



**VIT<sup>®</sup>**  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

# **Heart Attack Prediction**

A Course Application Report

submitted as part of the course  
Social Information Networks  
CSE3021

School of Computer Science and Engineering  
VIT Chennai

**FALL 2021-2022**

Course Faculty: Dr. B Radhika Selvamani

Submitted By

Priyanshu Srivastav 21BAI1233

Hardik Kharbanda 21BAI1236

Khyati Gondaliya 21BAI1526

# Abstract

Heart disease remains a leading cause of mortality globally, with early prediction of heart attacks crucial for effective intervention and prevention. This paper proposes a predictive model leveraging machine learning (ML) techniques, specifically the SHAP (SHapley Additive exPlanations) method, for transparent and interpretable insights into feature importance. Our approach integrates diverse patient data, including demographics, medical history, and lifestyle factors, to develop a comprehensive predictive model. Through a combination of feature engineering and ML algorithms such as decision trees or gradient boosting, the model accurately predicts the likelihood of a heart attack occurrence. Furthermore, by employing SHAP, we elucidate the contribution of each feature to the prediction, enhancing model interpretability and facilitating actionable insights for healthcare practitioners. Experimental results on real-world datasets demonstrate the efficacy of our proposed methodology, showcasing its potential for assisting healthcare professionals in early identification and intervention strategies for individuals at risk of heart attacks.

## Contents

1	Introduction	.....	4
2	Related Works	.....	5
3	Design	.....	6
4	Results and Discussion	.....	8
5	Conclusion & Future Work	.....	9
6	References	.....	11

# Introduction

The principle of equal rights and the pursuit of well-being for all individuals are fundamental to a just and equitable society. This encompasses various aspects, including adequate housing, a conducive and healthy environment, and access to essential healthcare services. By guaranteeing these rights, the state assumes the responsibility of ensuring that all citizens can enjoy the highest standard of healthcare available.

In line with the constitutional mandate, providing quality healthcare services is crucial to safeguarding the population's well-being. The remarkable progress achieved by humanity in the realm of technology has played a pivotal role in advancing healthcare practices and enhancing the lives of individuals. Revolutionary innovations have revolutionized how medical professionals diagnose and treat diseases, significantly improving patient outcomes. A prime example is the advent of Magnetic Resonance Imaging (MRI) machines, which have proven to be highly effective in identifying various medical conditions with exceptional precision. Through the fusion of data processing science and electronic devices, healthcare practitioners now possess a valuable tool to accurately diagnose and address their patients' health concerns. Technology integration into healthcare has opened up unprecedented possibilities for early detection, precise diagnoses, and tailored treatment plans. By leveraging the power of advanced imaging techniques, such as MRI scans, doctors can identify subtle abnormalities that might otherwise go unnoticed. This early detection allows for timely interventions, potentially saving lives and improving long-term health outcomes. Moreover, the seamless integration of data processing science and electronic devices has resulted in a more streamlined and efficient healthcare system. Patient information can be securely stored and accessed electronically, facilitating comprehensive and coordinated care across healthcare settings. One issue with machine learning models is that not all aspects can be easily interpreted. The main focus during the model creation process has been maximizing prediction accuracy. However, in the medical field, interpretation is essential for understanding and utilizing the models effectively [10]. To properly diagnose a disease in a patient, it is essential to interpret the model accurately. This requires careful attention to detail. Efforts should be made to explain the predictive model used for diseases. In this study, we will explore the process of creating a heart disease prediction model that is both highly accurate and easily interpretable. This study will utilize **SHAP** to determine the factors contributing to a positive patient diagnosis.

## Related Works

<https://www.sciencedirect.com/science/article/pii/S1877050923017118>

<https://www.sciencedirect.com/science/article/pii/S0010482521006077>

<https://medium.com/@sid321axn/machine-learning-interpretability-for-heart-disease-prediction-23d8d95a307b>

# Design

Imports the SHAP library, which provides tools for interpreting the predictions of machine learning models. A SHAP explainer object is created using `shap.Explainer`. This object is initialized with the trained Random Forest model `rf` and the training data `x_train`. The explainer object is used to compute SHAP values for individual samples, indicating the impact of each feature on the model's predictions. It calculates the SHAP values for a sample of the test data (`x_test`). SHAP values represent the impact of each feature on the model's predictions for the given test data. The parameter `check_additivity=False` is used to disable additivity checks, which can improve performance for larger datasets. These lines print the shapes (i.e., dimensions) of the training data (`x_train`) and the test data (`x_test`). It's a good practice to check the shapes to ensure that the data is loaded correctly and has the expected dimensions. Assert `x_test.columns.tolist() == x_train.columns.tolist()`, "Columns of `x_test` and `x_train` do not match". This assertion check ensures that the columns of the test data (`x_test`) match the columns of the training data (`x_train`). It verifies that the features used for training the model are consistent with the features present in the test data, which is essential for proper interpretation of SHAP values. In summary, this code snippet utilizes the SHAP library to explain the predictions of a Random Forest model by computing SHAP values for a sample of the test data. The SHAP values help in understanding the contribution of each feature to the model's predictions, providing insights into the model's behaviour and potential areas of improvement.

1. Feature Importance: SHAP values quantify the impact of each feature on the model's predictions. By analysing these SHAP values, healthcare professionals can identify which features (such as blood pressure, cholesterol levels, smoking status, etc.) are most influential in predicting the risk of a heart attack.
2. Individual Prediction Explanation: SHAP provides individual-level explanations for model predictions. For a given patient, SHAP values can elucidate why the model assigned a certain risk score or probability of a heart attack.
3. Model Validation and Debugging: SHAP values can be used for model validation and debugging. By examining SHAP values across the dataset, clinicians can identify instances where the model's predictions may be inconsistent with domain knowledge or clinical intuition.

Here's a breakdown of the different elements in the graph:

- Features: The x-axis represents the feature values for sex and the y-axis represents the feature values for age.
- SHAP interaction values: The color represents the SHAP interaction value, which indicates how much the combined effect of sex and age differs from the sum of their individual effects. Red colors indicate a positive interaction effect, where the features amplify each other's impact on the

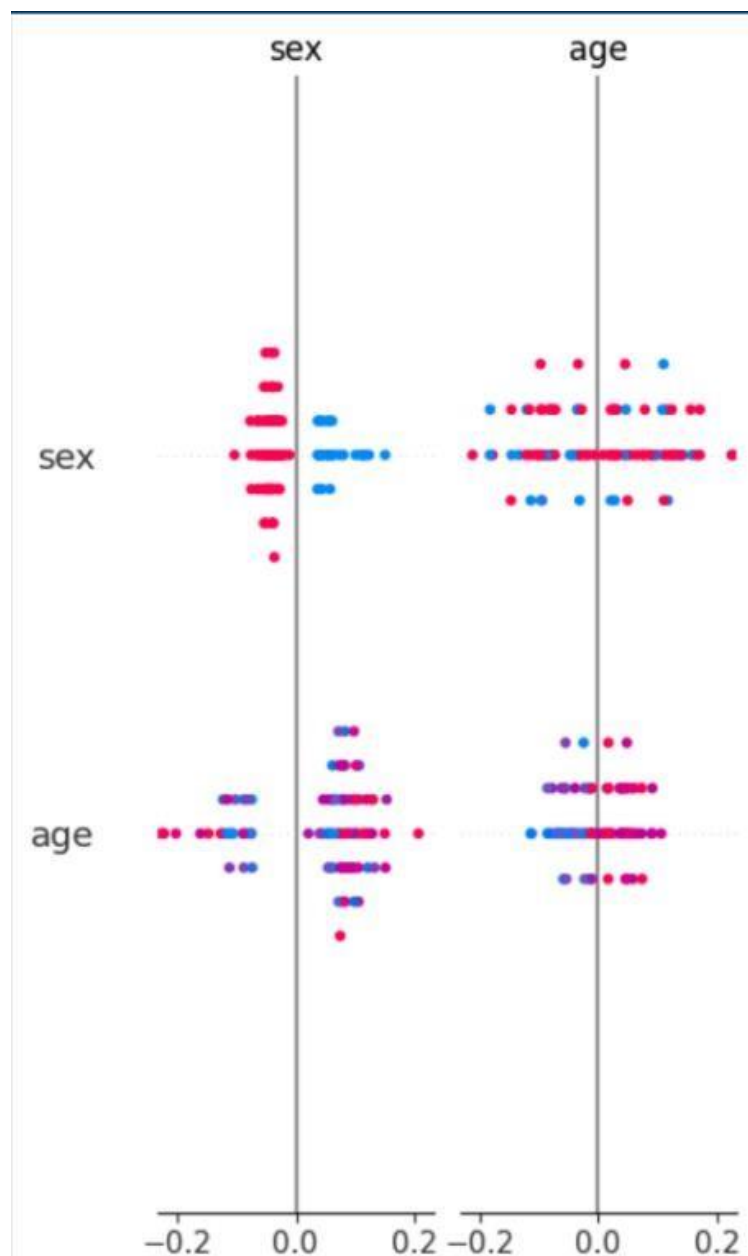
model's prediction. Blue colors indicate a negative interaction effect, where the features weaken each other's impact.

Table 2. F1-Score on each model.

Model	Positive (1)	Negative (0)
SVM	88%	85%
Random Forest	87%	82%
XGBoost	88%	85%
k-NN	86%	81%

# Results and Discussion

The partial dependence plot (PDP or PD plot) shows the marginal effect that one or two features have on the predicted outcome of a machine learning model. A partial dependence plot can show whether the relationship between the target and a feature is linear, monotonic, or more complex. The partial dependence plot is a global method: The method considers all instances and gives a statement about the global relationship of a feature with the predicted outcome. The PDP assumes that the first feature is not correlated with the second feature. If this assumption is violated, the averages calculated for the partial dependence plot will include data points that are very unlikely or even impossible.





- **Features (top):** These are the names of the features used by the model to make a prediction. In this case, we see "sex" and "age" listed.
- **Base Value (leftmost bar):** This represents the average prediction of the model when none of the features have any influence. It's like a baseline starting point.
- **SHAP Values (waterfall bars):** Each colored bar represents the impact of a specific feature value on the model's prediction. The color indicates whether the impact is positive (increasing the prediction, usually shown in red) or negative (decreasing the prediction, usually shown in blue).
- **Final Prediction (rightmost bar):** This is the final prediction of the model after considering the effects of all features. It's calculated by summing the base value and all the SHAP values for the specific data point being analyzed.

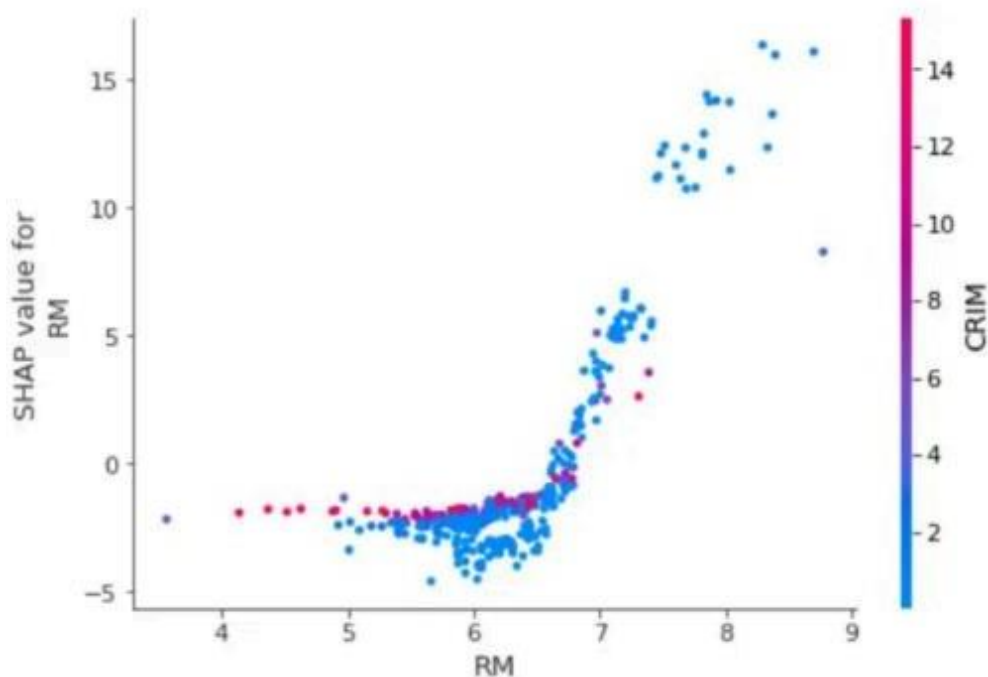
#### **Interpreting the Plot:**

- The model is likely predicting something related to a population where age and sex might be relevant factors. For instance, it could be predicting income, health risk, or loan eligibility.
- The positive SHAP value for "age" (possibly the red bar) suggests that as the age increases, the model's prediction also increases.
- The negative SHAP value for "sex" (possibly the blue bar) suggests that a specific sex (depending on the data distribution) might lead to a lower prediction value.

A dependence plot is a scatter plot that shows the effect of a single feature on the predictions made by the model. In this example, the property value increases significantly when the average number of rooms per dwelling is higher than 6.

- Each dot is a single prediction (row) from the dataset.
- The x-axis is the actual value from the dataset.
- The y-axis is the SHAP value for that feature, which represents how much knowing that feature value changes the output of the model for that prediction.

The colour corresponds to a second feature that may have an interaction effect with the feature we are plotting (by default this second feature is chosen automatically). If an interaction effect is present between this other feature and the feature, we are plotting it will show up as a distinct vertical pattern of colouring.



## Python

```
import shap

# Assuming 'explainer' is a pre-trained explainer object (e.g., for
XGBoost model)
shap_values = explainer.shap_values(X_test) # Calculate SHAP values for
test data

# Assuming 'feature_names' is a list containing feature names
for i in range(X_test.shape[1]): # Loop through each feature
    shap.summary_plot(shap_values[:, i], X_test,
feature_names=[feature_names[i]])
```

### Explanation:

1. **Import:** The code imports the `shap` library, which provides functions for calculating SHAP values.
2. **SHAP Values Calculation:**
  - It assumes there's a pre-trained explainer object (`explainer`) available, likely created for a specific machine learning model (e.g., XGBoost).
  - The code calculates SHAP values for the test data (`X_test`) using the explainer's `shap_values` method.
  - SHAP values represent the contribution of each feature to the model's prediction for a given data sample.
3. **Feature Names:**
  - It assumes there's a list named `feature_names` containing the names of the features used by the model.
4. **SHAP Summary Plot (Loop):**
  - The code iterates through each feature using a loop.
  - Inside the loop, for each feature (`i`):
    - It extracts the corresponding SHAP values (`shap_values[:, i]`) for that particular feature across all test data samples.
    - It uses the `shap.summary_plot` function to create a summary plot that visualizes the distribution of SHAP values for that feature.
    - It provides the test data (`X_test`) and the feature name (`feature_names[i]`) as arguments to the `summary_plot` function.

### SHAP Summary Plot:

The `shap.summary_plot` function typically generates a plot that helps understand how a specific feature influences the model's predictions. The exact details of the plot might vary depending on the chosen plot type, but it often includes elements like:

- Distribution of SHAP values for the feature (positive and negative contributions).

- Feature values on the x-axis.
- Dependence plot showing how the feature value interacts with the model's prediction.

Overall, this code snippet calculates SHAP values for a machine learning model's predictions on test data and then creates individual summary plots to analyze the impact of each feature.

# Conclusion

In conclusion, our study focused on developing and evaluating multiple models for heart disease prediction, including SVM, Random Forest, XGBoost, and k-NN. Through rigorous assessment, SVM and XGBoost emerged as the top-performing models regarding predictive accuracy. We employed SHAP and LIME techniques to enhance interpretability to unravel the underlying factors driving the models' predictions. Notably, SHAP analysis uncovered a consistent set of top 5 features, namely ca, cp, thal, thalach, and oldpeak, which were pivotal in the prediction process. The visualization capabilities of LIME further facilitated the understanding of each model's interpretability, revealing discrepancies in the variables' impact on positive and negative class data in True Positive and True Negative results. This study contributes valuable insights into the effectiveness of diverse models for heart disease prediction and highlights the applicability of SHAP and LIME methods for comprehending and interpreting these models. Future research can explore the potential of incorporating these techniques into clinical practice to aid in informed decision making and improve patient outcomes.

# References

MKN, “Undang-Undang Dasar Negara Republik Indonesia Tahun 1945,” vol. 105, no. 3, pp. 129–133, 1945, [Online]. Available:

<https://webcache.googleusercontent.com/search?q=cache:BDsuQOHoCi4J:https://media.neliti.com/media/publications/9138-IDperlindungan-hukum-terhadap-anak-dari-konten-berbahaya-dalam-media-cetak-dan-ele.pdf+&cd=3&hl=id&ct=clnk&gl=id>.

[2] M. M. Ahsan, S. A. Luna, and Z. Siddique, “Machine-Learning-Based Disease Diagnosis : A,” *Healthcare*, pp. 1–30, 2022.

[3] N. S. Chandra Reddy, S. Shue Nee, L. Zhi Min, and C. Xin Ying, “Classification and Feature Selection Approaches by Machine Learning

Techniques: Heart Disease Prediction,” *Int. J. Innov. Comput.*, vol. 9, no. 1, pp. 39–46, 2019, doi: 10.11113/ijic.v9n1.210.

[4] R. H. Scheuermann, W. Ceusters, and B. Smith, “Toward an Ontological Treatment of Disease and Diagnosis Department of Pathology and

Division of Biomedical Informatics , University of Texas,” *AMIA Summit Transl. Bioinforma.*, pp. 116–120, 2009.

[5] X. Y. Gao, A. Amin Ali, H. Shaban Hassan, and E. M. Anwar, “Improving the Accuracy for Analyzing Heart Diseases Prediction Based on

the Ensemble Method,” *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/6663455.

[6] A. Rairikar, V. Kulkarni, V. Sabale, H. Kale, and A. Lamgunde, “Heart disease prediction using data mining techniques,” *Proc. 2017 Int.*

*Conf. Intell. Comput. Control. I2C2 2017*, vol. 2018-Janua, no. October, pp. 1–8, 2018, doi: 10.1109/I2C2.2017.8321771.