

Heart Failure Prediction Using Variational Auto-Encoder and Extreme Gradient Boosted Neural Network

Rabi Mustapha¹, Yazid Ali Hassan², Muhammad Aminu Ahmad³ and Habeebah Adamu Kakudi⁴

^{1,3}Department of Computer Science, Kaduna state University, Kaduna.

²Directorate of ICT, Kaduna state University, Kaduna.

⁴Department of Computer Science, Faculty of Computer Science and Information Technology, Bayero University Kano, Nigeria

rabichubu@kasu.edu.ng yazid.hassan@kasu.edu.ng, muhdaminu@kasu.edu.ng, hakakudi.cs@buk.edu.ng

Abstract

This paper pioneers the integration of Variational Auto-encoder (VAE) techniques into the XGBNet model for heart failure prediction, leveraging the largest combined heart failure dataset from Kaggle. Through rigorous evaluation over 100 epochs, the model achieved a 92% prediction accuracy in distinguishing patients with potential heart failure. Comparative analysis revealed a 2-3% increase in accuracy over previous methodologies, highlighting the efficacy of VAE in tandem with XGBoost. These findings underscore the superiority of the conjugate gradient algorithm as an optimizer and its potential implications for healthcare providers, promising enhanced accuracy in early heart failure prediction for proactive intervention strategies and improved patient care.

Keywords: Heart failure, predictive system, Extremely Gradient Boosted Neural Network, conjugate gradient descent, medical datasets, xgboost, variational auto-encoders, early detection, cardiovascular

1. Introduction

The heart, a vital muscular organ, serves as the epicenter of the body's cardiovascular system, orchestrating the circulation of blood through an intricate network of blood vessels, including veins, arteries, and capillaries (Shadman et al., 2018). Cardiovascular diseases (CVD), encompassing a spectrum of ailments such as Myocardial infarction, Myocardial ischemia, Congenital heart disease, Coronary heart disease, Cardiac arrest, and Peripheral heart disease, have emerged as a leading cause of global mortality (Polaraju & Durga, 2017). The effective detection and diagnosis of specific cardiac conditions, particularly Coronary Heart Disease (CHD), stand as imperative measures to avert human casualties (Bayu et al., 2020). Despite health professionals' efforts in adopting early detection strategies, the increasing complexity of heart-related issues and associated symptoms pose challenges to successful identification at early stages (Dangare & Apte, 2012). Consequently, there arises a necessity for the development of predictive systems leveraging Artificial Intelligence (AI) techniques to facilitate early and accurate diagnosis, paving the way for prompt and appropriate treatment processes. Insufficient training data poses a significant challenge in deep learning, adversely affecting the accuracy of predictions and leading to overfitting of models, as emphasized by Celik (2022). To address this limitation and enhance the reliability of predicting heart failure, the research underscores the importance of adopting more efficient and dependable algorithms. In response, this study pioneers the development of an advanced deep learning model that strategically integrates the strengths of xgbnet and variational auto-encoders for data augmentation. By seamlessly combining these techniques, the model aims to predict heart failure effectively without the need for human intervention. The primary aim of this

research is to contribute to the prediction of heart failure by leveraging a XGBNet model, empowered by VAE for augmentation. The objectives outlined to achieve this aim encompass the application of variational auto-encoder to enrich the heart failure dataset, the design and implementation of the XGBNet model for predictive analysis, and a comprehensive comparison of the model's performance metrics with those presented by Celik (2022). The objectives are intricately aligned to ensure a holistic investigation into the effectiveness of the proposed model and its potential advancements over existing methodologies.

2. Related work

Heart failure encompasses a spectrum of cardiac and vascular conditions affecting blood circulation. Diagnosis is crucial but challenging, demanding accuracy and efficiency (Al-Milli, 2013). Various studies have explored leveraging machine learning techniques such as decision trees, support vector machines, deep learning, Convolutional Neural Networks (CNN), among others, to address heart disease issues. Existing literature is surveyed to pinpoint research gaps.

Vanisree and Jyothi (2011) proposed a Decision Support System for diagnosing Congenital Heart Disease, employing MATLAB's GUI and a Backpropagation Neural Network achieving 80% accuracy. Milan and Sunila (2011) evaluated data mining techniques like RIPPER, Decision Trees, Artificial Neural Networks (ANNs), and Support Vector Machines (SVM) on cardiovascular data, finding SVM most accurate. Sayed and Halkarnikar (2014) introduced a genetic neural-based algorithm reaching 89% accuracy using MATLAB. Jaymin et al. (2016) compared data mining approaches, with Decision Tree-C5.0 yielding 93.02% accuracy. Shrinivas et al. (2019) experimented with classifiers for heart disease prediction, finding Logistic Regression most effective (92.58% accuracy). Subhadra & Vikas (2019) developed a multi-layered neural network achieving 93.39% accuracy.

Recent studies have focused on heart failure's prevalence, leading to the curation of specialized datasets (Dua & Graff, 2019). Ridwan et al. (2021) classified heart failure using the Naïve Bayes algorithm (86.18% accuracy). Minh et al. (2021) developed an MLP model achieving 87% accuracy. Wang (2021) compared 18 ML algorithms, identifying SVM as the most accurate (86.67%).

Efforts to enhance heart failure detection led Celik (2022) to develop a Deep Neural Network (DNN) architecture, achieving 90.2% accuracy incorporating age and gender features. Research reveals a gap where traditional ML algorithms often outperform neural networks in heart disease prediction due to small datasets. Only Celik (2022) reached 90% accuracy using DNN. This study aims to combine XGBoost with a feed-forward neural network and VAE for dataset scaling, aiming to improve heart failure prediction accuracy, benchmarked against Celik's (2022) DNN model. The literature reviewed revealed that several machine learning techniques have been used for predicting of different heart related diseases. Predictions of heart disease have shown to produce a higher accuracy as compared to heart failure. Also, among several literatures reviewed, other machine learning algorithms outperformed deep learning method (i.e neural network). This may be because the dataset for heart related diseases are small and other machine learning algorithms gave a better interpretation. Also, it was observed that only Celik (2022) who used DNN, achieved an accuracy of upto 90%. This research will combine the strength of xgboost that have proven to perform extremely well in competitions involving tabular data (Memon et al., 2019) with a feed-forward neural network to make the model robust for all performance metrics. VAE will also be used to scale the dataset solving the problem of small data size. The result will be compared with the research work of Celik (2022) who used DNN in the classification of heart failure.

3. Methodology

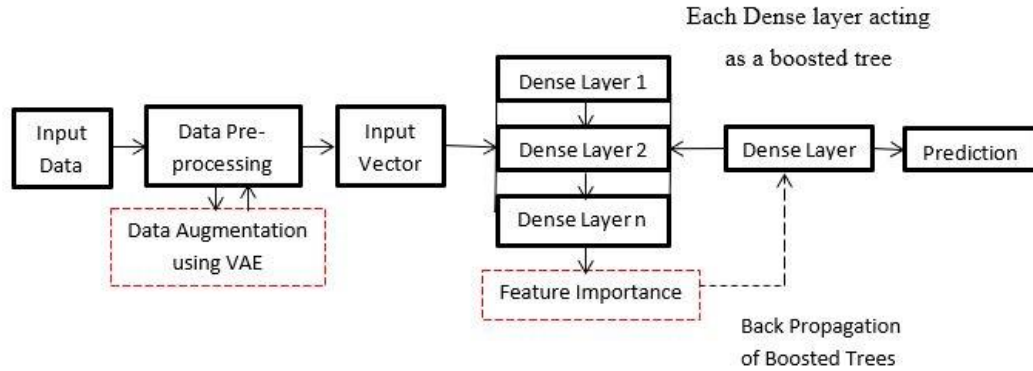


Figure 1: Proposed Model

A. Input Data

The data was gotten from kaggle heart failure prediction dataset. The dataset was created by combining different dataset already available independently but not combined before. In this dataset, 5 heart data were combined over 11 common features which makes it the largest heart failure dataset available. The five-dataset used for its curation are:

- Cleveland: 303 observation observations
- Hungarian: 294 observations
- Switzerland: 123 observations
- Long Beach VA: 200 observations
- Stalog (Heart) Data set: 270 observations

Total: 1190 observations, Duplicated: 272 observations, Final dataset: 918 observations

The heart failure dataset used can be found under the Index of heart failure dataset from UCI Machine Learning Repository 8 (Dua & Graff, 2019). The attributes description of the data is shown in Table 3.1

Table 3.1: Data Attributes and Descriptions

S/N	Attribute	Description	Values
1	Age	Age of the Patient	Years
2	Sex	Sex of The Patience	M: Male, F: Female

3	ChestPainType	Chest Pain Type	TA: Typical Angina ATA: Atypical Angina NAP: Non Anginal Pain ASP: Asymptomatic
4	RestingBP	Resting Blood Pressure	mm/Hg
5	Cholestorl	Serum Cholesterol	mm/dl
6	FastingBP	Fasting blood Sugar	1: if FastigBS > 120 mg/dl, 0: otherwise 0 for ≤ 120 mg/dl
7	RestingECG	Resting Electrocardiographic Result	Normal: Normal, ST:having ST-T wave abnormality(T wave inversions and/or ST elevation or depression of > 0.05Mv), LVH: showing probable or definite left ventricular hypertrophy by Estes'criteria
8	MaxHR	Maximum Heart Rate achieved	Numeric value between 60 and 202
9	Exercise Angina	Exercise Induced Angina	Y: Yes, N: No
10	Oldpeak	Oldpeak = ST (Numeric value measured in depression)	Continuous value
11	ST_Slope:	Slope of the peak exercise ST sement	Up: upsloping, Flat: flat, Down: downsloping
12	Possible Heart Failure:	Output class	1: Yes 0: normal

B. Data Augmentation

The proposed model aimed to explore the application of data augmentation techniques, specifically using a Variational Autoencoder (VAE), to improve the performance of models when dealing with tabular datasets. The VAE architecture consists of an encoder network, a decoder network, and a latent space (Mahendiran & Subramaniam). The encoder network maps the input data, such as features from a heart failure dataset, to the latent space by learning the mean and variance parameters of a probability distribution representing the latent

variables. The decoder network reconstructs the original input data by mapping samples from the latent space back to the original data space as shown in Figure 2. Data augmentation introduces additional variations and diversity into the training data, enabling the model to better generalize to different instances and variations of heart failure patterns. This reduces overfitting and enhances the model's ability to generalize to unseen data.

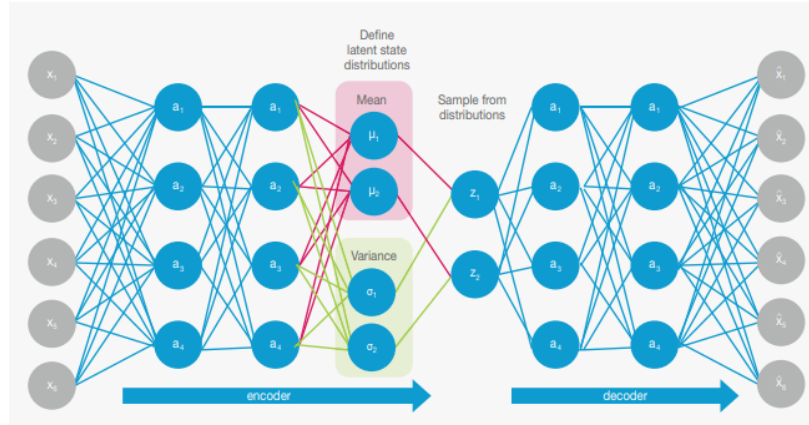


Figure 2: The proposed VAE

C. Classification

The model to be built in this research will be based on XBNNet by 10 Sarkar & Shah (2022) which combine tree-based models with neural networks to create a robust architecture trained by using a novel optimization technique that boost [gradient descent](#) for tabular data which increases its interpretability and performance. Our model also applied VAE so as to boost the performance of the model instead just taking raw data as input. Our XBNNet takes the encoded and augmented tabular data as input, and the model was trained using extreme boosted gradient descent which is initialized with the help of feature importance of a gradient boosted tree to further updates the weights of each layer in the neural network as shown in Figure 2. The combination of XGBoost and neural networks in XBNNet’s ensemble learning approach allows for the combination of multiple neural networks, enhancing the overall predictive power and generalization ability of the model.

D. Performance Evaluation

Performance evaluation is the process of assessing how well a model performs against real data. It primarily applies the model to test data in order to determine whether the model, built on a training set, is generalizable to other data. In particular, it helps to avoid the phenomenon of overfitting, which can occur when the model is trained and tested on the same data and fits the built data too well but performs poorly on a different data (Ritchie, 2018). The research will make use of accuracy, sensitivity, specificity and Area Under Receiver Operating Curve (AUROC).

i. Performance Metrics

Performance metrics of a model are obtained from values of it predictions. The performance metrics of our neural network is gotten from values of prediction in terms of its True positive (TP) value, True Negative (TN) value, False positive (FP) value, False negative (FN) value as obtained from the confusion matrix. They are defined as follow:

1. TP - shows the condition when condition is present.
2. TN - shows no condition when condition is not present.
3. FP - shows condition when condition is not present.
4. FN - shows no condition when condition is present.

ii. Confusion Matrix

A confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one. Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class. The name stems from the fact that it makes it easy to see if the system is confusing two classes. An example of confusion matrix is shown in Table 3:

Table 3.2 Confusion Matrix for two class classifier

		PREDICTED	
		Positive	Negative
ACTUAL	Positive	A (TP)	B (FN)
	Negative	C (FP)	D (TN)

4. Result and Discussions

A. Results of Experiments

. Figure 3 provides a detailed elaboration of this class distribution, enabling a better understanding of the dataset's composition and potential class imbalances.

Class 0: Disease: 410
 Class 1: Normal: 508



Figure 3: Target Class Distribution

To ensure the effectiveness of our training process, we operated on a substantial batch size of 1024, optimizing computational efficiency while maintaining the robustness of our model. In addition to these architectural decisions, the critical task of determining the ideal learning rate was addressed. This optimization process is graphically illustrated in Figure 4 which serves as a visual representation of our quest for the most suitable learning rate. Through the application of the Learning Rate Finder technique, we explored a range of learning rates, evaluating their impact on our model's performance. The outcome of this search revealed a critical revelation pertinent to our VAE. It identified the optimal learning rate for our specific problem as being equal to 10^0 . This optimal learning rate selection was a significant milestone in our research, as it directly contributed to the model's convergence and overall success.

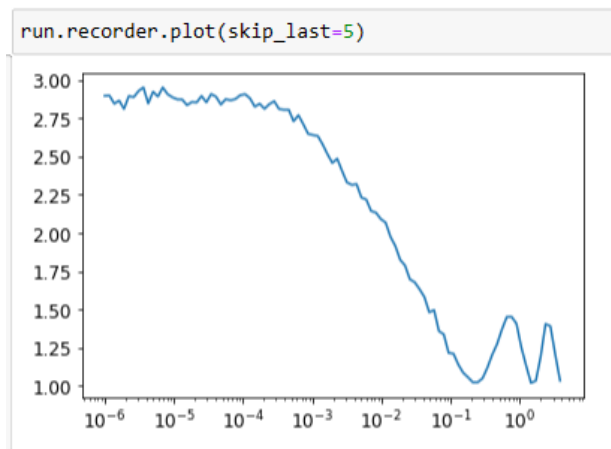


Figure 4: Learning Rate Finder

```
In [1164]: cbfs = [partial(dta.LossTracker, show_every=50), dta.Recorder, partial(dta.ParamScheduler, 'lr', sched)]
model = dta.Autoencoder(D_in, VAE_arch, latent_dim=20).to(device)
opt = optim.Adam(model.parameters(), lr=0.01)
learn = dta.Learner(model, opt, loss_func, data, target_name, target_class, df_cols)
run = dta.Runner(cb_funcs=cbfs)
run.fit(400, learn)

epoch: 50
train loss is: 4.957300186157227
validation loss is: 1.4340226650238037
epoch: 100
train loss is: 1.6409997940063477
validation loss is: 1.1541900634765625
epoch: 150
train loss is: 1.3099443912506104
validation loss is: 1.0800868272781372
epoch: 200
train loss is: 1.196912407875061
validation loss is: 1.053731918334961
epoch: 250
train loss is: 1.1400948762893677
validation loss is: 1.0403523445129395
epoch: 300
train loss is: 1.105884075164795
validation loss is: 1.0323100090026855
epoch: 350
train loss is: 1.0830448865890503
validation loss is: 1.0268447399139404
epoch: 400
train loss is: 1.0666910409927368
validation loss is: 1.02298104763031
```

Figure 5: Loss for the VAE Model

. Figure 5 visually illustrates the training process under 50 epoch and its corresponding performance, providing a snapshot of the dynamic nature of our model's learning process.

The outcomes of this training are showcased in Figure 6, where a comprehensive comparison between the original training dataset comprising 734 samples and the augmented dataset generated through our VAE model, consisting of 802 samples, is presented. This augmentation strategy effectively expanded our dataset with additional instances. Such augmentation is pivotal in enriching the dataset's diversity and consequently enhancing generalizability of our model, a critical milestone in our research efforts.

```
In [1231]: X_train.shape, X_train_aug.shape
Out[1231]: ((734, 11), (802, 11))

In [1232]: y_train.shape, y_train_aug.shape
Out[1232]: ((734,), (802,))
```

Figure 6: Distribution of Original and Augmented Training Samples

B. Model Performance of XGBNet

During the training phase, the model was partitioned into 80% for training and 20% for testing on the dataset for 100 epochs. Specifically, the model demonstrated impressive performance in predicting the possibility of heart failure, achieving an accuracy rate of 92% as shown in Figure 7. The loss value of 0.1985 indicates the amount of error or discrepancy between the predicted and actual values. A lower loss value signifies a better alignment between the model's predictions and the ground truth.

```
In [125]: # fit the model to the training data
#es = EarlyStopping(monitor='accuracy', mode='max', verbose=0, patience=60)
#history=model.fit(X_train, Y_train, validation_data=(X_test, Y_test),epochs=500, batch_size=10)
#from tensorflow.keras.utils import to_categorical
#y_binary = to_categorical(y_int)
H = model.fit(
    X_train, y_train, batch_size=BS,
    steps_per_epoch=len(X_train) // BS,
    validation_data=(X_test, y_test),
    validation_steps=len(X_test) // BS,
    epochs=EPOCHS )

Epoch 91/100
45/45 [=====] - 0s 4ms/step - loss: 0.2805 - acc: 0.8908 - val_loss: 0.2302 - val_acc: 0.9293
Epoch 92/100
45/45 [=====] - 0s 4ms/step - loss: 0.2740 - acc: 0.8934 - val_loss: 0.2243 - val_acc: 0.9293
Epoch 93/100
45/45 [=====] - 0s 4ms/step - loss: 0.2751 - acc: 0.8908 - val_loss: 0.2224 - val_acc: 0.9293
Epoch 94/100
45/45 [=====] - 0s 4ms/step - loss: 0.2737 - acc: 0.8900 - val_loss: 0.2208 - val_acc: 0.9293
Epoch 95/100
45/45 [=====] - 0s 4ms/step - loss: 0.2666 - acc: 0.8953 - val_loss: 0.2179 - val_acc: 0.9293
Epoch 96/100
45/45 [=====] - 0s 4ms/step - loss: 0.2701 - acc: 0.8908 - val_loss: 0.2150 - val_acc: 0.9293
Epoch 97/100
45/45 [=====] - 0s 4ms/step - loss: 0.2663 - acc: 0.8924 - val_loss: 0.2067 - val_acc: 0.9293
Epoch 98/100
45/45 [=====] - 0s 4ms/step - loss: 0.2611 - acc: 0.8995 - val_loss: 0.2036 - val_acc: 0.9348
Epoch 99/100
45/45 [=====] - 0s 4ms/step - loss: 0.2538 - acc: 0.8987 - val_loss: 0.1977 - val_acc: 0.9293
Epoch 100/100
45/45 [=====] - 0s 4ms/step - loss: 0.2592 - acc: 0.8976 - val_loss: 0.1985 - val_acc: 0.9239
```

Figure 7: Accuracy and Loss of the Model after 100 Epoch

These results indicate that the trained model was effective in accurately predicting the possibility of heart failure, showcasing its potential as a valuable tool in healthcare applications. The combination of high accuracy and low loss suggests that the model was able to capture and learn important patterns and features from the dataset, leading to accurate predictions. The developed model exhibited impressive capabilities in predicting the likelihood of heart failure, as demonstrated in Figure 8 by its high accuracy rate of 92%.

```
In [126]: # generate classification report using predictions for categorical model
from sklearn.metrics import classification_report, accuracy_score

categorical_pred = np.argmax(model.predict(X_test), axis=1)
y_test = np.argmax((y_test), axis=1)

print(accuracy_score(y_test, categorical_pred))
print(classification_report(y_test, categorical_pred))
```

```
0.9239130434782609
```

	precision	recall	f1-score	support
0	0.90	0.94	0.92	82
1	0.95	0.91	0.93	102
accuracy			0.92	184
macro avg	0.92	0.93	0.92	184
weighted avg	0.93	0.92	0.92	184

Figure 8: Classification Report of the Model

The results obtained from the experiment revealed that our model achieved a prediction accuracy of 92% for heart failure cases. Additionally, the proposed model demonstrated a precision of 92%, indicating that out of the 102 predicted heart failure cases, only 9 were incorrectly classified. These findings were reflected in the confusion matrix shown in Figure 10.

To assess the efficacy of our proposed model, we conducted a comparative analysis. Firstly, we benchmarked our model against the findings from 3 Celik (2022), who utilized a DNN for heart failure prediction. In addition, we performed a parallel comparison with XGBNet and other machine learning algorithms, but in this case, we excluded the use of VAE.

The outcomes of these comparisons are succinctly presented in Table 2, showcasing a comprehensive evaluation of performance metrics across three models. This evaluation encompasses key metrics such as precision, recall, F1-score, and accuracy, enabling a comprehensive understanding of the strengths and weaknesses of each approach.

Table 1: Comparison Table for the Performance Metrics of the Models

Method	Accuracy (%)	Precision	Recall	F1-Score
XGBoost	84.78	0.84	0.84	0.84
SVM	89.13	0.89	0.89	0.89
DNN by Celik (2022)	90.22	0.90	0.90	0.90
XBNet without VAE	89.13	0.89	0.89	0.89
Proposed Model	92.39	0.92	0.93	0.92

The results of Table 1 reveal significant findings into the comparative performance of other distinct models for heart failure prediction. These models, including 3 Celik's 2022 model of DNN, XGBNet without VAE, SVM, XGBoost and our Proposed Model, have undergone a comprehensive assessment through key performance metrics, namely precision, recall, F1-Score, and accuracy. precision, recall, F1-Score, and accuracy. Starting with 3 Celik's model, it is evident that the model exhibits commendable performance across all metrics, with precision, recall, F1-Score, and accuracy all achieving a balanced 90% as compared with SVM with 89% and XGBoost having about 85%. This demonstrates the model's reliability in making accurate predictions and detecting heart failure cases.

The XGBNet model, implemented without the integration of VAE, also exhibits robust performance. With precision, recall, F1-Score, and accuracy scores of 89%, the model demonstrates consistent and reliable prediction capabilities, albeit with a marginal reduction in performance compared to 3 Celik's model.

5.0 Conclusion

In this dissertation, we proposed the use of VAE method as an augmentor in training XGBNet for the heart failure prediction. The model designed and implemented was evaluated on heart dataset from kaggle which is a combination of five data set. It is the largest heart failure dataset available so far for this research purpose. The training and testing of the proposed model was carried out on 100 epochs. The model was able to predict (classify) whether a patient can have a heart failure (present) or not (absent). When compared to the previous technique, the prediction accuracy of our prediction model is relatively higher by 2 - 3% in some instance. The accuracy obtained for our model was 92%.

It is clear from the results obtained that VAE on XGBoost algorithm performs efficiently in training neural network for the heart failure prediction. So health providers can safely make this conclusion that conjugate gradient algorithm is a better optimizer than most of the implementation of other techniques

Based on the findings of this research, which demonstrated the superior performance of our model for heart failure prediction, the following recommendations are suggested:

- i. Adoption of VAE as an augmentor: Health providers and researchers should consider utilizing VAE as an augmentor when training Neural Networks for heart failure prediction. The results indicate indicated the improved performance compared to previous techniques.

- ii. Comparative Studies: Comparative studies between other augmentation techniques like Generative Adversarial Network (GANs) can provide deeper insights into the strengths and weaknesses of different augmentation methods for heart failure prediction. This would help researchers and practitioners select the most appropriate augmentation approach based on the specific requirements and constraints of their applications.

References

- Polaraju, K & Durga, P. (2017). Prediction of Heart Disease using Multiple Linear Regression Model. *International Journal of Engineering Development and Research Development*. ISSN:2321-9939.
- Shadman, R., Poole, J. E., Dardas, T. F., Mozaffarian, D., Cleland, J. G., Swedberg, K., ... & Levy, W. C. (2018). A novel method to predict the proportional risk of sudden cardiac death in heart failure: derivation of the Seattle Proportional Risk Model. *Heart Rhythm*, 12(10), 2069-2077.
- Celik, G. (2022). Prediction of Heart Failure Disease with a Proposed Method Based on Deep Neural Networks. *Ankara/Turkey*, 279.
- Minh, L. T., Mai, L., and Dao, S. V. (2021). Predicting heart failure using wrapper based feature selection. *Indonesian Journal of Electrical Engineering and Computer Science*, 21(3), 1530-1531
- Vanisree, K. & Jyothi, S. (2011). Decision Support System for Congenital Heart Disease Diagnosis based on Signs and Symptoms using Neural Networks. *International Journal of Computer Applications* 19(6).
- Milan, K., & Godara, S. (2011). Comparative Study of Data Mining Classification Methods in Cardiovascular Disease Prediction. *IJREE* 2(2).
- WestTennessee Healthcare (2021, October). Heart Disease... Heart Failure. *What is the Difference?* *Cardiology West Tennessee Medical Group*. <https://www.wth.org/blog/heart-diseaseheart-failure-what-is-the-difference/>
- Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science. Cleveland database: <http://archive.ics.uci.edu/ml/datasets/Heart+Disease>
- Sarkar, T. & Shah, D. (2022). Modelly: An open source all in one python package for developing machine learning models. *Software Impacts*, 14, 100407.