CHAPTER 4
ANALYSIS AND DESIGN

4.1. Analysis

4.1.1. Handling Missing Values

Several handling missing values will be analyzed in this research: (1) Dropping missing data rows [6], (2) Replacing them with mode, (3) Replacing them with median. Each of these handling missing values will be evaluated to see which one has better performance. To analyze the performance of each handling missing value, each of them will be done in the preprocessing process and will be trained without feature selection. The result of the dataset preprocessed with dropping missing data rows will be calculated using model evaluation such as precision, recall, f-score, roc area, and accuracy. Three of the handling missing values calculated results will be compared to each other to analyze which handling missing values is better for predicting heart disease. The flow of handling missing values can be seen in Figure 4.1 below.

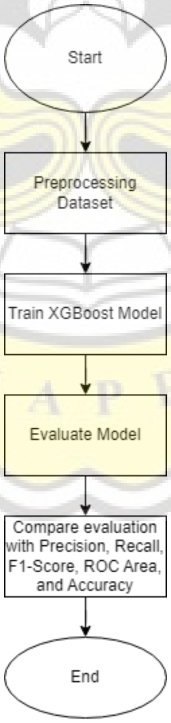
Start
Preprocessing Dataset
Train XGBoost Model
Evaluate Model
Compare evaluation with Precision, Recall, F1-Score, ROC Area, and Accuracy
End

Figure 4.1 Handling Missing Values Flowchart

4.1.2. Feature Selection

The Chi Square, Mutual Information, ANOVA, Forward Feature Selection, Backward Feature Selection, Recursive Feature Selection, and Feature Importance will be analyzed before comparing model with feature selection and without feature selection. The best result from handling missing values will be used in the preprocessing of the model. Each of the feature selection performance will be compared with precision, recall, f1-score, roc area, and accuracy.

4.1.3. Model Evaluation

Model evaluation metrics such as precision, recall, f1-score, roc area, and accuracy will be used to analyze the model performance. The best result from handling missing values will be used in the preprocessing of the model. The model without feature selection performance will be compared with AdaBoost algorithm that are used in previous research [12] to analyze whether XGBoost algorithm can achieve higher accuracy in the prediction.

4.2. Design

The overall design of this project can be seen in Figure 3.1 that has already been explained in the previous chapter. The first step in this research is to collect the dataset and do preprocessing on the dataset. The dataset can be obtained freely on the UCI Machine Learning Repository’s website [14]. The collected dataset will be processed by replacing the target value with only True (1) and False (0) [2] and by handling missing values. Each handling missing value will be analyzed and the best result will be used in the preprocessing method. The outliers will be removed using IQR method. Feature selection will also be analyzed to see which is better for the XGBoost. The data will be portioned using 10-fold cross validation. The XGBoost hyperparameter will be tuned using Bayesian Optimization with search space which can be seen in Table 4.1 below.

Table 4.1. XGBoost Hyperparameter Search Space

Hyperparameter	**Search Space**		
	Lower Bounds	***Upper Bounds***	***Optimal Result***
learning_rate	0.1	1	0.1
n_estimators	100	250	220
max_depth	1	15	1
min_child_weight	0	1	0.881
gamma	0	1	0.815
subsample	0.4	1	0.842
colsample_bytree	0.4	1	0.721

Table 4.2. AdaBoost Optimal Parameter

Hyperparameter	**Search Space**
	Optimal Result
learning_rate	1
n_estimators	50

The optimal result for the XGBoost hyperparameter achieved from the Bayesian Optimization and optimal result for the AdaBoost parameter [12] are shown in Table 4.1 and Table 4.2 above and will be used in this research. The best accuracy gained from the trained model with feature selected dataset will be used for the comparison with the trained model without feature selected dataset. The XGBoost algorithm will be compared with the AdaBoost algorithm that are used in previous research [12] with and without feature selection.