikit-learn. Front Neuroinform 8:14
18. Yang Y, Niehaus KE, Walker TM, Iqbal Z, Walker AS, Wilson DJ, Peto TE, Crook DW, Smith EG, Zhu T (2018) Machine learning for classifying tuberculosis drug-resistance from DNA sequencing data. Bioinformatics 34(10):1666–1671
19. Hazan H, Hilu D, Manevitz L, Ramig LO, Sapir S (2012) Early diagnosis of Parkinson's disease via machine learning on speech data. 2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel, pp 1–4. https://doi.org/10.1109/EEEI.2012.6377065
20. Park SH, Kim SG (2018) Comparison of hypertension prediction analysis using waist measurement and body mass index by age group. Osong Public Health Res Perspect 9(2):45–49

21. Muhammad I, Ganjar A, Muhammad S, Jongtae R (2018) Hybrid prediction model for type 2 diabetes and hypertension using DBSCAN-based outlier detection, synthetic minority over sampling technique (smote), and random forest. Appl Sci 8(8):1325
22. Wang A, An N, Xia Y, Li L, Chen G (2014) A logistic regression and artificial neural network-based approach for chronic disease prediction: a case study of hypertension. 2014 IEEE International Conference on Internet of Things (iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom), pp 45–52. https://doi.org/10.1109/iThings.2014.16
23. Hiroshi K, Kenji S, Kyohei F, Tetsuya I, Kazuomi K (2019) Highly precise risk prediction model for new-onset hypertension using artificial intelligence techniques. J Clin Hypertens 22:445–450
24. Zhao H, Ma Z, Sun Y (2019) A hypertension risk prediction model based on BP neural network. 2019 International Conference on Networking and Network Applications (NaNA), pp 464–469. https://doi.org/10.1109/NaNA.2019.00085
25. He J, Whelton PK, Appel LJ, Charleston J, Klag MJ (2000) Long-term effects of weight loss and dietary sodium reduction on incidence of hypertension. Hypertension 35(2):544–549
26. Emilia H, Pernilla D, Amira E, Claude M (2019) The effect of weight loss and weight gain on blood pressure in children and adolescents with obesity. Int J Obes 43:1988–1994
27. Trials of Hypertension Prevention Collaborative Research Group (1997) Effects of weight loss and sodium reduction intervention on blood pressure and hypertension incidence in over-weight people with high normal blood pressure: the trials of hypertension prevention. Arch Int Med 157:657–667
28. Cover T, Hart P (2003) Nearest neighbor pattern classification. IEEE Trans Inf Theory 13(1):21–27
29. Zanchettin C, Bezerra BLD, Azevedo WW (2012) A KNN-SVM hybrid model for cursive handwriting recognition. The 2012 International Joint Conference on Neural Networks (IJCNN), pp 1–8. https://doi.org/10.1109/IJCNN.2012.6252719
30. Ke G, Meng Q, Finely T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y (2017) Lightgbm: a highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems 30 (NIP 2017). https://papers.nips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf