Heart Disease Recognition Through Machine Learning and Deep Learning

Juntao Lin
Department of Electrical and Computer Engineering
Queen's University, Kingston, ON, Canada
21jl80@queensu.ca

***Abstract*—Obesity increases the risk of having heart disease, which is the one of the most severe causes of mortality in the world. Heart disease can be hard to detect in the preliminary stages since the multiple health indicators must be considered. Machine learning has been proved to be effective in assisting decision making as well as classification. This paper proposes to use different machine learning and deep learning techniques to identify heart disease. Th e proposed models are evaluated using the combined UCI heart disease dataset. The RF and SVM models have the best performance that results in a 100% validation accuracy.**

I. INTRODUCTION

Obesity increases the risk of diagnosing cardiovascular disease (CVD) that mostly related to heart failure and coronary heart disease. CVD could potentially alter the hemodynamics and heart structures [1]. The US obesity prevalence reached 42.4% in 2017-2018 [2]. Heart disease became one of the most significant causes of mortality in the world today. About 659,000 people in the United States die from heart disease each year, heart disease costs the United States about $363 billion each year from 2016 to 2017 [3].

In clinical data analysis, predicting cardiovascular disease is a critical challenge. Heart disease composed of different symptoms, such as diabetes, high blood pressure, high cholesterol, and abnormal pulse rate. Each symptom individual does not imply whether the patient has heart disease, whereas a patent showing multiple symptoms would be likely to diagnosed with cardiovascular disease.

Machine learning has been shown to be effective in assisting decisions making and predictions. Large quantity of data produced by the healthcare industry every day could benefit the development of different machine learning algorithms. Various techniques in machine learning and deep learning have been employed to analyze medical data. In regions that do not have enough cardiovascular experts, machine learning can fill in the gap to perform the preliminary analysis of the patient's records.

CVD data often come in the form of a combination of numerical and categorical data. The categorical data may include extra information about patient's medical history. The numerical data provide readings of patient's conditions. This paper proposes to use support vector machine (SVM), multilayer perceptron (MLP), and ensemble learning methods to perform heart disease classification.

II. RELATED WORK

Through machine learning approaches, CVD can be predicted though dataset contains categorical and numerical records, as well as electrocardiogram (ECG) signal. This section will provide research done previously through these approaches.

A. Categorical and Numerical Data

Dehkordi et al. [4] proposed a stacking model using ensemble learning techniques to determine patient's disease and the type of physician based on the drug record. A custom dataset had been collected for this task. Each sample in the dataset includes sex, age, and the name of the prescription drug. The labeling process of the custom dataset are completed by pharmacy students and professors. They compared the performance of the prosed stacking model with three classification algorithms: Decision Tree (DT), Naïve Bayes (NB), K-Nearest Neighbors (KNN). The prosed model had the best performance with an accuracy of 73.17%.

Malav et al. [5] proposed a hybrid model of KNN and neural network. After preprocessing, the model uses KNN to cluster the input samples, followed by a neural network classifier with 1 hidden layer. The UCI Heart Disease dataset [5] was used to validate the hybrid model, resulted in a 97% of validation accuracy.

B. CVD Classification through ECG Signal

Gao et al. [7] proposed a LSTM model to detect 8 different arrhythmias on imbalanced ECG dataset. A focal loss function was added to the LSTM network to counter the imbalance ECG category by lowering the weights for normal ECG samples. The model achieved more than 99% accuracy, recall, and precision in the MIT-BIH arrhythmia database [6].

Kachuee et al. [9] proposed a deep Convolutional Neural Network (CNN) with residual connections to classify heart beats into 5 arrhythmias under the AAMI EC57 standard. The proposed model uses 3 1-D convolution layers, 1 max pooling layer, and 2 Fully Connected (FC) layers. The model was validated through the PTB Diagnostics datasets [8]. The validation accuracy on the 2-lead ECG samples is 95.9%.

Zhang et al. [9] proposed a CNN network to classify heart disease from 1-lead ECG signals. The model has 4 1-D convolution layers, 1 max pooling layer, 1 global average

pooling layer, and 1 FC layer. To speed up the training, the input data are standardized with Z-score. The proposed model was tested with the MIT-BIH arrhythmia database. The precision, sensitivity, and F1 score achieved by the model are 97.7%, 97.6%, and 97.6% respectively.

III. METHODS

This section introduces the proposed machine learning and deep learning models for predicting heart disease.

A. SVM

The proposed SVM models is used for binary classification. The classic SVM model is designed for solving dataset with high dimensionality while the size of the training data is relatively small [12]. The SVM classifier that aims to create a decision boundary between the two classes. This decision boundary is also known as hyperplane. The optimization process maximizes the margin between the hyperplane and the closest data from each the classes. Theses closest data points are support vectors that helps the SVM model to adjust the decision boundary. Through binary classification, the hyperplane is defined as:

$$wx_i^T + b = 0 \quad (1)$$

To successfully identify the training samples, all the samples must satisfy the following constrains:

$$\mathrm{wx}_i^T + \mathrm{b} \geq +1 \quad \text{for} \quad y_i = 1 \quad (2)$$
$$\mathrm{wx}_i^T + \mathrm{b} \leq -1 \quad \text{for} \quad y_i = -1 \quad (3)$$

The kernel function combines the original features of the training data to create more features. Therefore, through kernel functions, SVM maps training samples to a feature space with higher dimension. In section IV, simulations will be performed using different kernel functions. Linear kernel, polynomial kernel, radio basis function (RBF) kernel, and sigmoid kernel are described in equations 4-7 respectively.

$$K(x_i, x_j) = x_i^T x_j \quad (4)$$

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \quad d > 1 \quad (5)$$

$$K(xi, xj) = \exp(-\gamma || x_i - x_j ||^2), \gamma > 0 \quad (6)$$

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r), \gamma > 0, r < 0 \quad (7)$$

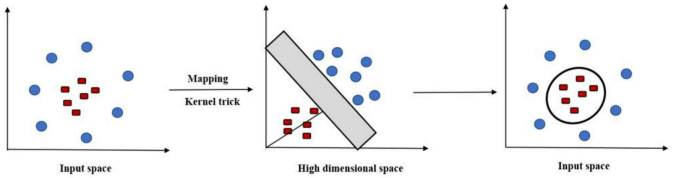

Fig. 1. SVM with kernel function [13]

B. MLP

The MLP model contains 1 input layer, 2 hidden layers, and 1 output layer. Both the input and the hidden layers are dense layers that use Rectified Linear Units (ReLU) as the activation functions. Each dense layer has 256 neurons. The input data received by this model are samples with 14 features that related to the presence of heart disease in the patient. The first hidden layer has 64 neurons that use sigmoid activation function. The outputs of the second hidden layer are flattened before feeding to the output layer. The output layer uses the Softmax function to generate a normalized probability distribution. The class with the highest probability will be selected as the predicted outcome.

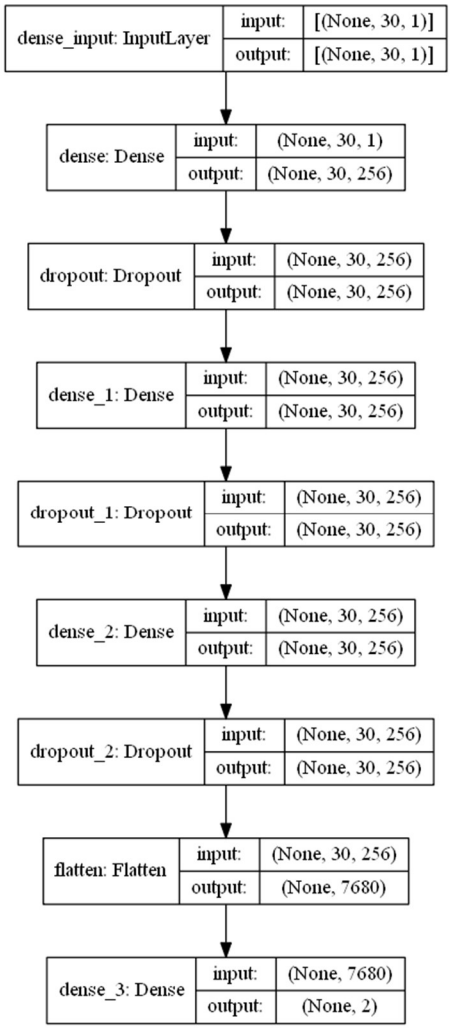

Fig. 2. Proposed MLP Structure

C. Ensemble Learning

In this section boosting and bagging methods will be introduced, Adaptive Boosting (AdaBoost) and Random Forest (RF) are proposed to solve the UCI heart disease dataset. AdaBoost [14] creates multiple weak learners to evaluate the samples in the dataset. After each of the weak learner finish the evaluation, the outputs will be combined to form a strong

learner that produces a more robust prediction. Initially, each sample in the dataset will be assigned a sample weight 1/N where N is the number of samples in the training data. During each iteration in the training process, all the sample weights will be modified individually. Misclassifying a sample increases its sample weight; correctly classify a sample lowers its sample weight. Therefore, in the next step, the subsequent weak learner must pay more attention to these misclassified samples. Once the weak learners finish the classification, classifier weights that are proportional to the accuracy will be assigned to each of the weak learners. The predictions from all the weak learners are combined through a weighted majority voting to make the final prediction. The classifier weight deicides the influence of each weak learner during the voting process. The proposed AdaBoost model uses 500 weak learners. Each weak learner is a stump, which is a decision tree that only has 1 node and 2 leaves.

RF [15] build multiple independent decision trees. The final prediction is made by combining the outputs of each tree through majority voting. Unlike AdaBoost, every decision tree has an equal weight during the voting process. Decision trees are built from a bootstrap sample from the training data. To increase computation efficiency, the splitting rule (ie. choosing the next nodes) can be randomized among a subset of the features included in training data [16]. When computation power is not a constrain, the splitting rule is evaluated quantitatively using the Gini or Entropy criterion. The Gini impurity and information entropy are described in equations 8 [17] and 9 [18]. p(i) is the probability of randomly picking a sample from class i. While evaluating the quality of a split, the optimal choice would have either a low Gini impurity or low entropy.

$$G = \sum_{i=1}^{C} p(i) \times (1 - p(i)) \qquad (8)$$

$$E = -\sum_{i=1}^{C} p(i) \log_2 p(i) \qquad (9)$$

RF creates distinct decision trees structures to encourage different instances of overfitting and outliers [19]. In the ensemble model, the predictions of the trees are combined through majority voting. RF generates decision trees that are high in variance and low in bias [20]. Although the individual tree has the overfitting problem, combing the outputs of these trees would mitigate the overfitting. Each tree votes for a specific class, the class with the highest vote will be the final prediction. The proposed RF model constructs 10 unique decision trees at a maximum depth of 15.

IV. Experiment Result

In this section, the experiment results are presented for each of the proposed models. The dataset [21] used in this paper combines the four UCI Heart Disease Data (Cleveland, Hungary, Switzerland, and the VA Long Beach). The combined dataset has 1025 sample, each sample have 13 features. The training set includes 80% of the dataset, the remaining 20% samples is validation set.

Features	Description
age	Age of the patient
sex	Male/Female
cp	Chest pain type
trestbps	Resting blood pressure
chol	Serum cholesterol in mg/dl
fbs	Fasting blood sugar
restecg	Resting ECG results
thalach	Maximum heart rate
exang	Exercise-induced angina
oldpeak	ST depression induced by exercise
slope	The slope of the peak exercise ST segment
ca	Number of major vessels colored by fluoroscopy
thalassemia	An inherited blood disorder
target	Whether the patient has heart disease

Table 1: Features included for each sample in the dataset

A. Preprocessing

The samples in the dataset are saved in a single CSV file that contains both categorical data and numerical data. The first step is to check for samples that have a missing feature (ie. all the sample should have 14 features). Followed by converting categorical feature into numerical features. Both One-Hot encoding and Label encoding are tested. Table 2 shows that models using One-Hot encoding have a better performance.

Method	Encoding	Accuracy
SVM	Label Encoding	0.834
MLP	Label Encoding	~0.85
SVM	One-Hot Encoding	0.863
MLP	One-Hot Encoding	~0.87

Table 2: Comparison of different encoding methods

B. SVM

Different kernel functions have been tested for the SVM model. The performances of different SVM models are summarized in Table 3.

Method	Kernel	Accuracy	Precision	Recall
SVM	Linear	0.863	0.853	0.886
SVM	Polynomial	0.839	0.84	0.848
SVM	Sigmoid	0.512	0.512	1
SVM	RBF	1	1	1

Table 3: SVM performance with different kernel functions

To improve the accuracy of SVM models, we decide to add a preprocessing step that is using K-means to cluster the raw data. The input of the K-means function is a 1025x13 matrix (1025 samples, 13 attributes each). The output of the K-means is a 1025xM matrix, where M is the number of clusters. After preprocessing, each sample has M features that represents the distances between the sample and the centers of different clusters. To determine the impact of this extra preprocessing step, we adjust the number of clusters used in the K-means

algorithm. Table 4 shows the impact of tuning the number of clusters. The performance of SVM increases as the number of clusters increase. Table 5 shows the performance of the SVM models improved after integrating K-means to preprocessing. The linear kernel SVM has the highest improvement from 86.3% to 100% validation accuracy. The K-means only applies to SVM models, other proposed models does not include K-means in the preprocessing step.

Method	Clusters	Accuracy	Precision	Recall
SVM	14	0.678	0.705	0.638
SVM	100	0.863	0.867	0.867
SVM	200	0.961	0.953	0.971
SVM	300	1	1	1

Table 4: Performance of SVM with different number of clusters, all the simulations use the same SVM model with linear kernel

Method	Kernel	Accuracy	Precision	Recall
SVM	Linear	1	1	1
SVM	Polynomial	0.893	0.888	0.905
SVM	Sigmoid	0.512	0.512	1
SVM	RBF	1	1	1

Table 5: Performance of SVM with K-means, 300 clusters

C. MLP

The MLP model reached the 80% validation accuracy within 5 epochs. To slow down the learning process, a 30% drop out is added after each of the dense layers. Different optimizers have been tested, as well as different loss functions. The MLP model has the best performance with the Adam optimizer is used with the Binary Cross Entropy loss function. The proposed model has 3 dense layers (input layer and two hidden layers), with 256 neurons on each layer. Increasing the number of neurons does not increase the validation accuracy, neither as increase the number of dense layers. Reducing the number of dense layers from 3 to 2 lowers the validation accuracy by 3~4%. Using the SGD optimizer, the model did not converge to an optimal solution, whereas the Adam optimizer could handle the gradients propagated in the MLP model. The final validation accuracy achieved by the MLP model is between 86~88%.

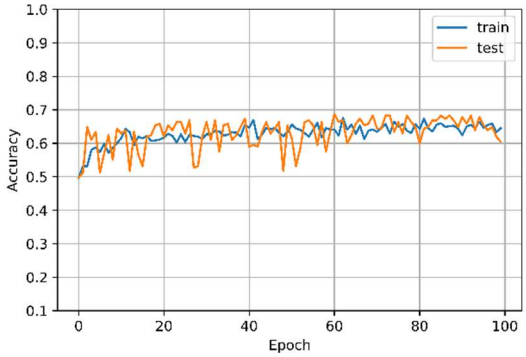

Fig. 3. MLP training curve, SGD optimizer

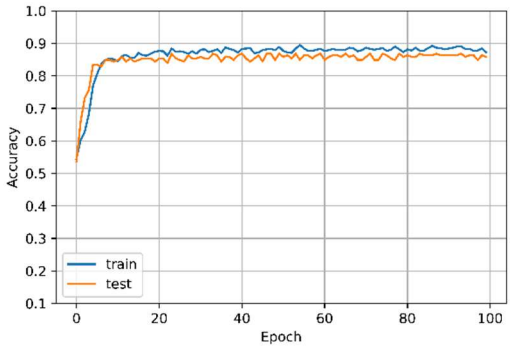

Fig. 4. MLP training curve, Adam optimizer

D. Ensemble Methods

Both the RF and AdaBoost reaches 100% accuracy on the validation set. In RF models, the maximum depth of each tree is set to 13 since the samples in the dataset contains 13 features. Either Gini or Entropy criterion can be selected for evaluating the quality of a split. Since the RF does not always generate the same result, multiple simulations are done to validate the performance of Gini and Entropy criterion. Table 6 shows these two criterions have similar performance. In this particular simulation, Gini has a better performance than Entropy.

Method	n_trees	Accuracy	Precision	Recall
RF, Gini	1	0.942	0.943	0.943
RF, Gini	5	1	1	1
RF, Gini	10	1	1	1
RF, Entropy	1	0.945	0.943	0.949
RF, Entropy	5	0.977	0.957	1
RF, Entropy	10	1	1	1

Table 6: RF with Gini and Entropy criterion

AdaBoost uses stump as the base estimator. The performance of AdaBoost is proportional to the amount of base estimator incorporated in the model. Compared with RF, AdaBoost has a more consistent performance. AdaBoost reaches 90% validation accuracy through 100 base estimators. With 2000 estimators, the validation accuracy reaches 100%, which also implies the model is severely overfitting.

Method	n_estimator	Accuracy	Precision	Recall
AdaBoost	100	0.899	0.894	0.911
AdaBoost	500	0.974	0.987	0.962
AdaBoost	1000	0.981	1	0.962
AdaBoost	2000	1	1	1

Table 7: AdaBoost with different amount of base estimators

V. CONCLUSION

Different machine learning and deep learning algorithms are presented to solve the UCI heart disease dataset. The RF, linear kernel SVM, and RBF kernel SVM have the best performance among the proposed models. These methods reach 100% validation accuracy while having a relatively low computation complexity. The proposed RF model uses 10 decision trees with the maximum depth of 13. The SVM models include K-means in the preprocessing step to improve the accuracy and robustness.

VI. REFERENCES

[1] S. Carbone, J. M. Canada, H. E. Billingsley, M. S. Siddiqui, A. Elagizi and C. J. Lavie, "Obesity paradox in cardiovascular disease: where do we stand?," *Vasc Health Risk Manag,* vol. 15, pp. 89-100, 2019.

[2] C. Hales, M. Carroll, C. Fryar and C. Ogden, "Obesity is a common, serious, and costly disease," *Centers for Disease Control and Prevention,* 2021, [online] Available: https://www.cdc.gov/nchs/products/databriefs/db360.htm.

[3] "Heart Disease Facts," *Centers for Disease Control and Prevention,* 2020, Accessed April 20, 2022, Available at: https://www.cdc.gov/heartdisease/facts.htm.

[4] S. K. Dehkordi and H. Sajedi, "Prediction of disease based on prescription using data mining methods," *Health and Technology,* vol. 9, pp. 37-44, 2019.

[5] A. Malav, K. Kadam and P. Kamat, "Prediction of Heart Disease Using K-Means and Artificial Neural Network as Hybrid Approach to Improve Accuracy," *International Journal of Engineering and Technology,* vol. 9, no. 4, pp. 3081-3082, 2017.

[6] A. Janosi, W. Steinbrunn, M. Pfisterer and R. Detrano, "Heart Disease Data Set," *[online], Available: https://archive.ics.uci.edu/ml/datasets/heart+disease, Accessed: April 23rd 2022.*

[7] J. Gao, H. Zhang, P. Lu and Z. Wang, "An Effective LSTM Recurrent Network to Detect Arrhythmia on Imbalanced ECG Dataset," *Journal of Healthcare Engineering,* 2019, Article ID 6320651, https://doi.org/10.1155/2019/6320651.

[8] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," *IEEE Eng in Med and Biol,* vol. 20, no. 3, pp. 45-50, 2001.

[9] M. Kachuee, S. Fazeli and M. Sarrafzadeh, "ECG Heartbeat Classification: A Deep Transferable," *arXiv,* 2018, arXiv:1805.00794v2.

[10] A. L. Goldberger , L. Amaral, L. Glass, J. M. Hausdorff, P. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. Peng and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation 101(23):e215-e220 [Circulation Electronic Pages; http://circ.ahajournals.org/content/101/23/e215.full],* 2000.

[11] W. Zhang, L. Yu, L. Ye, W. Zhuang and F. Ma, "ECG Signal Classification with Deep Learning for Heart Disease Identification," *International Conference on Big Data and Artificial Intelligence,* 2018.

[12] W. Fu, J. Tan, Y. Xu, K. Wang and T. Chen, "'Fault diagnosis for rolling bearings based on fine-sorted dispersion entropy and SVM optimized with mutation SCA-PSO," *Entropy,* vol. 21, no. 4, p. 404, 2019.

[13] S. Huang, N. Cai, P. Pacheco, S. Narrandes, Y. Wang and W. Xu, "Applications of Support Vector Machine (SVM) Learning in Cancer Genomics," *CANCER GENOMICS & PROTEOMICS,* vol. 15, pp. 41-51, 2018.

[14] Y. Freund and R. E. Schapire, "A desicion-theoretic generalization of online learning and an application to boosting," *in Proc. Eur. Conf. Comput.,* pp. 23-37, 1995.

[15] L. Breiman, "Random forests," *Mach. Learning,* vol. 45, no. 1, pp. 5-32, 2001.

[16] D. E. a. L. W. P. Geurts, "Extremely randomized trees," *Machine learning,* vol. 63, pp. 3-42, 2006, https://doi.org/10.1007/s10994-006-6226-1.

[17] S. Nembrini, I. R. Konig, and M. N. Wright, "The revival of the Gini importance?," *Bioinformatics,* vol. 34, no. 21, p. 3711–3718, 2018.

[18] M. Idhammad, K. Afdel and M. Belouch, "Detection System of HTTP DDoS Attacks in a Cloud Environment Based on Information Theoretic Entropy and Random Forest," *Security and Communication Networks,* 2018, https://doi.org/10.1155/2018/1263123.

[19] M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, and F. Mohammadimanesh, P. Ghamisi and S. Homayouni, "Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review," *in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 13, pp. 6308-6325, 2020, doi: 10.1109/JSTARS.2020.3026724.

[20] T. Hastie, R. Tibshirani and J. Friedman, "Random Forests," *The Elements of Statistical Learning. Springer Series in Statistics,* 2009.

[21] "UCI Heart Disease Dataset," *[online], Accessed: April 26, 2022, Available: https://www.kaggle.com/datasets/ketangangal/heart-disease-dataset-uci.*