Enhancing Transparency and Trust in Cardiovascular Disease Predictions: A Deep Analysis of Explainable AI

Submitted in partial fulfilment of the requirements for the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Bhargav Chopra – 20BCI0151

Srishti Sinha – 20BDS0329

Under the guidance of

Dr Govinda K

SCOPE

VIT, Vellore

VIT

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

May, 2024

DECLARATION

I hereby declare that the project entitled **"Enhancing Transparency and Trust in Cardiovascular Disease Predictions: A Deep Analysis of Explainable AI"** submitted by me, for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** to VIT is a record of bonafide work carried out by me under the supervision of Dr Govinda K.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Vellore

Date: May 8, 2024

Signature of the Candidate

CERTIFICATE

This is to certify that the project entitled **"Enhancing Transparency and Trust in Cardiovascular Disease Predictions: A Deep Analysis of Explainable AI"** submitted by **Bhargav Chopra – 20BCI0151 and Srishti Sinha – 20BDS0329, SCOPE**, VIT, for the award of the degree of **Bachelor of Technology in Computer Science and Engineering,** is a record of bonafide work carried out by him under my supervision during the period, January 2024 to May 2024, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place: Vellore

Date: May 8, 2024

Signature of the Guide

8/5/24

Internal Examiner

External Examiner

Head of the Department

ACKNOWLEDGEMENTS

I extend my deepest gratitude to Dr Govinda K, Professor Grade 1, School of Computer Science and Engineering, VIT, Vellore, for his invaluable guidance, support and expertise throughout the development of this capstone project. His insightful feedback and encouragement have been instrumental in shaping this work.

I owe my profound gratitude and heartfelt appreciation to my dedicated teammate, whose collaboration and assistance have been indispensable in navigating the complexities of this project.

Furthermore, I am thankful to all individuals at Vellore Institute of Technology, Vellore involved in the successful completion of this capstone project.

Srishti Sinha

Bhargav Chopra

EXECUTIVE SUMMARY

The project focuses on enhancing cardiovascular disease (CVD) prediction through the integration of Explainable AI (XAI) techniques. CVDs are a leading cause of global mortality, necessitating early intervention and treatment. While ML models like XGBoost show promise in CVD prediction, their opaque decision-making processes hinder clinical adoption. XAI methods such as Explainable Boosting Machines (EBMs) and SHAP provide transparency into model predictions.

The project aims to improve model interpretability and robustness using a dataset of around 70,000 records. Various ML models including Random Forest, LGBM, XGBoost, Logistic Regression, CatBoost, ANN and Decision Tree Classifier will be trained, and SHAP will be employed to unravel their decision-making processes. The objectives include offering insights to healthcare practitioners for selecting interpretable models and fostering confidence in AI-powered cardiovascular risk assessments. Ultimately, the project seeks to empower healthcare professionals with transparent predictive models for informed decision-making and better patient outcomes in addressing CVDs.

List of Figures

List of Tables

List of Abbreviations

CVD	Cardiovascular diseases
XAI	Explainable AI
EDA	Exploratory Data Analysis
SHAP	SHapley Additive exPlanations
MAPLE	Model Agnostic mapping and Linking of Explanations
XGB Classifier	eXtreme Gradient Boosting Classifier
LGBM	Light Gradient Boosting Machine
EBMs	Explainable Boosting Machines
QML	Quantum Machine Learning

1. INTRODUCTION

1.1 Objectives

Our study aims to improve the interpretability and robustness of cardiovascular disease prediction models through explainable AI (XAI) techniques. Using a dataset of approximately 70,000 records, we'll train various machine learning models, including Random Forest, LGBM, XGBoost, Logistic Regression, ANN, CatBoost and Decision Tree Classifier, for cardiovascular risk prediction.

By employing SHAP (SHapley Additive exPlanations) to unravel model decision-making processes, we aim to offer insights aiding healthcare practitioners in selecting the most suitable, interpretable model. Ultimately, this research seeks to pave the way for transparent and accessible AI-powered cardiovascular risk assessments, fostering confidence within the healthcare industry.

1.2 Motivation

This project is driven by the urgent global challenge of cardiovascular diseases (CVDs), which account for a significant portion of global fatalities. Despite medical advancements, accurately predicting and diagnosing CVDs remains complex due to various clinical variables. Traditional methods lack precision, prompting the exploration of Artificial Intelligence (AI) and Machine Learning (ML) techniques like XGBoost. However, the opaque nature of these models hinders their adoption in clinical practice.

Explainable AI (XAI) techniques, such as Explainable Boosting Machines (EBMs) and Shapley Additive Explanations (SHAP), address this limitation by offering interpretable explanations for model predictions.

In essence, this project aims to enhance CVD prediction accuracy, transparency, and accessibility by integrating AI and XAI techniques. The goal is to empower healthcare professionals with interpretable and trustworthy predictive models, ultimately improving patient outcomes and saving lives.

1.3 Background

The project is grounded in the ongoing global challenge of cardiovascular diseases (CVDs), which remain a leading cause of death worldwide. Traditional methods for predicting CVD risk lack precision, prompting interest in leveraging Artificial Intelligence (AI) and Machine Learning (ML) techniques like XGBoost and LGBM.

However, the adoption of these advanced ML techniques in clinical practice is hindered by their opaque decision-making processes, leading to skepticism among healthcare professionals. Explainable AI (XAI) techniques such as Explainable Boosting Machines (EBMs) and Shapley Additive Explanations (SHAP) offer interpretable insights into model predictions, addressing this limitation.

In summary, the project aims to improve the accuracy and transparency of CVD prediction models by integrating AI and XAI techniques. By addressing the limitations of traditional methods and enhancing model interpretability, it seeks to empower healthcare professionals with more effective tools for intervention and treatment of cardiovascular diseases.

2. PROJECT DESCRIPTION AND GOALS

2.1 Survey on Existing System

Worldwide, people are impacted by the complex problem of healthcare. Inadequate medical treatment raises the burden of disease and medical expenses, which is a global concern. One of the most prevalent ailments worldwide these days is cardiovascular diseases or CVDs and about one-third of all fatalities globally are attributed to it. One of the biggest challenges that precision medicine has to face is understanding the conditionally dependent clinical variables that influence cardiovascular health outcomes. Numerous risk factors, such as high blood pressure, high cholesterol, smoking, diabetes, and obesity, contribute to CVDs. Lowering related risks and averting effects need early intervention and treatment of CVDs. Hence, its diagnosis and prediction are very essential which is easier through the medium of AI and ML. Various ML models have been built to address this issue and identify cardiovascular disorders from heart strokes and failures to cardiac arrhythmias. Several algorithms, including boosting, decision trees, random forests, convolutional neural networks (CNNs), support vector machines (SVMs), and others, have been often used for this purpose. SVM, boosting algorithms, and CNN models have demonstrated some of the best outcomes in heart stroke diagnosis. Boosting algorithms and custom-built algorithms have been shown to be the most effective for coronary artery disease prediction. In certain research, in order to forecast the risk of CVD in adults, thirty-six teenage predictors are combined with four machine learning classifiers: decision tree, random forest (RF), extreme gradient boosting (XGBoost), and deep neural networks (DNN). The AUC-ROC and AUC-PR curves on the testing data indicated that XGBoost outperformed the other machine learning models in terms of prediction ability. Building predictive models for CVDs using ML and DL techniques has been the aim of various research initiatives. Nevertheless, prior studies have often failed to close the knowledge gap between complex machine learning models and their practicality in clinical situations, thus physicians are hesitant to use them while making critical decisions. An informative deep learning model provides some intuition to support how it arrived at a given result. But in order to translate abstract representations learned by a deep learning network into a language that a healthcare professional can understand, translation of these representations into practical intuitions is necessary. Hence, to have a substantial impact on the healthcare industry by providing a trustworthy and intelligible tool for the prediction of CVD, an efficient and clearly interpretable approach for the disease's prediction, enabled by explainable AI approaches, is needed. A particular study suggests that a greater risk of heart attack or sudden cardiac death could be indicated by AP. An innovative artificial intelligence (AI) technique called Explainable Boosting Machines (EBMs) combines the interpretability of linear models with the adaptability and precision of gradient boosting. EBMs are tree-based algorithms. Not just one, but a number of explainable AI techniques have emerged, aiding in the improved interpretation of our machine learning models. Pedro A Moreno-Sánchez addresses two separate heart failure prediction models in one of his works. The most balanced explainable prediction models of these are the Survival Gradient Boosting model for survival analysis and Random Forest for classification method which use many post hoc techniques for the model's explainability study. The key intake from this research was that of the external validation and head-to-head comparisons of the most promising current CVD risk models in this era of large and combined datasets. We must understand how to customize these models for local settings, investigate whether they can be expanded with new predictors, and finally quantify the clinical impact of the most promising models as opposed to creating such novel models for predicting CVD risk. A patient-specific XAI method is layer-wise relevance propagation (LRP), which is specific to neural networks. LRP distributes relevance across layers using simple back-propagation techniques. Thus, the approach may disentangle otherwise very complex and non-linear networks layer by layer. Specifically, the Deep Taylor Decomposition (DTD) framework propagates significance from the output layer to the input layer by applying a Taylor decomposition at each layer. DTD can be thought of as a theoretical framework incorporating many back-propagation relevance approaches, including LRP. While DTD and LRP can offer feature relevance at the patient-specific level, their applicability is limited to certain models, and there is currently no comprehensive implementation that supports all neural network topologies. It was discovered that the healthcare sector uses both SHAP (SHapley Additive exPlanations) and MAPLE (Model Agnostic mapping and Linking of Explanations) to evaluate the interpretability and spot bias in machine learning models. A comparative investigation reveals no discernible distinction between SHAP and MAPLE for this specific application in bias detection. The most popular models mentioned in many of the research papers was Shapley Additive Explanations (SHAP). It is patient-specific as well as model-agnostic. The idea of Shapley values and cooperative game theory serve as the foundation for SHAP values. SHAP values indicate how each feature contributes to the variation between the model's prediction and the average prediction in the context of machine learning. With respect to healthcare, the likelihood of contracting an illness or experiencing an incident can then be compared in the presence and absence of the risk factor in order to determine the contribution of each risk factor. Because this difference can be calculated for each patient individually, SHAP is a compelling option for clinical decision support systems and is very popular among tree models, such as XGBoost. When it comes to predicting CVDs, the model may be trained using a dataset that includes patient data like sex, age, smoking status, cholesterol, blood pressure, and other pertinent characteristics, along with labels designating whether or not the condition is present. After the model is trained, SHAP values can be found for each prediction. This can be achieved by creating a set of permutations for each feature and evaluating the model's prediction for each permutation. The way SHAP operates is by averaging the contributions of each and every combination of the attributes. This enables the identification of the critical traits—such as systolic blood pressure—that are essential for the prediction of CVDs. The model's predictions can be understood both locally and globally through SHAP Python module, which is widely used to calculate SHAP values and generate charts. The method has been applied to filter features, interpreting certain feature quality in relation to CVD, and calculate the risk of dying from all causes. When it comes to deciphering and comprehending machine learning models—including those that forecast cardiovascular illnesses—SHAP offers a useful resource. Better clinical decisions are made possible by its transparency and insights into the model's decision-making process. In addition to this, the results of certain studies also show that quantum classifiers outperform classical classifiers in terms of effectiveness. The results demonstrated that quantum classifier performance can be improved by ensemble learning models. Hence, it is believed that Quantum ML/AI along with the XAI techniques is a very advanced approach to tackle this issue of CVD prediction.

2.2 Gaps Identified

- **Limited XAI Studies in Healthcare:**
  - Scarcity of studies applying XAI techniques to cardiovascular disease prediction.
- **Integration Gap in Clinical Practice:**
  - Challenges in translating XAI insights for healthcare practitioners.
- **Lack of Standardized Evaluation Metrics:**
  - Absence of standardized metrics for assessing model interpretability.
- **Healthcare Professionals' Understanding:**
  - Limited familiarity among healthcare professionals with XAI techniques.
- **Exploration Gap in Advanced AI Techniques:**
  - Insufficient exploration of advanced AI techniques like Quantum Machine Learning.
- **Need for Real-world Validation:**
  - Lack of real-world validation of XAI-enhanced models in clinical settings.

2.3 Problem Statement

The opacity of machine learning algorithms in cardiovascular risk prediction poses a barrier to their adoption in clinical settings. Without transparent and comprehensible models, healthcare professionals and patients may lack confidence in their accuracy. This study aims to address this challenge by enhancing the interpretability of cardiovascular disease prediction models through the application of explainable AI (XAI) techniques. By improving model transparency, we seek to instill trust and facilitate the integration of AI-powered risk assessments into healthcare practices.

3. TECHNICAL SPECIFICATION

3.1 Requirements

3.1.1 Functional

- **Data Handling:**
  - Gather and preprocess a large dataset for cardiovascular disease prediction.
- **Model Training and Evaluation:**
  - Train various machine learning models (e.g., Random Forest, SVM, XGBoost) and evaluate their performance.
- **Explainable AI Integration:**
  - Integrate SHAP and other XAI techniques to interpret model predictions and feature importance.
- **Comparative Analysis:**
  - Conduct comparative analysis of model accuracy and interpretability.
- **Insights Generation:**
  - Generate insights from XAI interpretations for healthcare practitioners.
- **User Interface:**
  - Provide an intuitive interface for data input, model training, and result visualization.
- **Scalability and Performance:**
  - Ensure scalability to handle large datasets and efficient computation for timely results.
- **Documentation and Reporting:**
  - Generate comprehensive reports documenting methodology, results, and recommendations.
- **Integration with Healthcare Systems:**
  - Integrate with existing healthcare systems while ensuring data privacy and security compliance.

3.1.2 Non-Functional

- **Performance and Scalability:**
  - Ensure the system responds promptly and can handle increasing data volumes and user load efficiently.
- **Reliability:**
  - Ensure continuous availability of the system without unexpected downtime.
- **Usability:**
  - Design an intuitive interface for healthcare professionals with minimal training requirements.
- **Interpretability:**
  - Provide clear explanations for model predictions and feature importance to facilitate understanding by users.
- **Security and Privacy:**
  - Implement robust security measures to protect patient data and comply with privacy regulations.
- **Compatibility:**
  - Ensure compatibility with various operating systems and browsers for user convenience.
- **Maintainability:**
  - Design the system with modular components and well-documented code to facilitate maintenance and updates.

3.2 Feasibility Study

3.2.1 Technical Feasibility

The proposed study’s technical feasibility relies on key factors:

- **Data Availability:** Access to a diverse dataset with relevant parameters is essential for training accurate machine learning models. Availability of such data is critical for meaningful predictions.
- **Machine Learning Frameworks:** Utilizing robust machine learning frameworks like TensorFlow and scikit-learn is crucial. These frameworks provide necessary functionalities for model training, evaluation, and interpretation.
- **Explainable AI Techniques:** Integration of Explainable AI (XAI) techniques, such as SHAP, enhances model interpretability. Availability of libraries supporting XAI is necessary for transparent explanations.
- **Computational Resources:** Adequate computational resources, including processing power and memory, are needed for training models with large datasets.
- **Scalability:** Designing the system to scale efficiently is essential to accommodate growing data volumes and user load over time without sacrificing performance.
- **Security Measures:** Robust security measures, including encryption and access controls, are crucial to protect patient privacy and comply with data protection regulations.
- **Compatibility and Integration:** Ensuring compatibility with existing healthcare IT infrastructure and seamless integration with EHR systems are vital for successful deployment in clinical settings.

3.2.2 Economic Feasibility

The economic feasibility of implementing the proposed system for cardiovascular disease prediction is contingent upon several factors:

- **Cost of Data Acquisition:** Acquiring a comprehensive dataset for model training may incur costs, including purchasing data from external sources or investing resources in data collection efforts.
- **Development Costs:** The development of the system, including software engineering, algorithm development, and integration of XAI techniques, requires investment in skilled personnel and technology infrastructure.
- **Computational Resources:** The costs associated with computational resources for model training and evaluation, such as cloud computing services or dedicated hardware, should be considered.
- **Maintenance and Support:** Ongoing maintenance, updates, and technical support for the system incur recurring costs. This includes expenses related to bug fixes, software upgrades, and user support.
- **Training and Education:** Training healthcare professionals on how to use the system effectively may require investment in educational resources and training programs.
- **Return on Investment (ROI):** Assessing the potential return on investment, including potential cost savings from improved disease prediction accuracy, reduced healthcare expenses, and improved patient outcomes, is crucial for determining the economic viability of the system.
- **Long-term Sustainability:** Evaluating the long-term sustainability of the system, including its ability to adapt to evolving healthcare needs and technological advancements, is essential for ensuring continued economic feasibility.

3.2.3 Social Feasibility

The social feasibility of implementing the proposed system for cardiovascular disease prediction relies on several factors:

- **User Acceptance:** Assessing the acceptance of the system by healthcare professionals and patients is crucial. User feedback and engagement are essential for successful adoption and utilization.
- **Ethical Considerations:** Ensuring that the system upholds ethical principles, including fairness, transparency, and patient privacy, is paramount. Adhering to ethical standards fosters trust and confidence among users.
- **Accessibility:** Ensuring the system is accessible to diverse populations, including underserved communities and individuals with varying levels of digital literacy, promotes equitable healthcare access and inclusivity.
- **Impact on Healthcare Practices:** Evaluating the potential impact of the system on healthcare practices, including its ability to improve disease prevention, diagnosis, and treatment, is essential. Positive outcomes contribute to social acceptance and support.
- **Cultural Considerations:** Considering cultural factors and preferences when designing and implementing the system ensures relevance and acceptance within different cultural contexts.

3.3 System Specification

3.3.1 Hardware Specification

- Processor:
    - Intel Core i5 or AMD Ryzen 5 series processor or better.
- Memory (RAM):
    - 8 GB or higher for optimal performance.
- Storage:
    - SSD recommended.
- Networking:
    - Reliable Ethernet or WiFi connection.

3.3.2 Software Specification

- Operating System:
    - Compatible with Windows, macOS, or Linux.
- Programming Language:
    - Python 3
- Machine Learning Frameworks:
    - TensorFlow
    - scikit-learn
    - XGBoost
    - lightgbm
    - catboost
    - keras
- Explainable AI (XAI) Libraries:
    - SHAP (SHapley Additive exPlanations)
- Development Environment:
    - IDE such as VS Code or Jupyter Notebook for Python programming.
- Visualization Tools:
    - Matplotlib and Seaborn for data visualization.
    - Tableau for interactive visualizations.
- Documentation:
    - LaTeX or Microsoft Word.
- Deployment:
    - Cloud platforms such as AWS, Google Cloud, or Microsoft Azure for deployment.

3.3.3 Standards and Policies

- Data Privacy and Security:
    - Adhere to GDPR and HIPAA regulations for protecting patient data and ensuring confidentiality.
- Ethical Guidelines:
    - Follow ethical guidelines for research involving human subjects, ensuring informed consent and fairness.
- Model Transparency:
    - Implement standards for model transparency to enhance trust among users.
- Interoperability:
    - Ensure interoperability with existing healthcare systems for seamless data exchange.
- Documentation:
    - Maintain thorough documentation for transparency and reproducibility.
- Quality Assurance:
    - Implement quality assurance measures to ensure system reliability and accuracy.
- Regulatory Compliance:
    - Stay updated on relevant regulations and ensure compliance with industry standards.

4. DESIGN APPROACH AND DETAILS

4.1 System Architecture

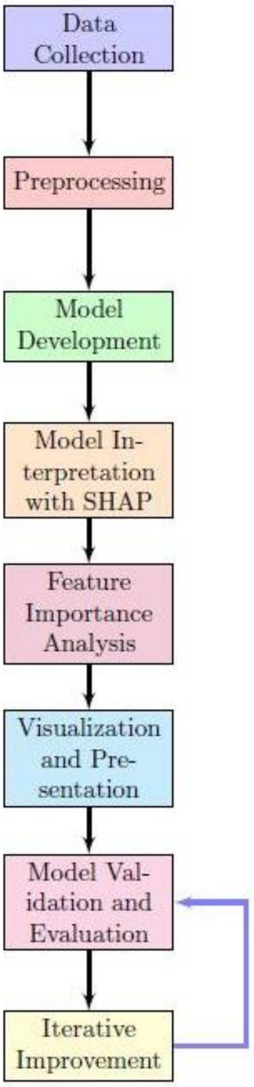

Data Collection
Preprocessing
Model Development
Model Interpretation with SHAP
Feature Importance Analysis
Visualization and Presentation
Model Validation and Evaluation
Iterative Improvement

Figure 4.1: Project Architecture

- **Data Collection and Preprocessing:**
    - ○ Compiling information about cardiovascular diseases, considering variables like age, blood pressure, cholesterol, smoking, diabetes, etc.
    - ○ Handling missing values, outliers, and ensuring data quality through preprocessing techniques.

    - Normalizing features, encoding categorical variables, and preparing the dataset for model training.
    - Conducting exploratory data analysis to gain insights into the distribution and relationships of variables, informing preprocessing decisions.

- **Model Training and Evaluation:**
    - Training various machine learning models such as Random Forest, LGBM, and XGBoost.
    - Evaluating models using metrics like accuracy, precision, and recall.
    - Validating the predictive model's performance and ensuring alignment with clinical expertise.
    - Assessing model robustness through cross-validation techniques to ensure generalizability to new data.

- **Explainable AI (XAI) Integration:**
    - Applying SHAP and other XAI techniques to interpret model decisions.
    - Analyzing feature importance to understand cardiovascular risk prediction.
    - Enhancing trust and confidence in the model's decision-making process by providing transparent explanations for predictions.
    - Investigating the impact of different XAI methods on model interpretability and decision support.

- **Comparative Analysis and Model Selection:**
    - Comparing models based on accuracy, interpretability, and suitability.
    - Selecting the most suitable model(s) that balance accuracy and transparency.
    - Considering computational efficiency and scalability for practical implementation in healthcare settings.

- **Insights Generation and Documentation:**

  - Documenting findings on feature importance and decision-making processes.
  - Providing recommendations for healthcare practitioners based on insights generated from the models.
  - Developing interactive visualizations to facilitate understanding and interpretation of model insights.

- **Future Directions:**
  - Discussing potential research avenues such as Quantum Machine Learning (QML) integration.
  - Delivering a concise report with methodology, results, and recommendations for further research and development.
  - Exploring opportunities for collaboration with researchers and healthcare organizations to validate and implement the predictive models in real-world clinical settings.

.

The project intends to provide transparent and reliable predictions of cardiovascular disease by adhering to this structured approach and using the SHAP model for interpretation. This will enable healthcare providers and patients to gain important insights into disease risk factors and prediction mechanisms.

4.2 Design

4.2.1 Data Flow Diagram

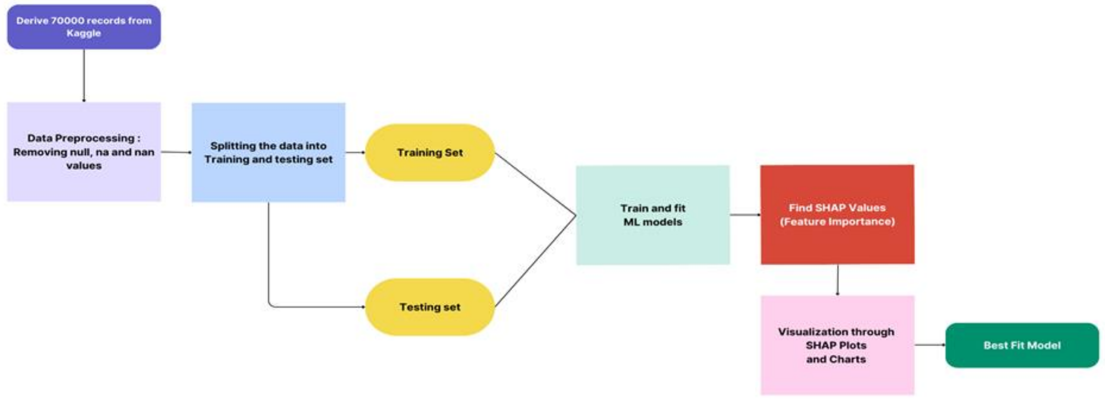

Derive 70000 records from Kaggle
Data Preprocessing : Removing null, na and nan values
Splitting the data into Training and testing set
Training Set
Testing set
Train and fit ML models
Find SHAP Values (Feature Importance)
Visualization through SHAP Plots and Charts
Best Fit Model

Figure 4.2: Data Flow Diagram

Data Collection and Preprocessing: Using the pandas package, effective data manipulation was achieved by obtaining a dataset of roughly 70,000 records. To find and manage missing values, outliers, and inconsistencies, the dataset underwent extensive preprocessing using ydata-profiling. The next step was to perform exploratory data analysis, or EDA, to learn more about the features of the dataset.

Data Splitting: Next, an 80:20 split of the preprocessed data was made into training and testing sets. Model training and optimisation were carried out using the training set, which made up 80% of the data. However, the testing set, which made up 20% of the data, was held aside to assess how well the model performed with data that had not yet been viewed.

Model Development: A variety of machine learning (ML) models, such as Random Forest, LightGBM (LGBM), XGBoost, Logistic Regression, ANN, CatBoost, and Decision Tree Classifier, were used to predict cardiovascular disease (CVD). The training dataset was used to teach each model how to find patterns and connections in the data.

Integration with SHAP: To improve interpretability, each ML model was integrated with the SHAP (SHapley Additive exPlanations) library. To ascertain the significance of a feature in the model's predictions, SHAP values were computed. This integration increased the models' transparency by illuminating the contributions of each feature to the overall forecasts.

Visualisation for Interpretability: Visualisation charts were created in order to improve the black-box models' interpretability even more. These featured SHAP summary plots, which gave a summary of feature importance, dependency plots, which illustrated how a single feature affected predictions across a range of values, and force plots, which displayed individual predictions and feature contributions.

Evaluation of the Model: A number of performance indicators, including F1-score, Accuracy, Precision, Recall, etc. were used to carefully assess the model's performance. Based on these metrics, the optimal model for the selected application was chosen, guaranteeing the model's ability to forecast the occurrence of cardiovascular illness.

4.2.2 Use Case Diagram

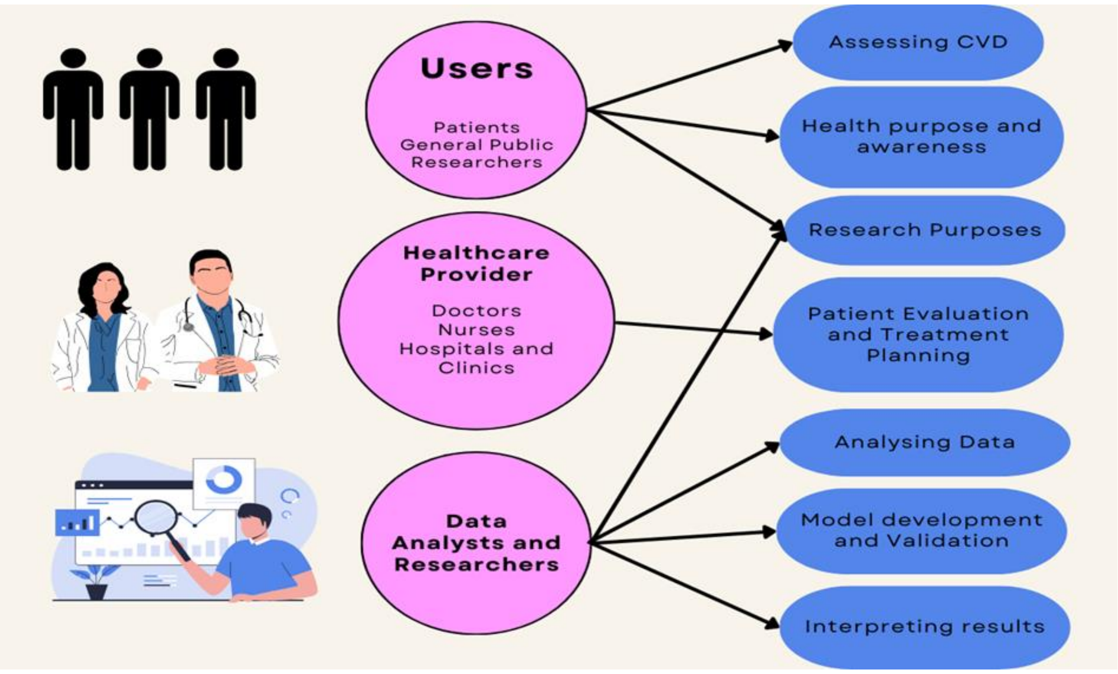


Figure 4.3: Use Case Diagram

4.2.3 Class Diagram

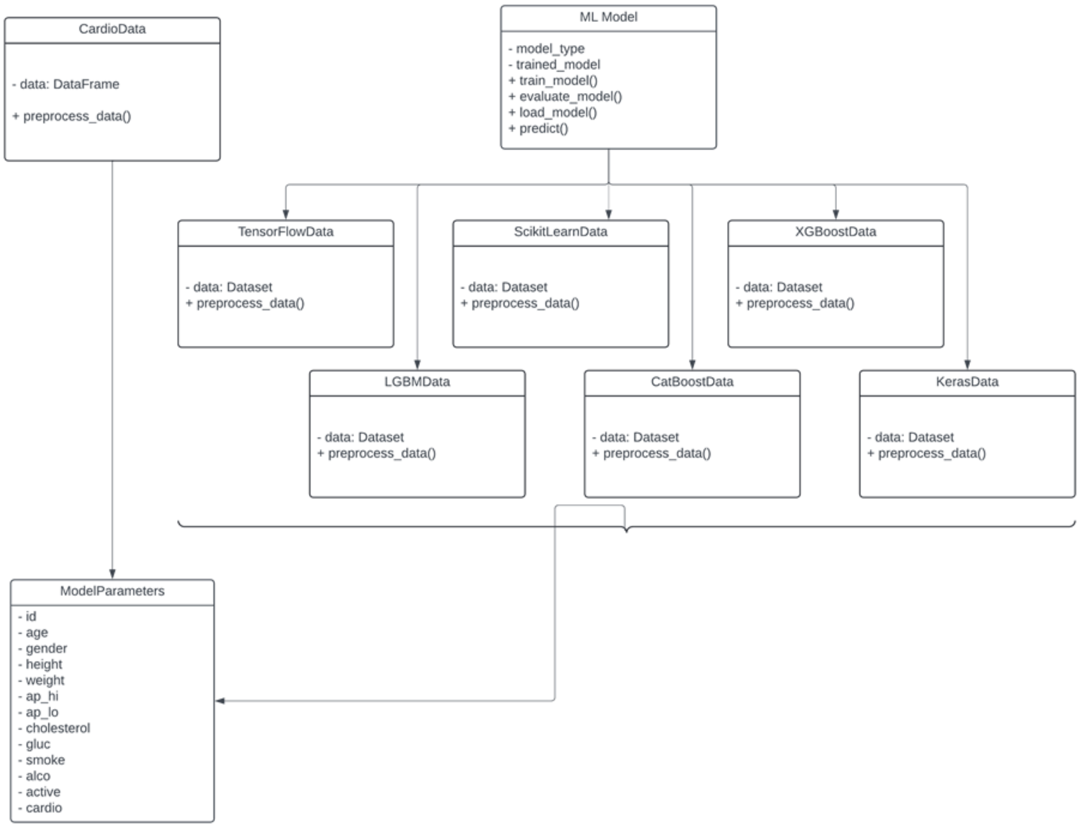


Figure 4.4: Class Diagram

4.2.4 Sequence Diagram

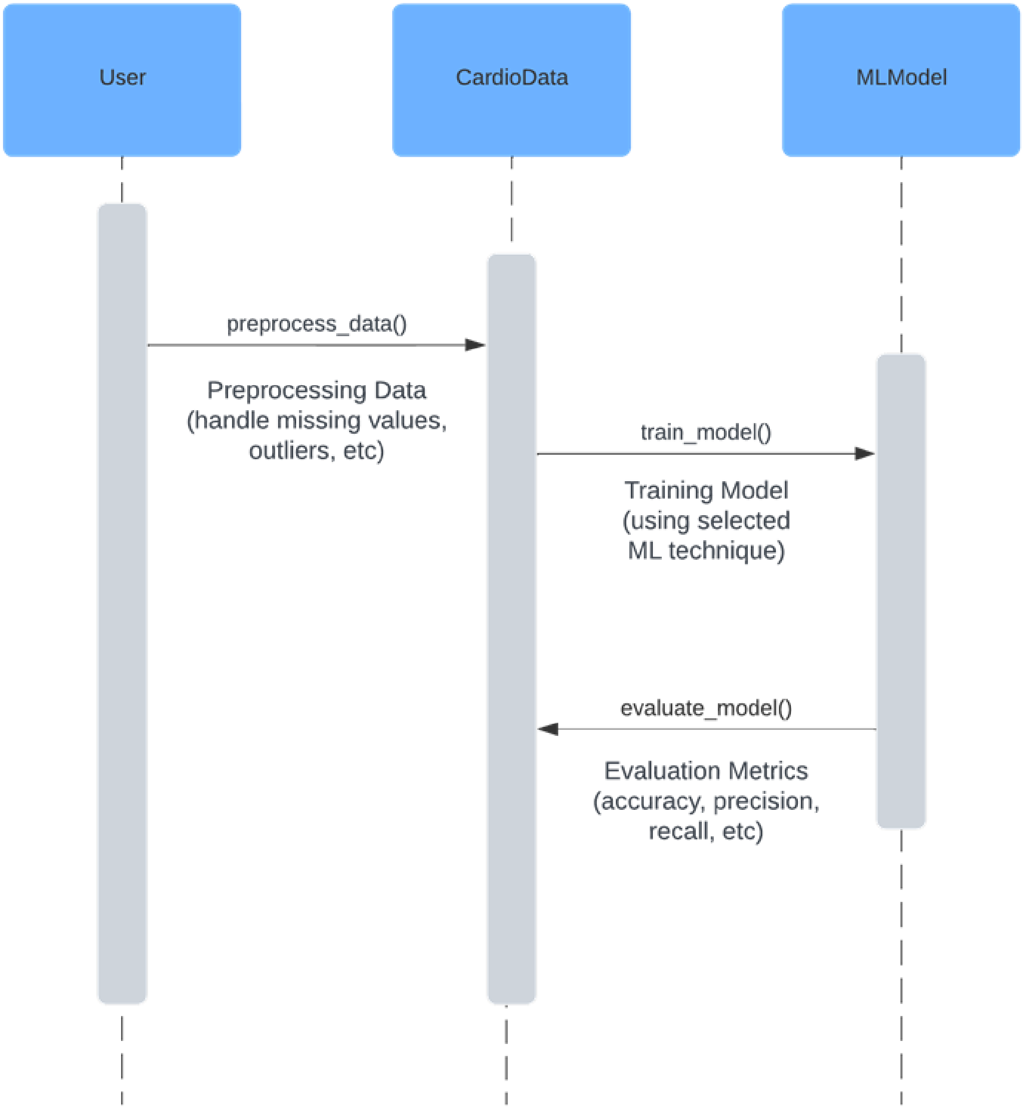

User
CardioData
MLModel
preprocess_data()
Preprocessing Data (handle missing values, outliers, etc)
train_model()
Training Model (using selected ML technique)
evaluate_model()
Evaluation Metrics (accuracy, precision, recall, etc)

Figure 4.5: Sequence Diagram

4.3 Constraints, Alternatives and Tradeoffs

- **Data Size and Quality**

  Constraint: Limited data availability may limit the diversity and amount of the dataset, which could have an impact on the performance of the model.

  Alternatives: Acquiring more data sources or enhancing data using methods like data synthesis.

  Trade-offs: While additional data sources may differ in quality and relevance, augmented data may contribute noise.

- **Computational Resources**

  Constraint: The complexity of models that may be trained and the computation speed may be affected by a lack of computational resources.

  Alternatives: Make use of distributed computing frameworks or cloud computing services.

  Trade-offs: While distributed frameworks could need more setup and maintenance, cloud services might cost more.

- **Model Complexity**

  Constraint: While increasing model complexity may boost predictive accuracy, it also increases the risk of overfitting and reduced interpretability.

  Alternatives: Investigate less complex models or group methods to strike a compromise between interpretability and complexity.

  Trade-offs: While ensemble approaches may need greater processing resources, simpler models may forfeit prediction accuracy.

- **Interpretability vs Performance**

  Constraint: Improving the interpretability of a model by methods such as SHAP may result in a decrease in its predictive performance.

Alternatives: Use model-agnostic interpretability tools or modify model complexity to strike a balance between interpretability and performance.

Trade-offs: While complex models may lack transparency, highly interpretable models may forfeit prediction accuracy.

- **Feature Engineering**

  Constraint: Inadequate domain expertise or unavailability of features could limit the efficacy of feature engineering.

  Alternatives: To create pertinent features, use automated feature engineering methods or domain experts.

  Trade-offs: Expert-driven feature engineering can be time-consuming, while automated approaches might miss domain-specific insights.

- **Model Selection**

  Constraint: Choosing the top-performing model only on the basis of assessment metrics could leave out important aspects such as interpretability and model complexity.

  Alternatives: Take into account a multi-criteria decision-making process that balances interpretability and model complexity with performance measures.

  Trade-offs: Prioritising interpretability above performance measures may result in fewer interpretable models, whereas emphasising interpretability alone may compromise predictive performance.

5. SCHEDULE, TASKS AND MILESTONES

5.1 Gantt Chart

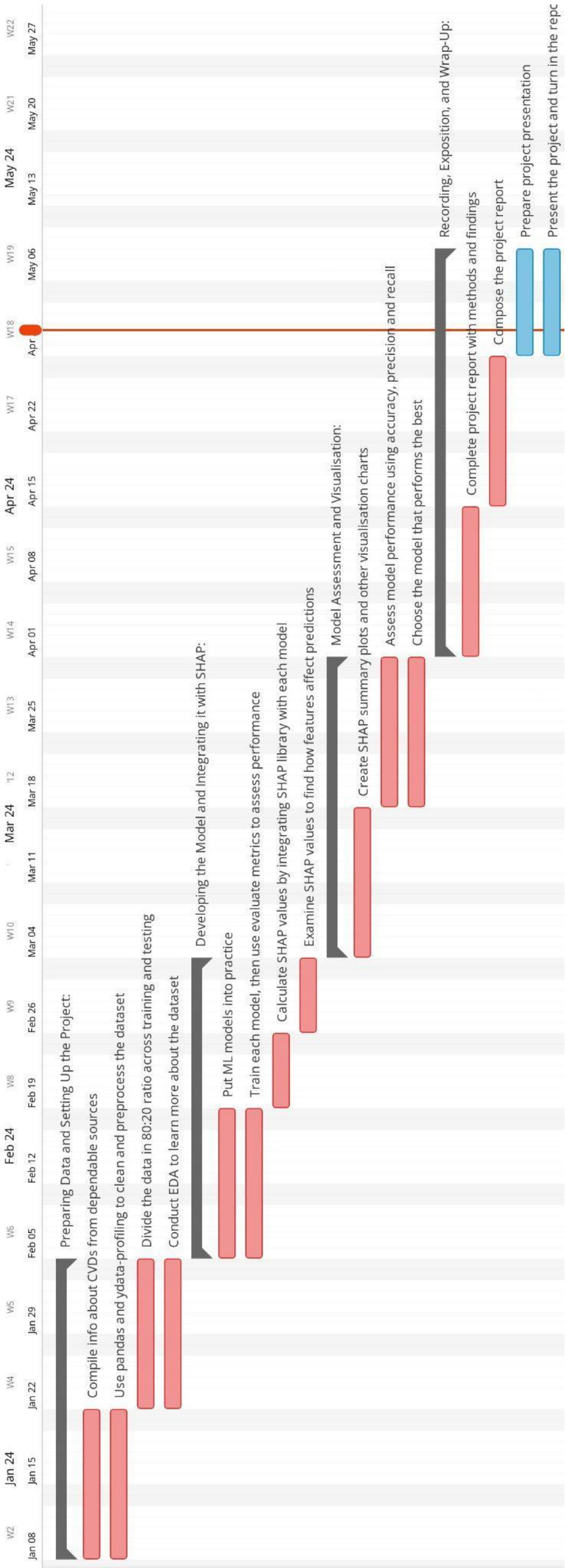


Figure 5.1: Gantt Chart

5.2 Module Description

5.2.1 Module 1 – Data Collection and Preprocessing Module:

This module is responsible for collecting relevant healthcare datasets containing electronic health records, medical imaging data, genetic information, etc. It preprocesses the data to handle missing values, outliers, and ensure data quality.

Tasks:

Data acquisition from various sources

Data cleansing and preprocessing

5.2.2 Module 2 – Exploratory Data Analysis (EDA) Module:

This module performs exploratory data analysis to gain insights into the characteristics of the data, identify patterns, and visualize relationships between variables.

Tasks:

Descriptive statistics analysis

Visualization of data distributions and correlations

Identification of relevant features for modeling

5.2.3 Module 3 – Model Development and Evaluation Module:

This module encompasses the development, training, and evaluation of machine learning models for cardiovascular disease prediction.

Tasks:

Selection of algorithms (e.g., logistic regression, random forests, neural networks)

Model training

Performance metric calculation

5.2.4 Module 4 – Explainability Analysis Module:

This module focuses on interpreting model predictions and understanding the factors influencing them using explainable AI techniques, such as SHAP.

Tasks:

Calculation of SHAP values for feature importance

Visualization of SHAP summary plots and individual explanations

Analysis of feature contributions to predictions

5.3 Testing

5.3.1 Unit Testing

Unit testing involves testing individual components or units of code to verify that they function correctly in isolation. Here's how unit testing can be applied:

Model Component Testing:
Verify the correctness of individual machine learning models (e.g., Random Forest, LGBM, XGBoost) to ensure they are trained and produce accurate predictions.

Preprocessing Function Testing:
Validate data preprocessing functions (e.g., handling missing values, normalizing features) to ensure data integrity and consistency before model training.

Explainable AI (XAI) Technique Testing:
Confirm the accuracy of Explainable AI (XAI) techniques, such as SHAP values calculation, to provide transparent explanations for model predictions and feature importance.

Data Quality Checks:
Ensure data quality by testing for outliers and inconsistencies during preprocessing, maintaining the integrity of the dataset used for model training and evaluation.

Integration Testing:
Validate end-to-end workflow, including data preprocessing, model training, and interpretation, to ensure seamless interaction between different components and the reliability of the overall system.

5.3.2 Integration Testing

End-to-End Workflow Validation:
Validate the entire workflow from data preprocessing to model training, evaluation, and interpretation to ensure seamless integration between different components.

Data Flow Verification:
Verify the flow of data between preprocessing steps, machine learning models, and Explainable AI (XAI) techniques to ensure consistency and accuracy in data processing.

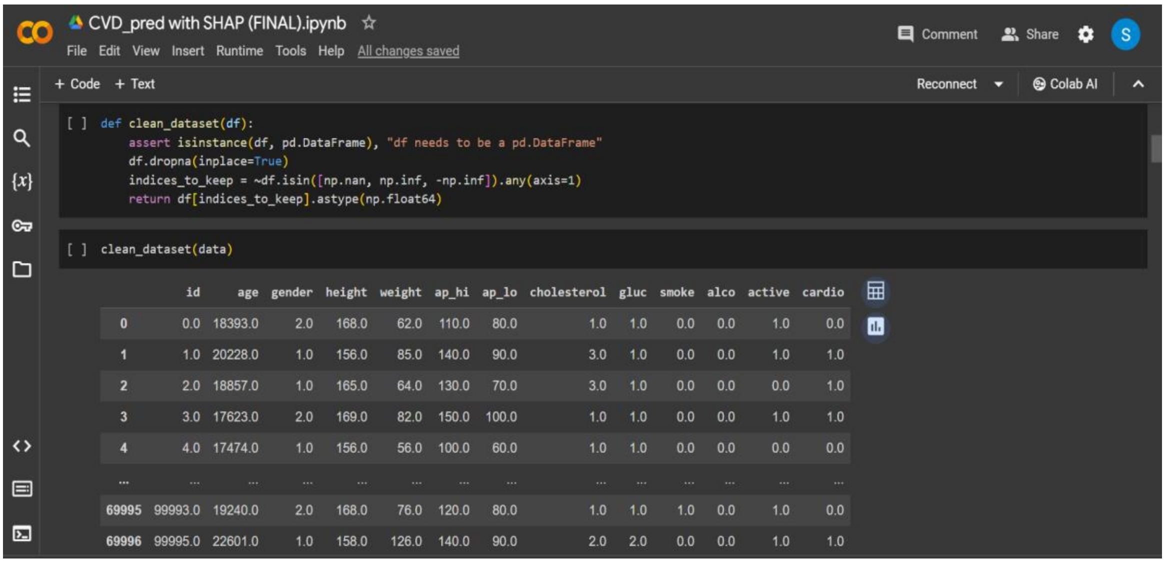
Interactions Between Components:
Test interactions and dependencies between different components, such as preprocessing functions and machine learning algorithms, to ensure they work together as intended.

Boundary Conditions Testing:
Evaluate system behavior under various scenarios, including edge cases and boundary conditions, to ensure robustness and reliability in different situations.

Error Handling and Recovery:
Test error handling mechanisms and recovery procedures to ensure the system can gracefully handle exceptions and unexpected events, maintaining stability and integrity throughout the workflow.

6. PROJECT DEMONSTRATION

Cleaning the dataset: Removed all the na and nan values from the dataset.


CVD_pred with SHAP (FINAL).ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
Reconnect
Colab AI

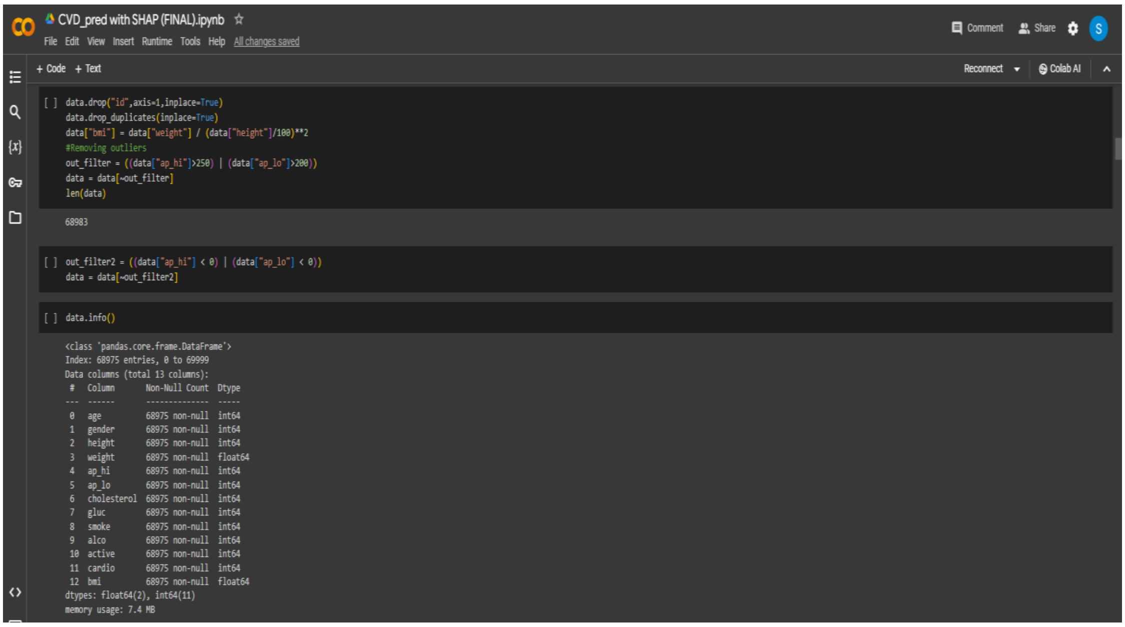
```
[ ] def clean_dataset(df):
        assert isinstance(df, pd.DataFrame), "df needs to be a pd.DataFrame"
        df.dropna(inplace=True)
        indices_to_keep = ~df.isin([np.nan, np.inf, -np.inf]).any(axis=1)
        return df[indices_to_keep].astype(np.float64)
```

```
[ ] clean_dataset(data)
```

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
0	0.0	18393.0	2.0	168.0	62.0	110.0	80.0	1.0	1.0	0.0	0.0	1.0	0.0
1	1.0	20228.0	1.0	156.0	85.0	140.0	90.0	3.0	1.0	0.0	0.0	1.0	1.0
2	2.0	18857.0	1.0	165.0	64.0	130.0	70.0	3.0	1.0	0.0	0.0	0.0	1.0
3	3.0	17623.0	2.0	169.0	82.0	150.0	100.0	1.0	1.0	0.0	0.0	1.0	1.0
4	4.0	17474.0	1.0	156.0	56.0	100.0	60.0	1.0	1.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
69995	99993.0	19240.0	2.0	168.0	76.0	120.0	80.0	1.0	1.0	1.0	0.0	1.0	0.0
69996	99995.0	22601.0	1.0	158.0	126.0	140.0	90.0	2.0	2.0	0.0	0.0	1.0	1.0

Figure 6.1: Data Preprocessing (1)

Removing outliers: Removed all the extreme values for numerical attributes such as bmi, ap_hi, ap_lo to reduce the noise and bias during our analysis.


CVD_pred with SHAP (FINAL).ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
Reconnect
Colab AI

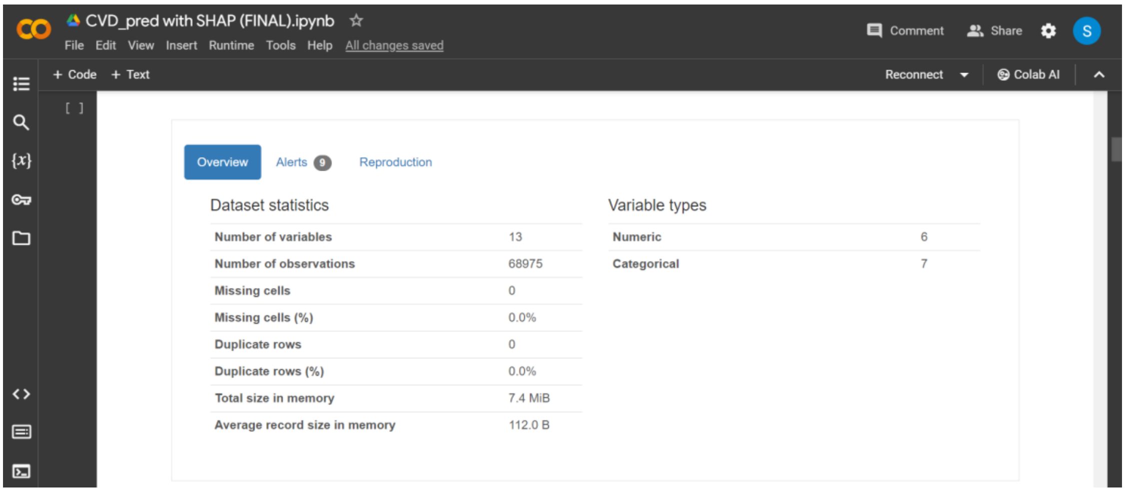
```
[ ] data.drop("id",axis=1,inplace=True)
    data.drop_duplicates(inplace=True)
    data["bmi"] = data["weight"] / (data["height"]/100)**2
    #Removing outliers
    out_filter = ((data["ap_hi"]>250) | (data["ap_lo"]>200))
    data = data[~out_filter]
    len(data)

    68983
```

```
[ ] out_filter2 = ((data["ap_hi"] < 0) | (data["ap_lo"] < 0))
    data = data[~out_filter2]
```

```
[ ] data.info()

    <class 'pandas.core.frame.DataFrame'>
    Index: 68975 entries, 0 to 69999
    Data columns (total 13 columns):
     #   Column       Non-Null Count  Dtype  
    ---  ------       --------------  -----  
     0   age          68975 non-null  int64  
     1   gender       68975 non-null  int64  
     2   height       68975 non-null  int64  
     3   weight       68975 non-null  float64
     4   ap_hi        68975 non-null  int64  
     5   ap_lo        68975 non-null  int64  
     6   cholesterol  68975 non-null  int64  
     7   gluc         68975 non-null  int64  
     8   smoke        68975 non-null  int64  
     9   alco         68975 non-null  int64  
     10  active       68975 non-null  int64  
     11  cardio       68975 non-null  int64  
     12  bmi          68975 non-null  float64
    dtypes: float64(2), int64(11)
    memory usage: 7.4 MB
```

Figure 6.2: Data Preprocessing (2)

Exploratory Data Analysis: By using the ydata_profiling module, we derived a detailed report of the dataset as well as each of its attributes. Various visualization charts such as heatmaps and bar plots gave us insights about correlation factors, etc.


CVD_pred with SHAP (FINAL).ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
S
+ Code + Text
Reconnect
Colab AI
[ ]
Overview
Alerts 9
Reproduction

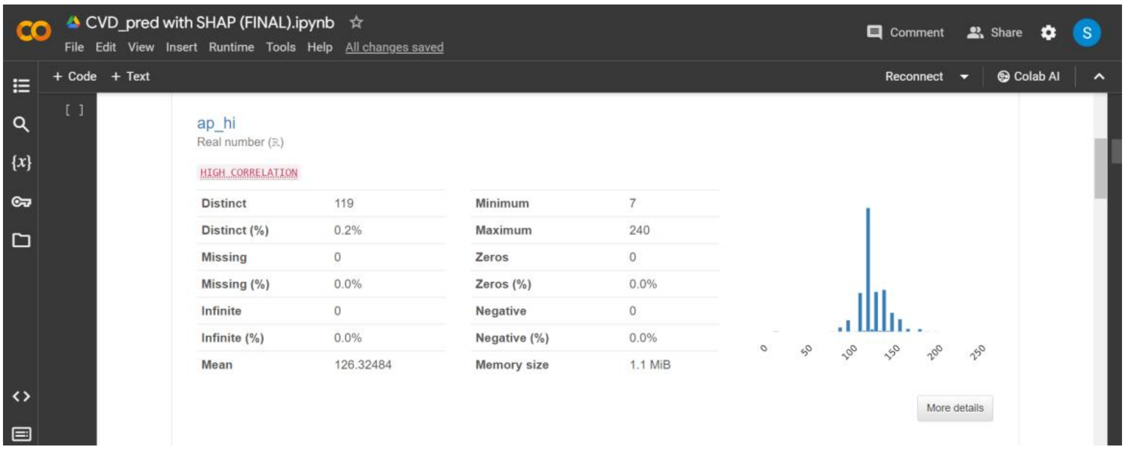
Dataset statistics

Number of variables	13
Number of observations	68975
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	7.4 MiB
Average record size in memory	112.0 B

Variable types

Numeric	6
Categorical	7

Figure 6.3: Dataset Statistics


CVD_pred with SHAP (FINAL).ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
S
+ Code + Text
Reconnect
Colab AI
[ ]
0
50
100
150
200
250
More details

ap_hi

Real number (ℝ)

HIGH CORRELATION

Distinct	119	Minimum	7
Distinct (%)	0.2%	Maximum	240
Missing	0	Zeros	0
Missing (%)	0.0%	Zeros (%)	0.0%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%
Mean	126.32484	Memory size	1.1 MiB

Figure 6.4: Attribute Statistics

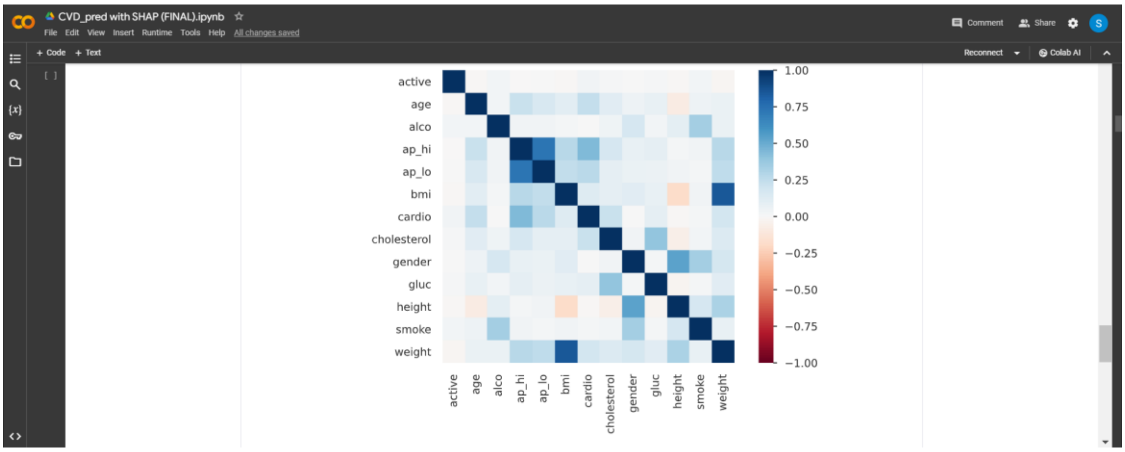
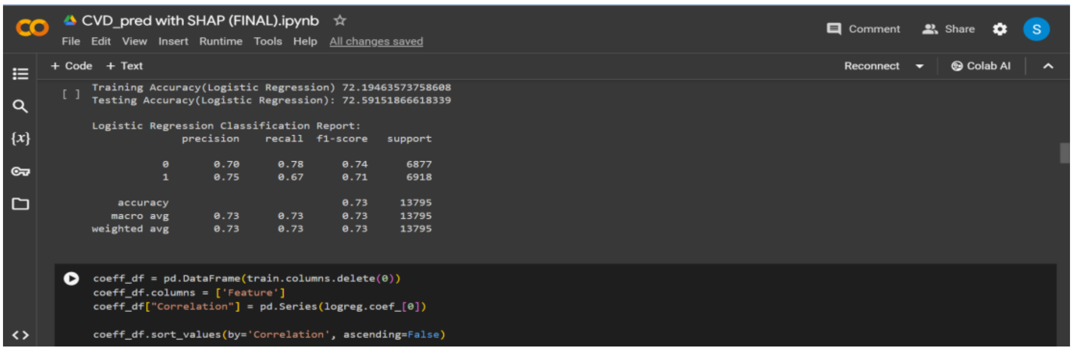


Figure 6.5: Heatmap for all the Attributes in the Dataset

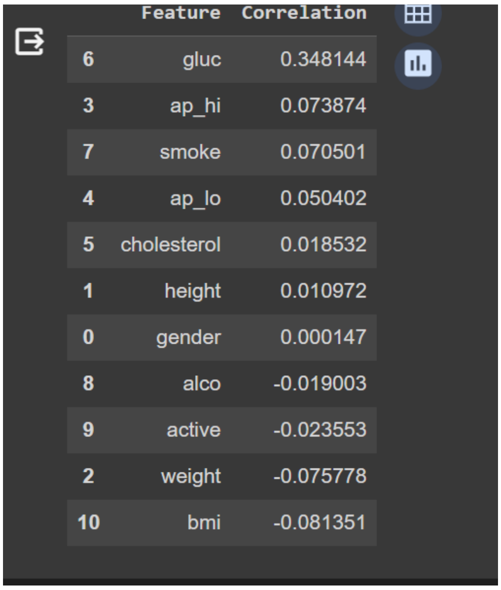
Logistic Regression: We chose logistic regression as the base algorithm which gave us an accuracy of about 72.6% after which we found the correlations among all the features with respect to logistic regression which was highest for glucose intake.


CVD_pred with SHAP (FINAL).ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
Reconnect
Colab AI

```
Training Accuracy(Logistic Regression) 72.19463573758608
Testing Accuracy(Logistic Regression): 72.59151866618339

Logistic Regression Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.78      0.74      6877
           1       0.75      0.67      0.71      6918

    accuracy                           0.73     13795
   macro avg       0.73      0.73      0.73     13795
weighted avg       0.73      0.73      0.73     13795
```

```
coeff_df = pd.DataFrame(train.columns.delete(0))
coeff_df.columns = ['Feature']
coeff_df["Correlation"] = pd.Series(logreg.coef_[0])

coeff_df.sort_values(by='Correlation', ascending=False)
```

Figure 6.6: Logistic Regression Evaluation Metrics



	Feature	Correlation
6	gluc	0.348144
3	ap_hi	0.073874
7	smoke	0.070501
4	ap_lo	0.050402
5	cholesterol	0.018532
1	height	0.010972
0	gender	0.000147
8	alco	-0.019003
9	active	-0.023553
2	weight	-0.075778
10	bmi	-0.081351

Figure 6.7: Feature Correlation for Logistic Regression

Summary plot for Logistic Regression: This plot shows that weight has the highest absolute magnitude of the SHAP values, with the longest bars appearing at the top of the plot. Hence, it is the most significant feature in a model's predictions followed by bmi and then ap_hi (systolic blood pressure).

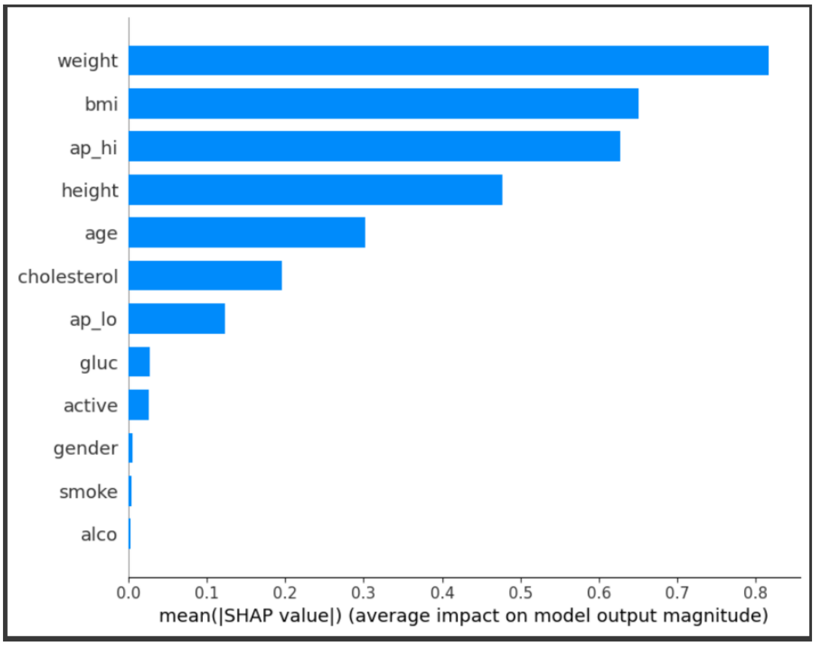


Figure 6.8: Bar Summary Plot for LR

Beeswarm Summary Plot for LR: An overview of each feature's effect on model predictions for the whole dataset is given by a summary plot. Every feature is represented by a vertical line, and the average magnitude of the SHAP value for each feature across all samples is indicated by the line's horizontal position. The line's colour, which is blue for low values and red for high values, indicates the feature's value. Here, we notice that weight has the highest average SHAP value magnitude leading it to be the most prevalent feature in model prediction. The high values of weight lead to greater impact in causing CVD and vice-versa.

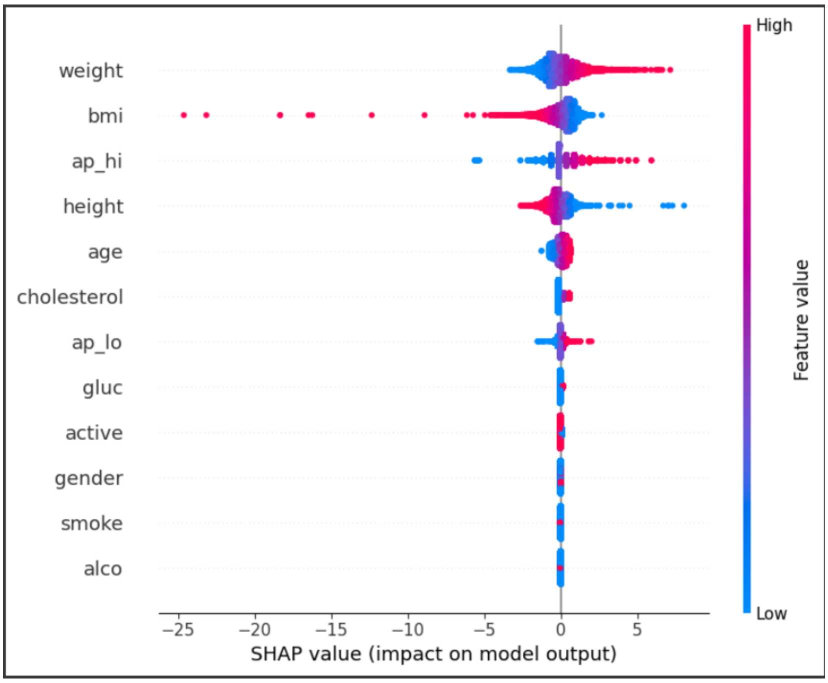
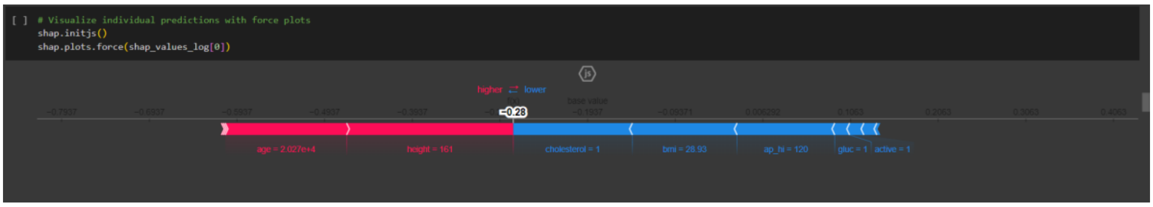


Figure 6.9: Beeswarm Summary Plot for LR

Force plots for LR: A force plot shows in great detail how each value of a feature affects a particular prediction that the model makes. Every force plot illustrates how several features contribute to the model's forecast deviating from the average or baseline prediction. Here, the predictive value is -0.28 which means that the features result in a negative outcome. The attributes in blue such as the cholesterol, bmi and blood pressure all being normal for this particular prediction, push the results to be more driven towards a person to not have CVD.



```
[ ] # Visualize individual predictions with force plots
    shap.initjs()
    shap.plots.force(shap_values_log[0])
```

Figure 6.10: Force Plot for LR

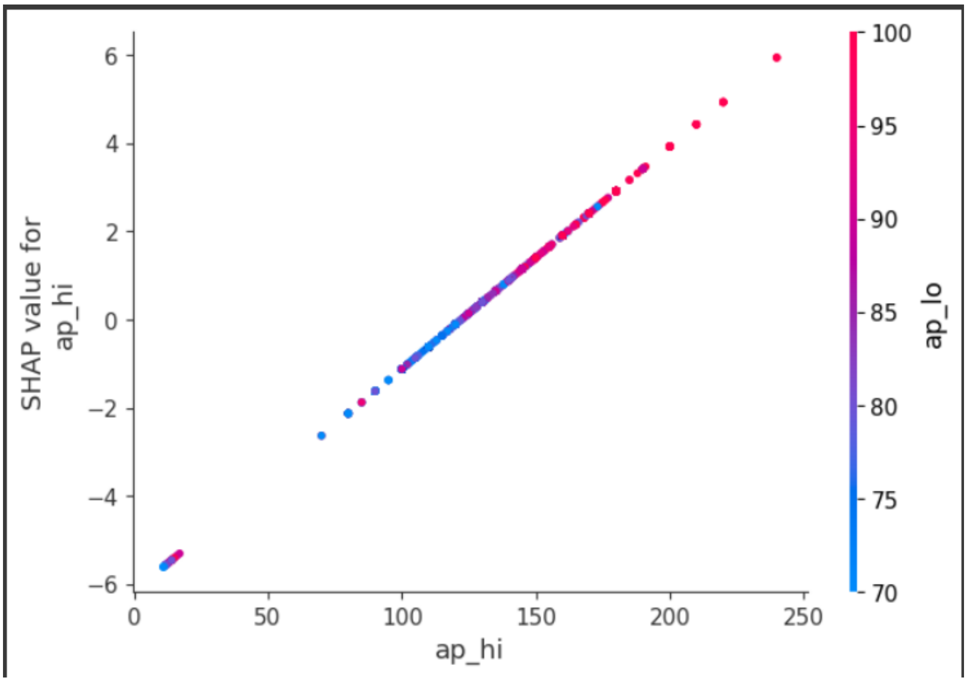
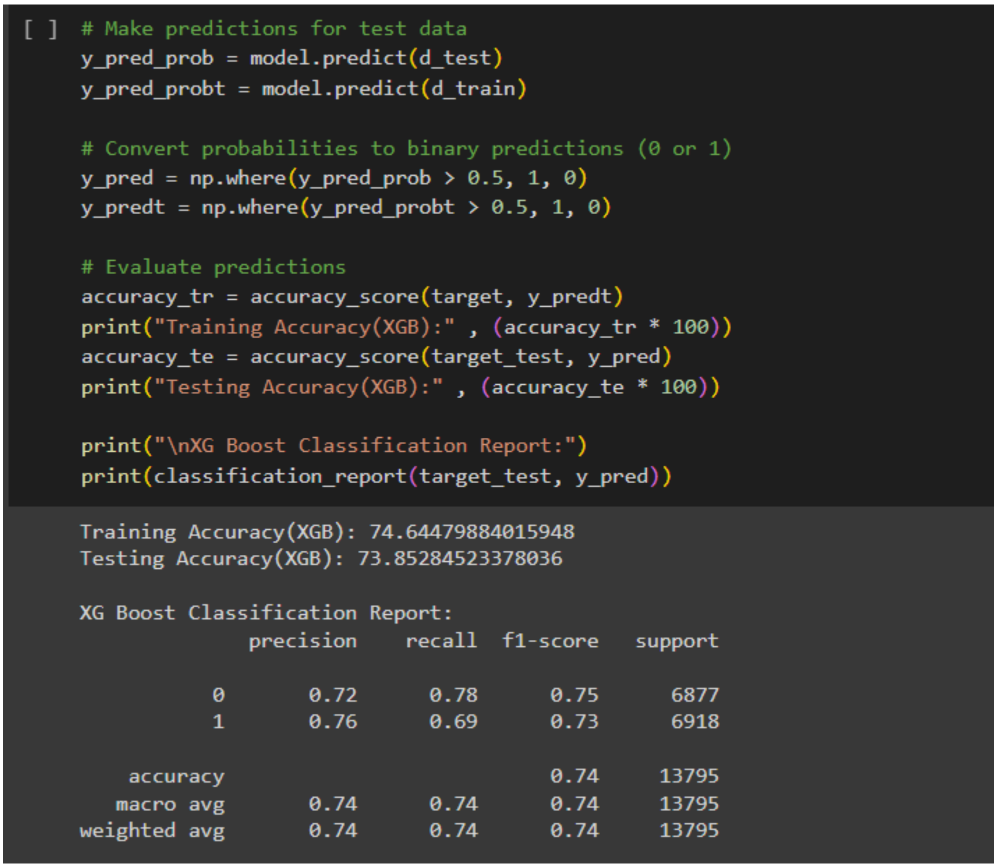


Figure 6.11: Dependency plot for LR (Shows the dependency between ap_hi (systolic BP) and ap_lo (diastolic BP) to be in a linear trend)

XGB: Out of the six models implemented, XGB showcased promise in CVD prediction giving an accuracy of 73.85%. It is one of the boosting algorithms (ensemble technique) and hence integration with SHAP increases the transparency and interpretability. When we look at the feature importance plot, we see that age was the most important parameter in terms of weight.



```
# Make predictions for test data
y_pred_prob = model.predict(d_test)
y_pred_probt = model.predict(d_train)

# Convert probabilities to binary predictions (0 or 1)
y_pred = np.where(y_pred_prob > 0.5, 1, 0)
y_predt = np.where(y_pred_probt > 0.5, 1, 0)

# Evaluate predictions
accuracy_tr = accuracy_score(target, y_predt)
print("Training Accuracy(XGB):" , (accuracy_tr * 100))
accuracy_te = accuracy_score(target_test, y_pred)
print("Testing Accuracy(XGB):" , (accuracy_te * 100))

print("\nXG Boost Classification Report:")
print(classification_report(target_test, y_pred))
```

```
Training Accuracy(XGB): 74.64479884015948
Testing Accuracy(XGB): 73.85284523378036

XG Boost Classification Report:
              precision    recall  f1-score   support

           0       0.72      0.78      0.75      6877
           1       0.76      0.69      0.73      6918

    accuracy                           0.74     13795
   macro avg       0.74      0.74      0.74     13795
weighted avg       0.74      0.74      0.74     13795
```

Figure 6.12: XGB Classifier Evaluation Metrics

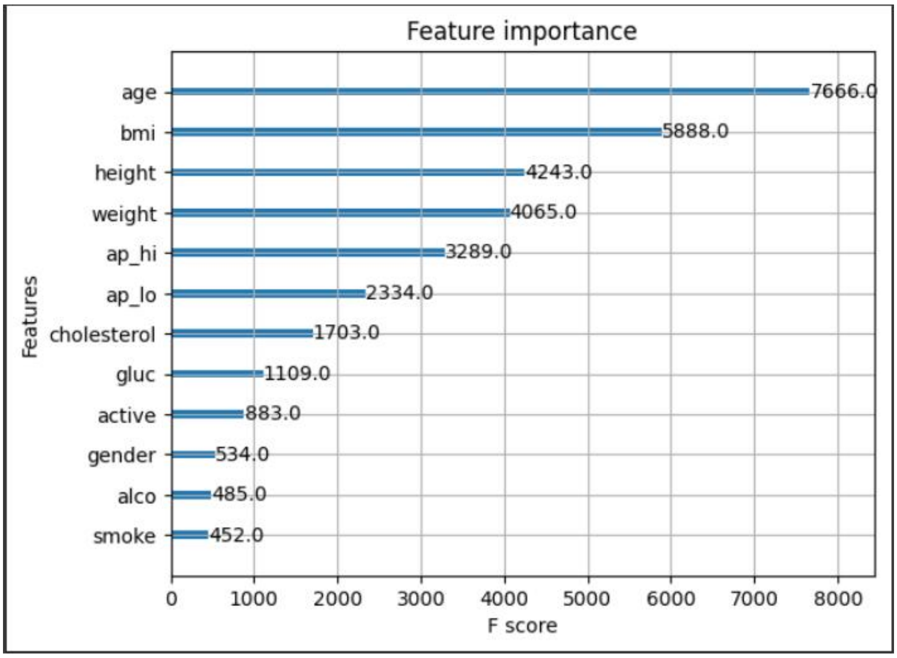


Figure 6.13: Feature Importance (XGB)

Beeswarm Summary Plot for XGB: Here, we notice that ap_hi has the highest average SHAP value magnitude leading it to be the most prevalent feature in model prediction. The high values of ap_hi (systolic blood pressure) lead to greater impact in causing CVD and vice-versa. Infact, ap_hi is the most important factor in most of the models that we implemented, making it the key feature for CVD prediction.

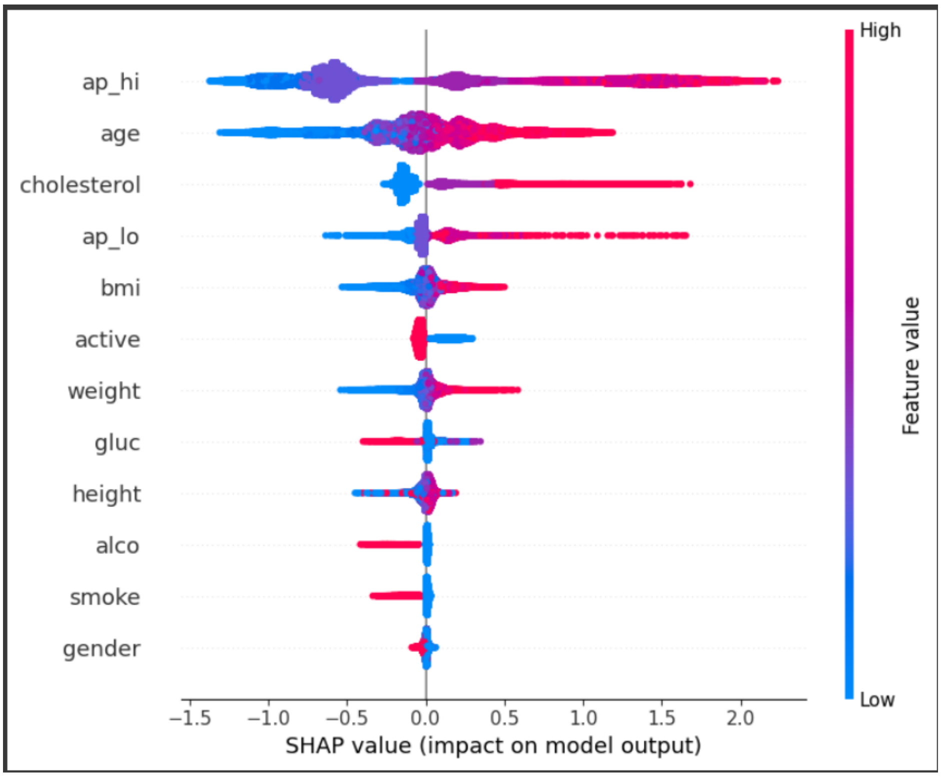


Figure 6.14: Beeswarm Plot for XGB

Force plots for XGB: Here, the predictive value is -0.49 which means that the features result in a negative outcome. The attributes in blue such as the alcohol intake, cholesterol and ap_lo (diastolic blood pressure) all being normal for this particular prediction, push the results to be more driven towards a person to not have CVD.

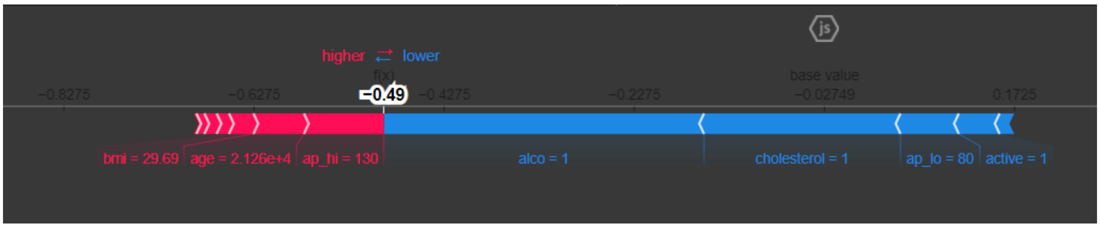


Figure 6.15: Force Plot for XGB

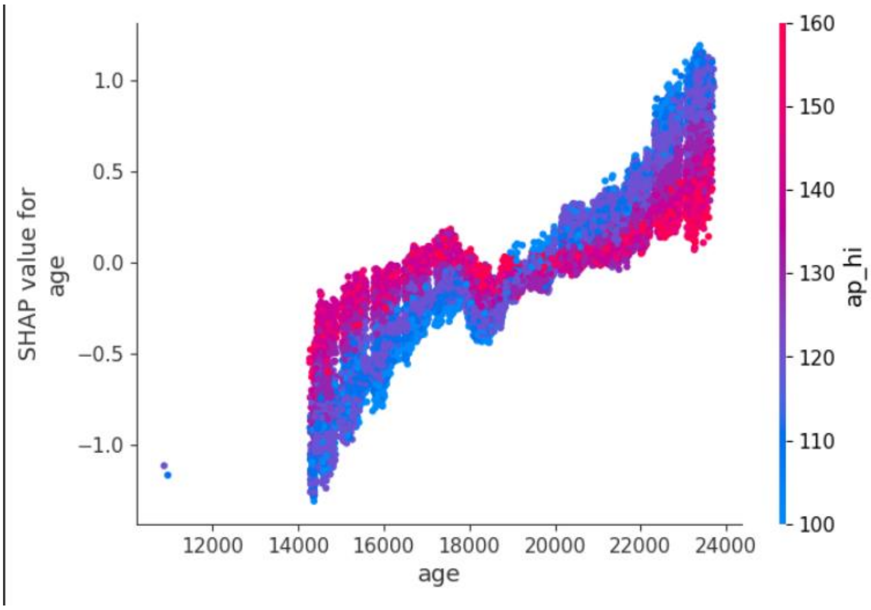
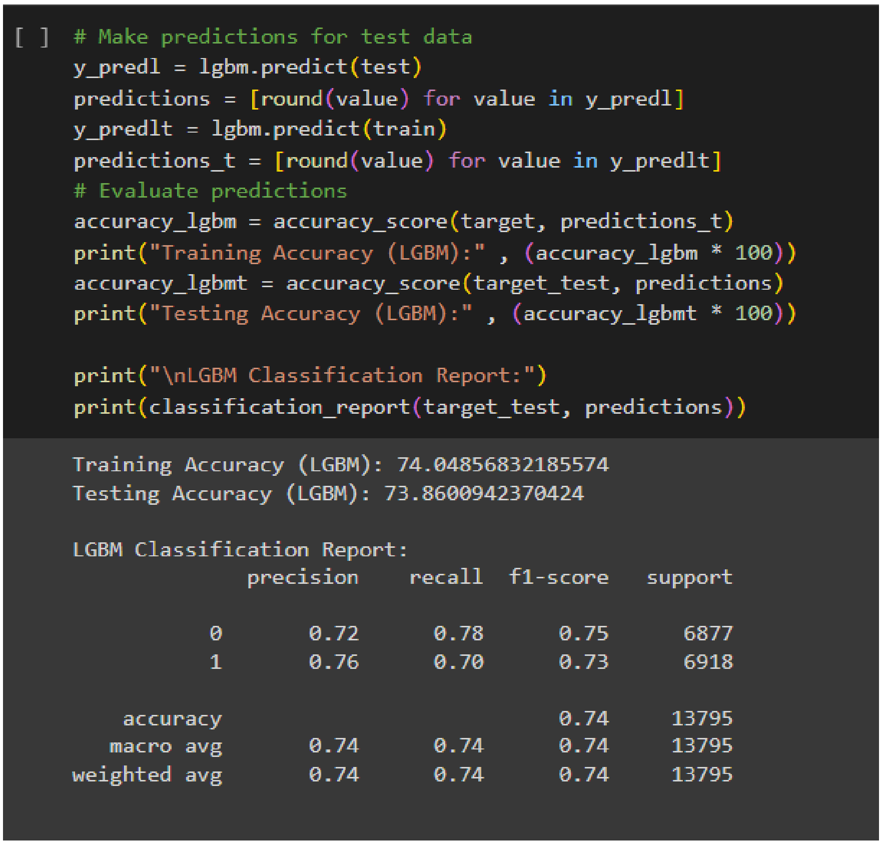


Figure 6.16: Dependency Plot for XGB (Shows the dependency between age and ap_hi (systolic BP))

LGBM: This model turned out to be the best suited model for CVD prediction based on the evaluation metrics and the SHAP metrics as well. It gave us an accuracy of 73.86% and it also had the least score difference. When we look at the feature importance plot, we see that age was the most important parameter in terms of weight followed by ap_hi (systolic blood pressure).



```
[ ]  # Make predictions for test data
     y_predl = lgbm.predict(test)
     predictions = [round(value) for value in y_predl]
     y_predlt = lgbm.predict(train)
     predictions_t = [round(value) for value in y_predlt]
     # Evaluate predictions
     accuracy_lgbm = accuracy_score(target, predictions_t)
     print("Training Accuracy (LGBM):" , (accuracy_lgbm * 100))
     accuracy_lgbmt = accuracy_score(target_test, predictions)
     print("Testing Accuracy (LGBM):" , (accuracy_lgbmt * 100))

     print("\nLGBM Classification Report:")
     print(classification_report(target_test, predictions))
```

```
Training Accuracy (LGBM): 74.04856832185574
Testing Accuracy (LGBM): 73.8600942370424

LGBM Classification Report:
              precision    recall  f1-score   support

           0       0.72      0.78      0.75      6877
           1       0.76      0.70      0.73      6918

    accuracy                           0.74     13795
   macro avg       0.74      0.74      0.74     13795
weighted avg       0.74      0.74      0.74     13795
```

Figure 6.17: LGBM Metrics

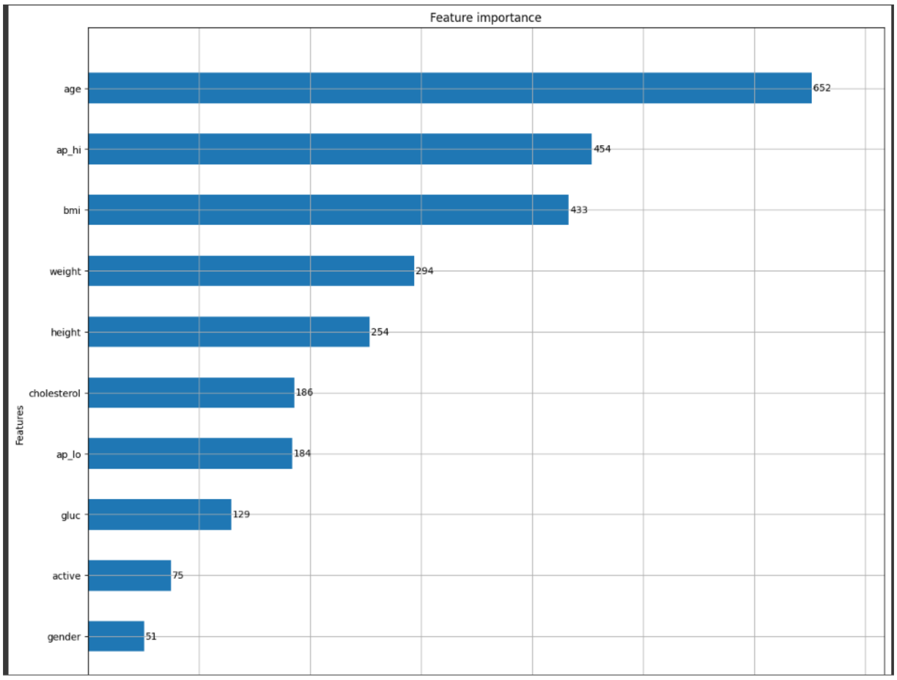


Figure 6.18: Feature Importance (LGBM)

Summary plot for LGBM: This plot shows that ap_hi (systolic blood pressure) has the highest absolute magnitude of the SHAP values, with the longest bars appearing at the top of the plot. Hence, it is the most significant feature in a model's predictions followed by age and then cholesterol levels.

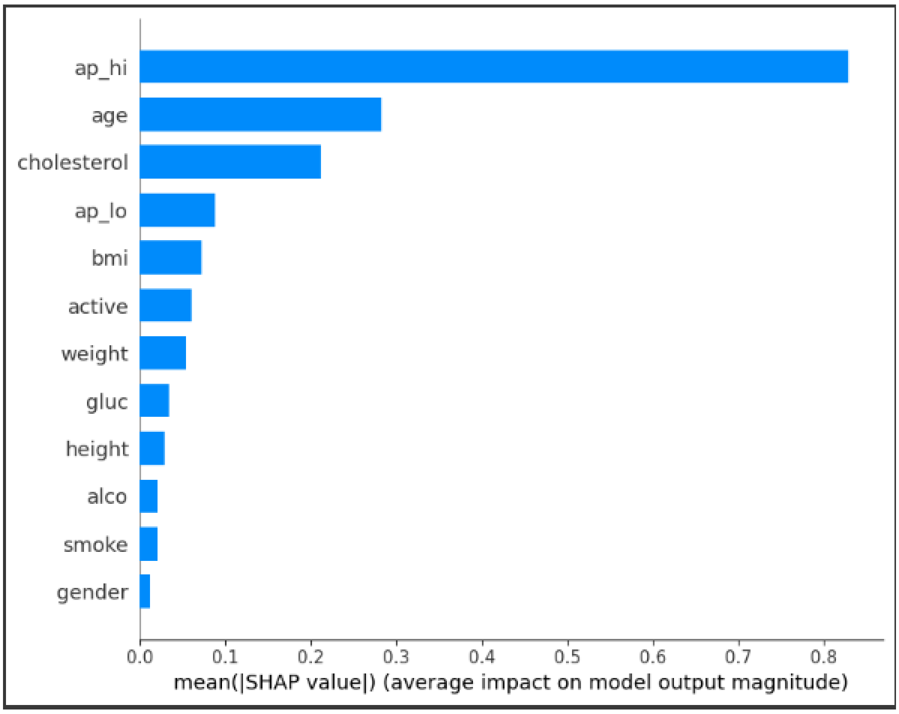


Figure 6.19: Bar Summary Plot for LGBM

Beeswarm Summary Plot for LGBM : Here, we notice that ap_hi has the highest average SHAP value magnitude leading it to be the most prevalent feature in model prediction. The high values of ap_hi (systolic blood pressure) lead to greater impact in causing CVD and vice-versa. Infact, ap_hi is the most important factor in most of the models that we implemented, making it the key feature for CVD prediction.

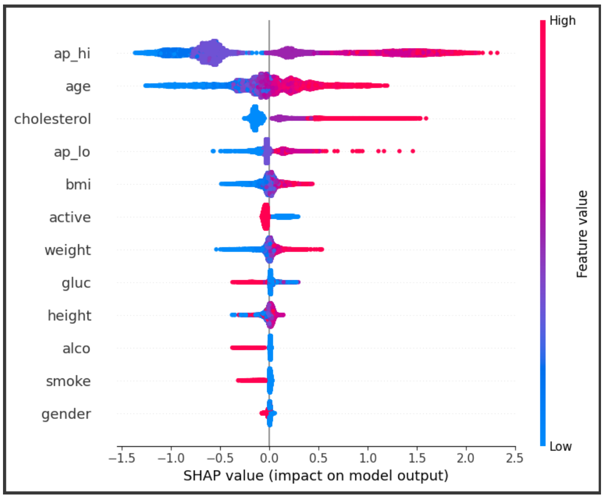


Figure 6.20: Beeswarm Plot for LGBM

Force plots for LGBM: Here, the predictive value is -0.45 which means that the features result in a negative outcome. The attributes in blue such as the ap_hi (systolic blood pressure), cholesterol, bmi and ap_lo (diastolic blood pressure) all being normal for this particular prediction, push the results to be more driven towards a person to not have CVD despite the fact that age which is more in this case is pushing the results in the opposite direction.

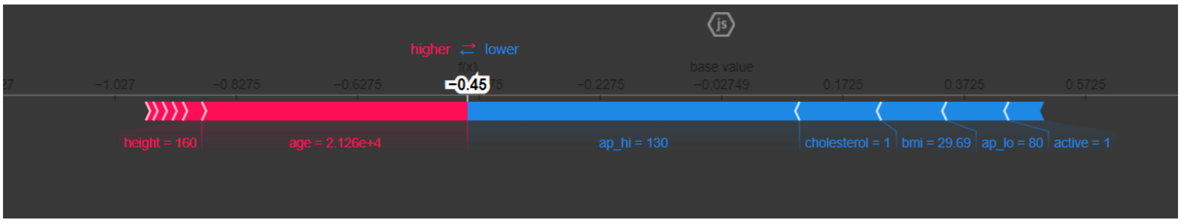
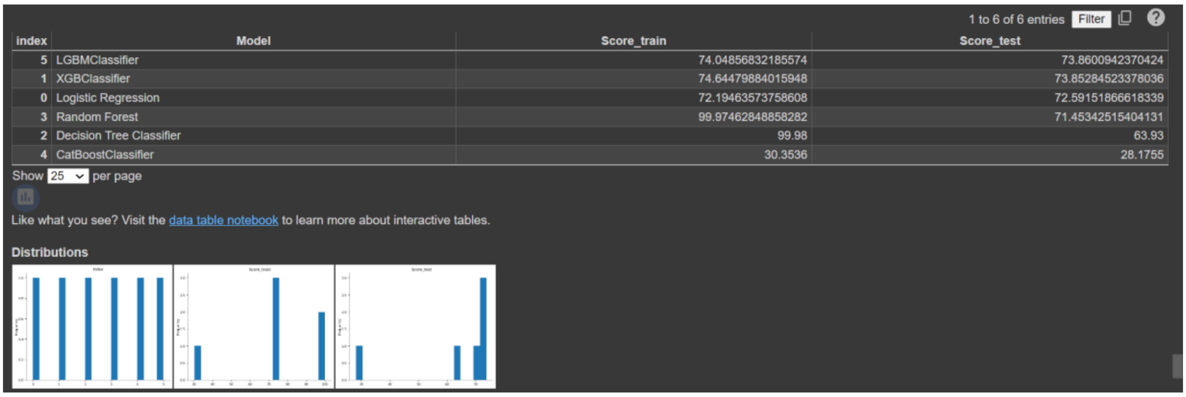


Figure 6.21: Force Plot for LGBM

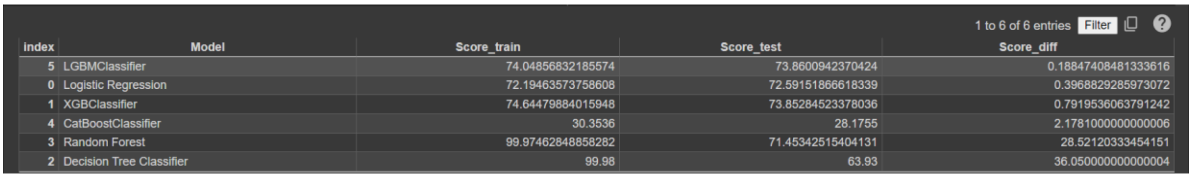
Evaluation: We evaluated the models based on the evaluation metrics including accuracy, precision, recall, f1 score followed by comparing the training and testing score and the score difference. All these criterias have pushed LGBM to be the best suitable model for our application.



index	Model	Score_train	Score_test
5	LGBMClassifier	74.04856832185574	73.8600942370424
1	XGBClassifier	74.64479884015948	73.85284523378036
0	Logistic Regression	72.19463573758608	72.59151866618339
3	Random Forest	99.97462848858282	71.45342515404131
2	Decision Tree Classifier	99.98	63.93
4	CatBoostClassifier	30.3536	28.1755

Like what you see? Visit the data table notebook to learn more about interactive tables.

Figure 6.22: Training and Testing Scores of the Models



index	Model	Score_train	Score_test	Score_diff
5	LGBMClassifier	74.04856832185574	73.8600942370424	0.18847408481333616
0	Logistic Regression	72.19463573758608	72.59151866618339	0.3968829285973072
1	XGBClassifier	74.64479884015948	73.85284523378036	0.7919536063791242
4	CatBoostClassifier	30.3536	28.1755	2.1781000000000006
3	Random Forest	99.97462848858282	71.45342515404131	28.52120333454151
2	Decision Tree Classifier	99.98	63.93	36.050000000000004

Figure 6.23: Best Suited Models based on Scores Difference

7. RESULTS & DISCUSSION

Models	**Accuracies**
LGBM	*73.8601*
XGB	*73.8528*
Logistic Regression	*71.8883*
Random Forest	*71.4099*
Decision Tree	*64.1000*
ANN	*49.8500*
CatBoost	*30.3500*

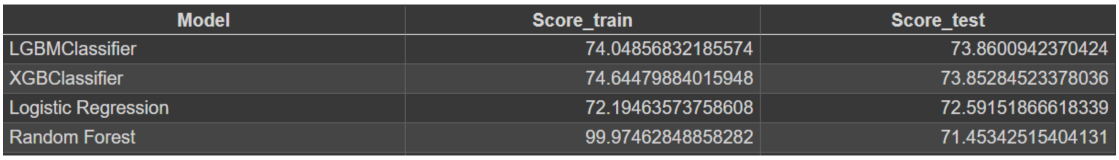
Table 7.1: Accuracies of Different Models

In the project, the observed results comprise a variety of quantitative indicators and qualitative insights obtained through the utilisation of explainable AI techniques and predictive models. The study uses quantitative measures to assess model performance. This allows for an evaluation of the models' predictive power for cases of cardiovascular disease. Logistic regression taken as the base algorithm, gave us an accuracy of 71.8883 and while the systolic pressure showcased more correlation, the SHAP values suggested that the weight was the most prevalent feature having the highest SHAP value. Out of the predictive models chosen, XGB and LGBM outperformed having an accuracy of about 73.8528 and 73.8601 respectively. Other models such as CatBoost, Decision Tree and Neural networks didn’t show much of a promise regarding the prediction of CVDs.

Furthermore, the analysis explores how interpretable the models are, using SHAP values to reveal information about the significance of individual forecasts, global trends, and feature importance. Through the various plots obtained through SHAP, we notice that the systolic pressure was one of the features which was most likely to be of great importance in the predictions. All the more, we observed the various plots such as the dependency plots, summary plots, waterfall plots, force plots,etc. for the different algorithms based on training and testing sets as well as for individual predictions giving us more insights in each of the scenarios. Summary plots assess the impact of risk factors on the prediction of CVD, which helps prioritise them. Healthcare practitioners can concentrate on interventions aimed at these parameters to reduce the risk of disease and enhance patient outcomes by identifying the most important qualities. Force plots improve trust and transparency in the model's decision-making process by helping doctors comprehend the reasoning behind each prediction. They help medical practitioners make sense of the relative importance of many risk factors for each patient and effectively communicate with them about their cardiovascular health. Dependency charts make it easier to see how different risk factors—such as age, blood pressure, and cholesterol—affect the chance of developing cardiovascular disease. They draw attention to how modifications to these variables affect the model's forecasts, supporting risk assessment and patient grouping. By giving clear explanations of the models' decision-making procedures, this strengthens clinician adoption of the models and improves patient outcomes. The initiative also adds to the body of knowledge in preventive cardiology by identifying new risk factors for cardiovascular disease and validating existing risk factors.

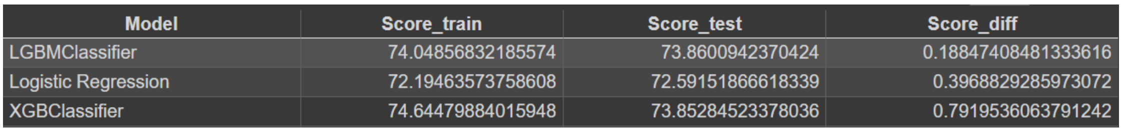
In order to create prediction models for complicated datasets, this research project included a variety of model architectures, such as neural networks, tree-based models, and linear models. When used in conjunction with neural networks and tree-based models like Random Forest, XGBoost, and LGBM, SHAP (SHapley Additive exPlanations) proved to be especially beneficial, according to our examination into the interpretability and transparency of these models. By quantifying the effect of input features on the result, SHAP improved interpretability and contributed to a thorough knowledge of the model's predictions.

But even though SHAP proved to be quite effective, there were issues with the TreeExplainer part of the SHAP system. TreeExplainer's applicability to different model architectures is limited because it is designed for use with only a certain set of algorithms such as XGBoost, CatBoost, decision trees, and random forests. As a result, KernelExplainer—a different approach that works with a wider variety of algorithms can be used as an alternative. KernelExplainer is a versatile tool, however its implementation is computationally expensive and resource-intensive due to its significant computational overhead which poses a problem.



Model	Score_train	Score_test
LGBMClassifier	74.04856832185574	73.8600942370424
XGBClassifier	74.64479884015948	73.85284523378036
Logistic Regression	72.19463573758608	72.59151866618339
Random Forest	99.97462848858282	71.45342515404131

Figure 7.1: Model Results



Model	Score_train	Score_test	Score_diff
LGBMClassifier	74.04856832185574	73.8600942370424	0.18847408481333616
Logistic Regression	72.19463573758608	72.59151866618339	0.3968829285973072
XGBClassifier	74.64479884015948	73.85284523378036	0.7919536063791242

Figure 7.2: Model Results on Score Differences

Our comprehensive evaluation of various criteria revealed that the LGBM Classifier emerged as the optimal model choice. This conclusion was drawn from its remarkable performance on both individual training and test scores as well as minimal score disparity. Notably, the transparency and interpretability of the LGBM Classifier were initially problematic due to its ensemble nature. But our implementation framework's strategic integration of the SHAP (SHapley Additive exPlanations) technique allowed us to effectively solve these challenges. This collaborative approach strengthened the model's overall credibility and suitability for use in practical situations, in addition to improving its interpretability.

All things considered, the project shows how to use explainable AI to improve predictability of cardiovascular illness and increase transparency and trust in those forecasts, which will ultimately improve patient care and healthcare decision-making. Although SHAP shows great promise in improving interpretability and transparency in predictive modelling, especially when combined with neural networks and tree-based models, the drawbacks of certain explanatory components make other approaches worth exploring carefully. Subsequent investigations could concentrate on reducing computing burdens and broadening the scope of model architectures that SHAP explanation techniques can be used to, improving the interpretability and practicality of predictive models in a variety of contexts. Furthermore, our study framework’s integration with QML has intriguing prospects for expanding the frontiers of conventional machine learning theories and improving the performance, efficiency, and interpretability of predictive models. We can solve practical problems in predictive modelling, such as those pertaining to the prediction of cardiovascular illness, by utilising the special powers of quantum computing.

8. SUMMARY

The findings of this research project include a thorough examination of both quantitative indicators and qualitative insights that were acquired by combining explainable AI methods with predictive models. The predictive ability of models for cardiovascular disease (CVD) was assessed quantitatively, using logistic regression as the foundational technique. When it came to CVD prediction, models like CatBoost, Decision Trees, and Neural Networks showed little potential, but XGBoost and LGBM outperformed the other predictive models that were chosen. The study also explored the interpretability of models that use SHAP values to reveal information about feature significance, worldwide trends, and individual predictions. Dependency plots, summary plots, waterfall plots, force plots, and other visualisation approaches offered important insights into the importance of risk variables and the model decision-making process. The risk assessment, patient stratification, and efficient communication between medical professionals and patients on cardiovascular health were made easier by these visualisations. Furthermore, by discovering new risk factors for CVD and confirming those that already exist, our research advances the field of preventive cardiology. Through a combination of different model architectures, such as neural networks, tree-based models, and linear models, we investigated how well SHAP could improve interpretability and transparency. But there were issues with the TreeExplainer part of SHAP, therefore it was necessary to use an alternative such as KernalExplainer but it was computationally expensive. After being integrated with SHAP, the LGBM Classifier proved to be the best option, exhibiting exceptional performance and improved interpretability. This cooperative approach enhanced the model's applicability for real-world scenarios while also bolstering its trustworthiness. To sum up, the research highlights how explainable AI can increase the prediction of cardiovascular disease and boost transparency and confidence in predictive models, all of which will eventually improve patient care and healthcare decision-making. Prospective research avenues could delve into methods for expanding the application of SHAP explanation approaches and examine the incorporation of Quantum Machine Learning (QML) to enhance predictive modelling potential in the healthcare industry and other domains.

9. REFERENCES

Weblinks:

1. https://en.wikipedia.org/wiki/Cardiovascular_disease
2. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10621606/
3. https://pubmed.ncbi.nlm.nih.gov/38248005/
4. https://link.springer.com/article/10.1007/s11936-020-00814-0
5. https://www.nature.com/articles/s41598-023-49673-2
6. https://ieeexplore.ieee.org/abstract/document/8550857
7. https://www.sciencedirect.com/science/article/abs/pii/S1746809423010777
8. https://www.researchsquare.com/article/rs-3068941/v1

Journals:

1. Kırboğa KK, Küçüksille EU. Identifying Cardiovascular Disease Risk Factors in Adults with Explainable Artificial Intelligence. Anatol J Cardiol. 2023 Nov 1;27(11):657-663. doi: 10.14744/AnatolJCardiol.2023.3214. Epub 2023 Aug 25. PMID: 37624075; PMCID: PMC10621606.

2. Wesołowski S, Lemmon G, Hernandez EJ, Henrie A, Miller TA, Weyhrauch D, Puchalski MD, Bray BE, Shah RU, Deshmukh VG, Delaney R, Yostl HJ, Eilbeck K, Tristani-Firouzi M, Yandell M. An explainable artificial intelligence approach for predicting cardiovascular outcomes using electronic health records. PLOS Digit Health. 2022;1(1):e0000004. doi: 10.1371/journal.pdig.0000004. Epub 2022 Jan 18. PMID: 35373216; PMCID: PMC8975108.

3. Lundström J, Hashemi AS, Tiwari P. Explainable Graph Neural Networks for Atherosclerotic Cardiovascular Disease. Stud Health Technol Inform. 2023 May 18;302:603-604. doi: 10.3233/SHTI230214. PMID: 37203757.

4. Krittanawong, C., Virk, H.U.H., Bangalore, S. et al. Machine learning prediction in cardiovascular diseases: a meta-analysis. Sci Rep 10, 16057 (2020)

5. Salah H, Srinivas S. Explainable machine learning framework for predicting long-term cardiovascular disease risk among adolescents. Sci Rep. 2022 Dec 19;12(1):21905. doi: 10.1038/s41598-022-25933-5. PMID: 36536006; PMCID: PMC9763353.

6. Srinivasu PN, Sirisha U, Sandeep K, Praveen SP, Maguluri LP, Bikku T. An Interpretable Approach with Explainable AI for Heart Stroke Prediction. Diagnostics (Basel). 2024 Jan 5;14(2):128. doi: 10.3390/diagnostics14020128. PMID: 38248005; PMCID: PMC10813874.

7. van Smeden M, Heinze G, Van Calster B, Asselbergs FW, Vardas PE, Bruining N, de Jaegere P, Moore JH, Denaxas S, Boulesteix AL, Moons KGM. Critical appraisal of artificial intelligence-based prediction models for cardiovascular disease. Eur Heart J. 2022 Aug 14;43(31):2921-2930. doi: 10.1093/eurheartj/ehac238. PMID: 35639667; PMCID: PMC9443991.

8. Schlesinger, D.E., Stultz, C.M. Deep Learning for Cardiovascular Risk Stratification. Curr Treat Options Cardio Med 22, 15 (2020)

9. Guldogan, E., Yagin, F.H., Pinar, A. et al. A proposed tree-based explainable artificial intelligence approach for the prediction of angina pectoris. Sci Rep 13, 22189 (2023)

10. Makimoto, H., Kohro, T. Adopting artificial intelligence in cardiovascular medicine: a scoping review. Hypertens Res (2023)

11. Siegersma KR, Leiner T, Chew DP, Appelman Y, Hofstra L, Verjans JW. Artificial intelligence in cardiovascular imaging: state of the art and implications for the imaging cardiologist. Neth Heart J. 2019 Sep;27(9):403-413. doi: 10.1007/s12471-019-01311-1. PMID: 31399886; PMCID: PMC6712136.

12. Moreno-Sánchez PA. Improvement of a prediction model for heart failure survival through explainable artificial intelligence. Front Cardiovasc Med. 2023

Aug 1;10:1219586. doi: 10.3389/fcvm.2023.1219586. PMID: 37600061; PMCID: PMC10434534.

13. Damen, Johanna AAG, et al. "Prediction models for cardiovascular disease risk in the general population: systematic review." bmj 353 (2016).

14. K. G. Dinesh, K. Arumugaraj, K. D. Santhosh and V. Mareeswari, "Prediction of Cardiovascular Disease Using Machine Learning Algorithms," 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), Coimbatore, India, 2018, pp. 1-7, doi: 10.1109/ICCTCT.2018.8550857.

15. Westerlund, Annie M., Johann S. Hawe, Matthias Heinig, and Heribert Schunkert. 2021. "Risk Prediction of Cardiovascular Events by Exploration of Molecular Data with Explainable Artificial Intelligence" International Journal of Molecular Sciences 22, no. 19: 10291.

16. Abdulsalam, Ghada, Souham Meshoul, and Hadil Shaiba. "Explainable Heart Disease Prediction Using Ensemble-Quantum Machine Learning Approach." Intell. Autom. Soft Comput 36 (2023): 761-779.

17. ElShawi, Radwa, et al. "Interpretability in healthcare: A comparative study of local machine learning interpretability techniques." Computational Intelligence 37.4 (2021): 1633-1650.

18. Prendin, Francesco, et al. "The importance of interpreting machine learning models for blood glucose prediction in diabetes: an analysis using SHAP." Scientific reports 13.1 (2023): 16865.

19. Mohanty, Somya D., et al. "Machine learning for predicting readmission risk among the frail: Explainable AI for healthcare." Patterns 3.1 (2022).

20. Samek, Wojciech, et al. "Interpreting the predictions of complex ml models by layer-wise relevance propagation." arXiv preprint arXiv:1611.08191 (2016).

21. Tang, Qinhua, Xingxing Cen, and Changqing Pan. "Explainable and efficient deep early warning system for cardiac arrest prediction from electronic health records." Mathematical Biosciences and Engineering 19.10 (2022): 9825-9841.

22. Castronuovo, Gianfranco, et al. "Analyzing the Interactions between Environmental Parameters and Cardiovascular Diseases Using Random Forest and SHAP Algorithms." Reviews in Cardiovascular Medicine 24.11 (2023): 330.

23. Miranda, Eka, et al. "Understanding Arteriosclerotic Heart Disease Patients Using Electronic Health Records: A Machine Learning and Shapley Additive exPlanations Approach." Healthcare Informatics Research 29.3 (2023): 228-238.

24. Faisal, Alina, et al. "A COMPARATIVE STUDY OF PREDICTIVE SUPERVISED-MACHINE LEARNING ALGORITHMS ON CARDIOVASCULAR DISEASES (CVD)." Journal of Population Therapeutics and Clinical Pharmacology 30.19 (2023): 1159-1177.

25. Akkur, Erkan. "Prediction of Cardiovascular Disease Based on Voting Ensemble Model and SHAP Analysis." Sakarya University Journal of Computer and Information Sciences 6.3 (2023): 226-238.

26. M. Athanasiou, K. Sfrintzeri, K. Zarkogianni, A. C. Thanopoulou and K. S. Nikita, "An explainable XGBoost–based approach towards assessing the risk of cardiovascular disease in patients with Type 2 Diabetes Mellitus," 2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE), Cincinnati, OH, USA, 2020, pp. 859-864, doi: 10.1109/BIBE50027.2020.00146.

27. Abdellatif, Abdallah, et al. "Computational detection and interpretation of heart disease based on conditional variational auto-encoder and stacked ensemble-learning framework." Biomedical Signal Processing and Control 88 (2024): 105644.

28. Rustamov, Zahiriddin, et al. "Enhancing Cardiovascular Disease Prediction: A Domain Knowledge-Based Feature Selection and Stacked Ensemble Machine Learning Approach." (2023).

29. Chen, Bangwei, et al. "Machine learning improves risk stratification of coronary heart disease and stroke." Annals of Translational Medicine 10.21 (2022).

30. Liu, Jimin, et al. "Predictive classifier for cardiovascular disease based on stacking model fusion." Processes 10.4 (2022): 749.

APPENDIX A – SAMPLE CODE

```
data = pd.read_csv("cardio_train.csv", sep=";")

def clean_dataset(df):
    assert isinstance(df, pd.DataFrame), "df needs to be
a pd.DataFrame"
    df.dropna(inplace=True)
    indices_to_keep = ~df.isin([np.nan, np.inf, -
np.inf]).any(axis=1)
    return df[indices_to_keep].astype(np.float64)

clean_dataset(data)

data.drop("id",axis=1,inplace=True)
data.drop_duplicates(inplace=True)
data["bmi"] = data["weight"] / (data["height"]/100)**2
#Removing outliers
out_filter = ((data["ap_hi"]>250) | (data["ap_lo"]>200))
data = data[~out_filter]
len(data)

out_filter2 = ((data["ap_hi"] < 0) | (data["ap_lo"] < 0))
data = data[~out_filter2]

data.info()

yp.ProfileReport(data)
```

```
target_name = 'cardio'

data_target = data[target_name]

data = data.drop([target_name], axis=1)


train, test, target, target_test = train_test_split(data,
data_target, test_size=0.2, random_state=0)


train.head(3)


test.head(3)


train.info()


test.info()


# Spliting training set to validation set

Xtrain, Xval, Ytrain, Yval = train_test_split(train,
target, test_size=0.2, random_state=0)


# Logistic Regression


logreg = LogisticRegression()

logreg.fit(train, target)

y_pred_logreg = logreg.predict(test)

accuracy_logreg = accuracy_score(target_test,
y_pred_logreg)

print(f"Logistic Regression Accuracy: {accuracy_logreg}")

print("\nLogistic Regression Classification Report:")
print(classification_report(target_test, y_pred_logreg))

coeff_df = pd.DataFrame(train.columns.delete(0))
coeff_df.columns = ['Feature']
coeff_df["Correlation"] = pd.Series(logreg.coef_[0])

coeff_df.sort_values(by='Correlation', ascending=False)

# Explain logistic regression(base model) predictions
using SHAP
explainer_log = shap.Explainer(logreg, train)
shap_values_log = explainer_log(test)

# Visualize SHAP summary plot
shap.summary_plot(shap_values_log, test,
feature_names=test.columns)

# Visualize individual predictions with force plots
shap.initjs()
shap.plots.force(shap_values_log[0])

# Visualize dependence plots for individual features
shap.dependence_plot('ap_hi', shap_values_log.values,
test)
plt.show()

shap.summary_plot(shap_values_log,test, plot_type="bar",
class_names=['Healthy', 'Disease'])
```

```
d_train =xgb.DMatrix(train, label= target)
d_test =xgb.DMatrix(test, label= target_test)

# XGB

params = {
    "eta": 0.01,
    "objective": "binary:logistic",
    "subsample": 0.5,
    "base_score": np.mean(Ytrain),
    "eval_metric": "logloss",
}
model = xgb.train(
    params,
    d_train,
    5000,
    evals=[(d_test, "test")],
    verbose_eval=100,
    early_stopping_rounds=20,
)

# Make predictions for test data
y_pred_prob = model.predict(d_test)

# Convert probabilities to binary predictions (0 or 1)
y_pred = np.where(y_pred_prob > 0.5, 1, 0)

# Evaluate predictions
```

```
accuracy = accuracy_score(target_test, y_pred)
print("Accuracy(XGB):" , (accuracy * 100))

xgb.plot_importance(model)
plt.show()
plt.close()

xgb.plot_importance(model, importance_type="cover")
plt.title('xgb.plot_importance(model,
importance_type="cover")')
plt.show()

xgb.plot_importance(model, importance_type="gain")
plt.title('xgb.plot_importance(model,
importance_type="gain")')
plt.show()

# Explain XGB predictions using SHAP
explainer = shap.TreeExplainer(model)
shap_values = explainer.shap_values(train)

shap.initjs()
# Visualize SHAP summary plot
shap.summary_plot(shap_values, train)

shap.initjs()
shap.force_plot(explainer.expected_value, shap_values[0,
:], train.iloc[0, :])
```

```
shap.summary_plot(shap_values, train, plot_type="bar")

for name in train.columns:
    shap.dependence_plot(name, shap_values, train,
display_features= train)

# Decision Tree Classifier

decision_tree = DecisionTreeClassifier()
decision_tree.fit(train, target)
acc_decision_tree = round(decision_tree.score(train,
target) * 100, 2)
acc_decision_tree

acc_test_decision_tree = round(decision_tree.score(test,
target_test) * 100, 2)
acc_test_decision_tree

# Print accuracy
print("Training Accuracy (Decision Tree):",
acc_decision_tree)
print("Testing Accuracy (Decision Tree):",
acc_test_decision_tree)

# Explain decision tree predictions using SHAP
explainer_dt = shap.TreeExplainer(decision_tree)
shap_values_dt = explainer_dt(test)

# Visualize SHAP summary plot
shap.summary_plot(shap_values_dt[..., 0], test)
shap.initjs()
# Visualize individual predictions with force plots
shap.force_plot(explainer_dt.expected_value[0],
shap_values_dt.values[0, :])

shap.summary_plot(shap_values_dt[..., 0], train,
plot_type="bar")

# Random Forest

rf = RandomForestClassifier()
rf.fit(train, target)
y_pred = rf.predict(test)
y_pred_rf_train = rf.predict(train)

accuracy = accuracy_score(target_test, y_pred)
print("Accuracy (Random Forest):", accuracy*100)

df_fi = pd.DataFrame()
df_fi['feature'] = train.columns
df_fi['importance'] = rf.feature_importances_

df_fi.sort_values('importance', ascending=False).head(10)

# Explain Random Forest predictions using SHAP
train_sample = train.sample(500, random_state=12)
explainer_rf = shap.TreeExplainer(rf)
shap_values_rf = explainer_rf.shap_values(train_sample)
```

```
# Visualize SHAP summary plot

shap.summary_plot(shap_values_rf[...,0], train_sample)


shap.initjs()

# Visualize individual predictions with force plots

shap.force_plot(explainer_rf.expected_value[0],
shap_values_rf[...,0], train_sample)


# Visualize SHAP summary plot

shap.summary_plot(shap_values_rf[...,0],
train_sample,plot_type="bar")


shap.dependence_plot("bmi", shap_values_rf[...,0],
train_sample)


#CatBoost


Cat_Boost = CatBoostRegressor(iterations=300,
learning_rate=0.1, random_seed=123)

Cat_Boost.fit(train, target, verbose=False, plot=False)

Cat_Boost.score(train, target)

acc_CatBoost = round(Cat_Boost.score(train, target) *
100, 2)

print("Accuracy (CatBoost): " , acc_CatBoost)


# Explain model predictions using SHAP

explainer_cb = shap.TreeExplainer(Cat_Boost)

shap_values_cb = explainer_cb(train)
```

```
# Visualize SHAP summary plot
shap.summary_plot(shap_values_cb)


shap.initjs()
# Visualize the first prediction's explanation
shap.plots.force(shap_values_cb[0, ...])


shap.summary_plot(shap_values_cb, train, plot_type="bar")


for name in train.columns:
  shap.dependence_plot(name, shap_values_cb.values,
train)


shap.plots.waterfall(shap_values_cb[0])


dd_train = lgb.Dataset(train, label=target)
dd_test = lgb.Dataset(test, label=target_test)


#LGBM


params = {
    "max_bin": 512,
    "learning_rate": 0.05,
    "boosting_type": "gbdt",
    "objective": "binary",
    "metric": "binary_logloss",
    "num_leaves": 10,
    "verbose": -1,
    "min_data": 100,
    "boost_from_average": True,
    "early_stopping_round": 50,
}

lgbm = lgb.train(
    params,
    dd_train,
    10000,
    valid_sets=[dd_test],
)

# Make predictions for test data
y_pred = lgbm.predict(test)
predictions = [round(value) for value in y_pred]
# Evaluate predictions
accuracy = accuracy_score(target_test, predictions)
print("Accuracy (LGBM):" , (accuracy * 100))

fig =  plt.figure(figsize = (15,15))
axes = fig.add_subplot(111)
lgb.plot_importance(lgbm,ax = axes,height = 0.5)
plt.show()
plt.close()

# Explain LGBM predictions using SHAP
explainer_lgb = shap.TreeExplainer(model)
shap_values_lgb = explainer_lgb(test)
```

```
# Visualize SHAP summary plot
shap.summary_plot(shap_values_lgb)


shap.initjs()
# Visualize individual predictions with force plots
shap.force_plot(explainer_lgb.expected_value,
shap_values_lgb.values[1, :], train.iloc[0, :])


shap.summary_plot(shap_values_lgb, train,
plot_type="bar")
```

Enhancing Transparency and Trust in Cardiovascular Disease Predictions

Bhargav Chopra, Srishti Sinha | Dr Govinda K | SCOPE

Introduction

Worldwide, about one-third of people are impacted by **Cardiovascular Diseases (CVDs)**. Although ML models such as XGBoost show potential, its **opaque decision-making** hampers clinical adoption. Hence, this project aims to increase the robustness and interpretability of the model by using **SHAP** as there are a lot of gaps in bridging the clinical usability and AI efficacy gap.

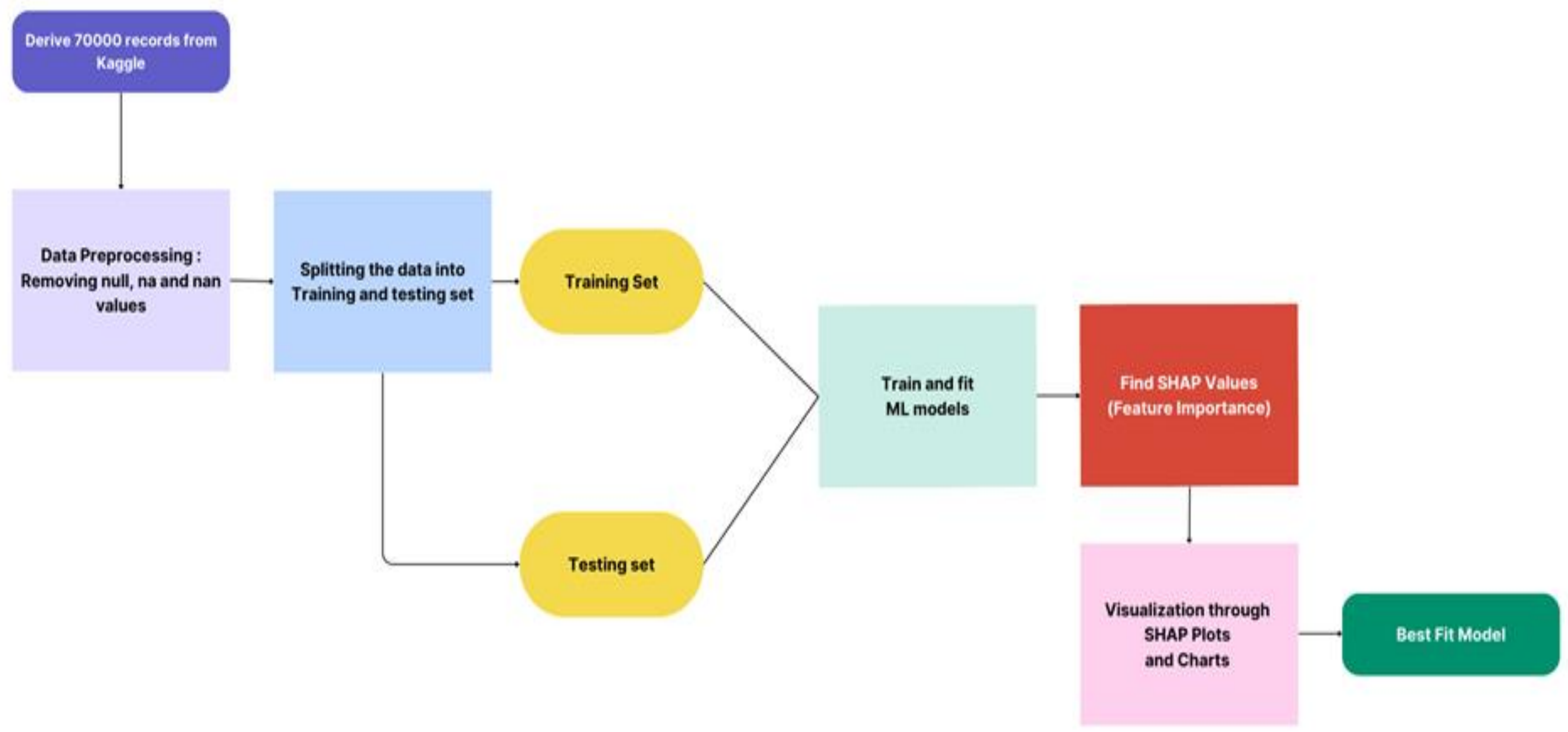
Motivation

The study tackles the pressing problem of effectively predicting cardiovascular illnesses (CVDs) which is a very fatal disease. Nevertheless, various ML models' opacity prevents them from being widely used in clinical settings.

SCOPE of the Project

The project intends to overcome the drawbacks of conventional methodologies by incorporating **AI and XAI techniques** to improve **CVD prediction** accuracy and transparency. It aims to provide healthcare professionals with more effective tools for the intervention and treatment of cardiovascular diseases by utilizing cutting-edge **ML models like XGBoost and LGBM** in conjunction with XAI techniques like SHAP. This helps to dispel the doubts surrounding the use of these models in clinical practice.

Methodology


Derive 70000 records from Kaggle
Data Preprocessing : Removing null, na and nan values
Splitting the data into Training and testing set
Training Set
Testing set
Train and fit ML models
Find SHAP Values (Feature Importance)
Visualization through SHAP Plots and Charts
Best Fit Model

Data Flow Diagram

DATA COLLECTION AND PREPROCESSING:

- Compiling information about cardiovascular diseases, considering variables like age, blood pressure, cholesterol, smoking, diabetes, etc.
- Handling missing values, outliers, ensuring data quality through preprocessing techniques and preparing the dataset for model training.
- Conducting exploratory data analysis to gain insights.

MODEL TRAINING AND EVALUATION:

- Training various machine learning models such as Random Forest, LGBM, and XGBoost.
- Evaluating models using metrics like accuracy, precision, and recall.

EXPLAINABLE AI (XAI) INTEGRATION:

- Applying SHAP to interpret model decisions.
- Analyzing feature importance to understand cardiovascular risk prediction.
- Enhancing trust in the model's decision-making process by providing transparent explanations for predictions.

VISUALIZATION AND PRESENTATION:

- Generate insights from plots and charts based on SHAP values such as summary plots, force plots, dependence plots, etc.

COMPARATIVE ANALYSIS AND MODEL SELECTION:

- Comparing models based on accuracy, interpretability, and suitability.
- Selecting the most suitable model(s) that balance accuracy and transparency.
- Considering computational efficiency and scalability for practical implementation in healthcare settings.

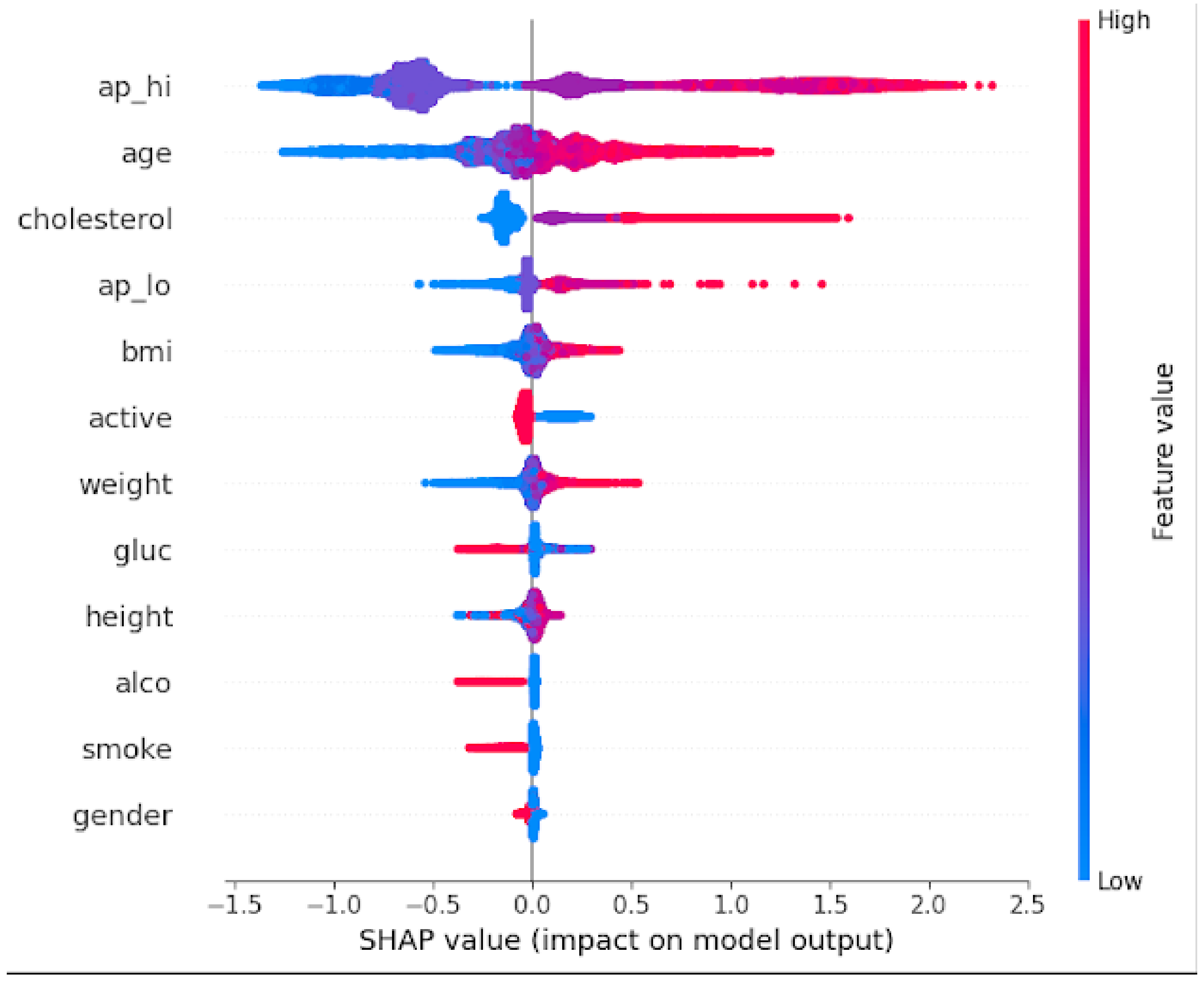
Results

Accuracies of Different models

Models	Accuracies
LGBM	*73.8601*
XGB	*73.8528*
Logistic Regression	*71.8883*
Random Forest	*71.4099*
Decision Tree	*64.1000*
ANN	*49.8500*
CatBoost	*30.3500*

In this project, we evaluated models for the prediction of cardiovascular disease (CVD) by combining explainable AI methods with various algorithms. With an accuracy of 71.8883%, logistic regression determined that weight was the most significant characteristic based on SHAP values. With accuracies of roughly 73.85% and 73.86%, respectively, XGBoost and LGBM significantly outperformed other models, but CatBoost, Decision Tree, and Neural networks only partially demonstrated potential.

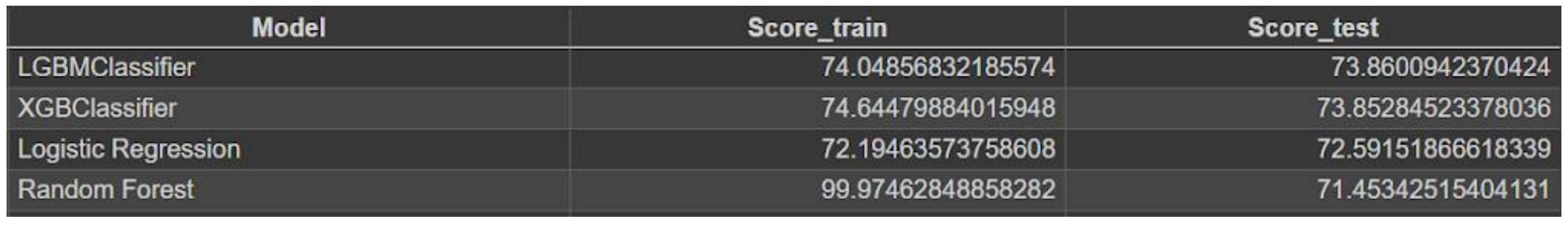
The significance of systolic pressure in predictions was underlined by SHAP analysis.



***Beeswarm Summary Plot for LGBM :** Here, we notice that ap_hi has the highest average SHAP value magnitude leading it to be the most prevalent feature in model prediction. The high values of ap_hi (systolic blood pressure) lead to greater impact in causing CVD and vice-versa. Infact, ap_hi is the most important factor in most of the models that we implemented, making it the key feature for CVD prediction.*

There were issues with the TreeExplainer in SHAP. An alternative would have been KernelExplainer, but it is quite costly to use computationally. Based on our evaluation, LGBM Classifier was found to be the best model, with low score disparity and superior performance. This study demonstrates how explainable AI can improve the prediction of CVD and increase confidence in projections.

Training and Testing scores of some predictive models



Model	Score_train	Score_test
LGBMClassifier	74.04856832185574	73.8600942370424
XGBClassifier	74.64479884015948	73.85284523378036
Logistic Regression	72.19463573758608	72.59151866618339
Random Forest	99.97462848858282	71.45342515404131

Conclusion

This study demonstrates how explainable AI may be used to improve model transparency and CVD prediction, which would improve patient care and healthcare decision-making. To evaluate the prediction of cardiovascular disease (CVD), we integrated SHAP with around six prediction models. XGBoost and LGBM achieved the highest training and testing scores as well as the least score difference, outperforming other models with the use of SHAP values for interpretability. Risk assessment and patient communication were made easier by visualizations. Upcoming advancements might concentrate on solving computational difficulties and also integrating Quantum Machine Learning (QML) to enhance predictive modelling in the healthcare industry and other domains.

References

1. Kırboğa KK, Küçüksille EU. Identifying Cardiovascular Disease Risk Factors in Adults with Explainable Artificial Intelligence. Anatol J Cardiol. 2023 Nov 1;27(11):657-663
2. Wesołowski S, Lemmon G, Hernandez EJ, Yandell M. An explainable artificial intelligence approach for predicting cardiovascular outcomes using electronic health records. PLOS Digit Health. 2022;1(1):e0000004.