CHAPTER 4
ANALYSIS AND DESIGN

4.1. Analysis

At first, we collected the datasets from the internet as stated at chapter 3 first paragraph. These datasets contain general information such as age, sex type, to specific information like chest pain type, cholesterol, resting ECG, max heart rate, etc. The dataset also provides a column of information which determines whether a person has been diagnosed with a heart disease or not. There are 11 features (total 12 columns/features, include 1 heart disease column) that will be used to analyze which person has a heart disease or not. These 11 columns (from Age column to ST Slope column) will predict final result in heart disease column in "0" as normal heart or "1" as heart disease as shown in Table 4.1.

Age	Sex	Chest Pain Type	Resting Blood Pressure	Cholesterol	Fasting Blood Sugar	Resting ECG	Max Heart Rate	Exercise Angina	Old Peak	ST Slope	Heart Disease
40	M	ATA	140	289	0	Normal	172	N	0	Up	0
49	F	NAP	160	180	0	Normal	156	N	1	Flat	1
37	M	ATA	130	283	0	ST	98	N	0	Up	0
48	F	ASY	138	214	0	Normal	108	Y	1,5	Flat	1
54	M	NAP	150	195	0	Normal	122	N	0	Up	0
39	M	NAP	120	339	0	Normal	170	N	0	Up	0
45	F	ATA	130	237	0	Normal	170	N	0	Up	0
54	M	ATA	110	208	0	Normal	142	N	0	Up	0
37	M	ASY	140	207	0	Normal	130	Y	1,5	Flat	1
48	F	ATA	120	284	0	Normal	120	N	0	Up	0

Table 4.1. 10 Examples of Heart Datasets Downloaded from The Internet

By comparing the two ways the algorithm works on the heart attack patient datasets, it can be seen that both the training and testing results of these two algorithms have a good accuracy value. From experiments conducted to predict the potential for heart disease in a person, the two algorithms provide results with fairly high accuracy. The two algorithms were then carried out in various experimental scenarios by changing the amount of training data for the algorithms. By

changing the amount of training data to train the algorithm in predicting, different accuracy values were found for the Neural Network and Random Forest algorithms.

4.1.2. Neural Network

Neural Network is an algorithm that was created to imitate the working function of the human brain. The human brain is believed to consist of billions of tiny processing unit cells, called neurons, that work in parallel. One neuron cell will be connected to one other neuron through a neuron connection. Each node of a neuron takes input from a set of neurons. Which neuron then processes that input and passes the output to other nodes of neuron cells. The output is then collected by other neurons for further processing. In this process, the input can go through several processes, this is called the hidden layer.

A neural network is a representation of a computation using a network of simple processing elements called artificial neurons, or nodes. It is an example of parallel distributed processing. The network simulates the behavior of the human nervous-system and the data can be fed to the network algorithm by software. There are a number of different types of neural networks and each has its own strengths and weaknesses. Some of the most common are feedforward networks, recurrent networks and deep belief networks. Feedforward networks are the most common type of neural network as they have more mathematical properties which can be used to interpret the data. Recurrent networks are good at performing predictions on sequences of data but they cannot represent arbitrary sequences of data without additional techniques. Deep belief networks are a type of generative model that can be used to approximate probability density functions and perform unsupervised learning. There are two types of training methods that are used for neural networks, supervised learning and unsupervised learning. Supervised learning uses labeled examples to learn the behavior of the artificial neurons in the network whereas unsupervised learning learns the behavior of the network without any supervision.

4.1.2.1. How Neural Network Works?

The human brain is the inspiration behind the architecture of neural networks. Human brain cells, called neurons, form a complex network that is highly interconnected and sends electrical signals to each other to help humans process information. Similarly, an artificial neural network is made up of fabricated neurons that work together to solve a problem. Artificial neurons are

software modules, known as nodes, and artificial neural networks are software programs or algorithms that basically use a computer system to solve mathematical calculations.

The basic of Artificial Neural Network architecture are in three main layers:

a. Input Layer

   Information from certain sources, it can be from a dataset or many more enters the neural network from the input layer. Input nodes process the incoming information, analyze or classify it and pass it onto the next layer called Hidden Layer.

b. Hidden Layer

   From the previous process, the Hidden Layer gets input from the input layer or other hidden layers. An artificial neural network can have an enormous number of hidden layers. Each hidden layer evaluates the output of the previous layer, does further processing, and passes it on to the next layer. The next process could be passed on to another hidden layer or to the final process called the output layer.

c. Output Layer

   The output layer gives the final result of all data processing using the artificial neural network. It can have one or more outputs or nodes. For example, if we have a binary classification problem (yes/no), the output layer will have one output node, which will give return value either 1 or 0. However, if we have multiple classification problems, the output of the layer can include from many output nodes.

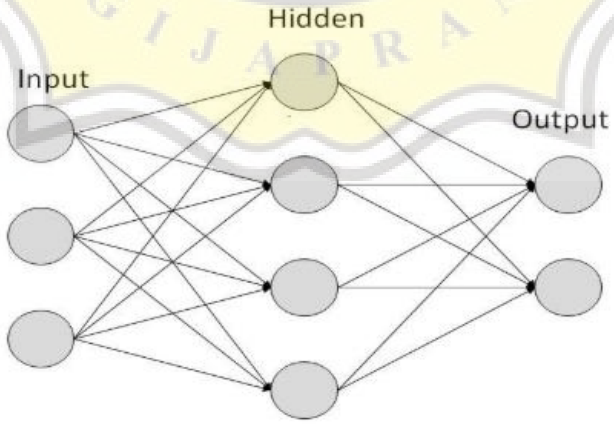

Figure 4.1 Scheme of How Neural Network Algorithm Works

4.1.3. Random Forest

Random Forest is an algorithm that consists of a collection of decision trees. This algorithm combines several decision trees which are combined and become one big model. This combination of several decision trees is then called a forest. This algorithm is usually used for training on large datasets. Random Forest depends on a random vector value with the same distribution in all trees, where each decision tree has the maximum depth. To explain this algorithm in a simpler way, imagine a dense forest in which thousands of trees are growing. Each tree in the forest is made up of many smaller trees which are connected to each other to form one giant tree. The objective of random forest is to find which of all these smaller trees will be the strongest tree and the trunk of which will become our big tree that is the forest.

The random forest algorithm makes multiple decisions or splits on each data set. It then learns to make a decision for the data based on the information received during the split process. The information obtained from the previous set of decisions is then used to rate how good a particular classification can be for that new data sample.

4.1.3.1. How Random Forest Works?

Random Forest develops multiple merged decision trees for more accurate predictions. The logic behind the random forest model is that multiple uncorrelated models (individual decision trees) work in groups much better than they do individually. When using Random Forest for classification, each tree will give a classification or "vote". The forest will choose the highest rank that has the majority of "votes". When using Random Forest for regression, the forest chooses the mean score of the outputs from all the trees.

The key here is that there is little (or no) correlation between the individual models, especially between the decision trees that make up the larger random forest model. Although individual decision trees can produce errors, the majority of the group will be right, thus putting the overall results in the right direction.

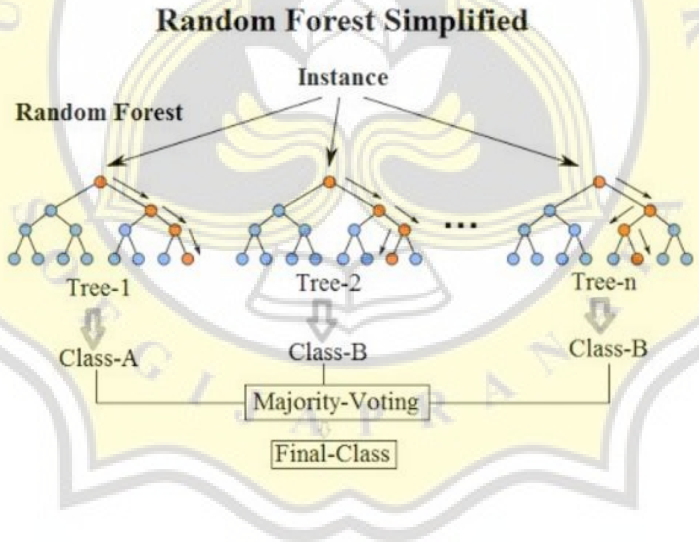

Figure 4.2 Scheme of How Random Forest Algorithm Works

4.2. Design

This is a workflow design for implementing an algorithm to predict heart disease in a person. This project uses Orange Data Mining to process patient datasets from hospitals in 5 regions. In figure 4.3 is an overview of the workflow used in this project

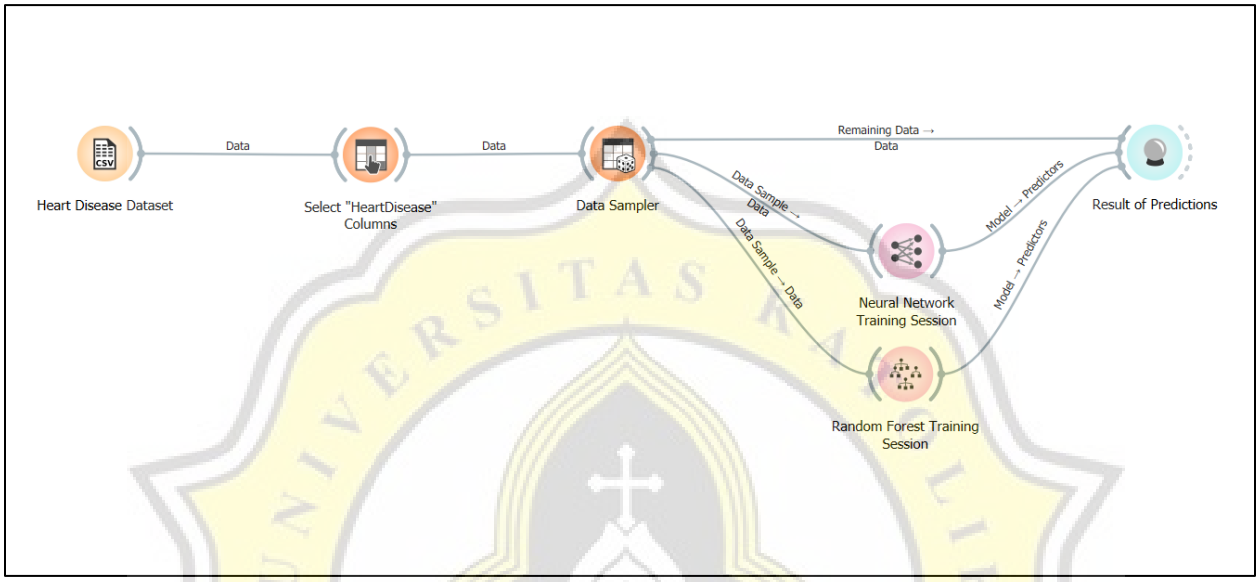

Figure 4.3 Workflow Of Heart Attack Prediction Using Neural Network and Random Forest on Orange Data Mining Application