Learning Vector Quantization

- **Learning Vector Quantization (or LVQ)** is a type of Artificial Neural Network which also inspired by biological models of neural systems. It is based on prototype supervised learning classification algorithm and trained its network through a competitive learning algorithm similar to Self Organizing
- The architecture of the Learning Vector Quantization with the number of classes in an input data and n number of input Map

How Learning Vector Quantization works?

- Let’s say that an input data of size (m, n) where m is the number of training examples and n is the number of features in each example and a label vector of size (m, 1). First, it initializes the weights of size (n, c) from the first c number of training samples with different labels and should be discarded from all training samples. Here, c is the number of classes. Then iterate over the remaining input data, for each training example, it updates the winning vector (weight vector with the shortest distance (e.g Euclidean distance) from the training example).
- The weight updation rule is given by:

```
if correctly_classified:
wij(new) = wij(old) + alpha(t) * (xi^k - wij(old))
else:
wij(new) = wij(old) - alpha(t) * (xik - wij(old))
```

- **Algorithm:**
- **Step 1:** Initialize reference vectors.
- from a given set of training vectors, take the first “n” number of clusters training vectors and use them as weight vectors, the remaining vectors can be used for training.
- Assign initial weights and classifications randomly
- **Step 2:** Calculate Euclidean distance for i=1 to n and j=1 to m,
- $D(j) = \Sigma\Sigma (x_i - W_{ij})^2$
- find winning unit index j, where D(j) is minimum
- **Step 3:** Update weights on the winning unit w_i using the following conditions:
- if T = J then $w_i(new) = w_i(old) + \alpha[x - w_i(old)]$
- if T ≠ J then $w_i(new) = w_i(old) - \alpha[x - w_i(old)]$
- **Step 4:** Check for the stopping condition if false repeat the above steps.

Decision Tree Classification

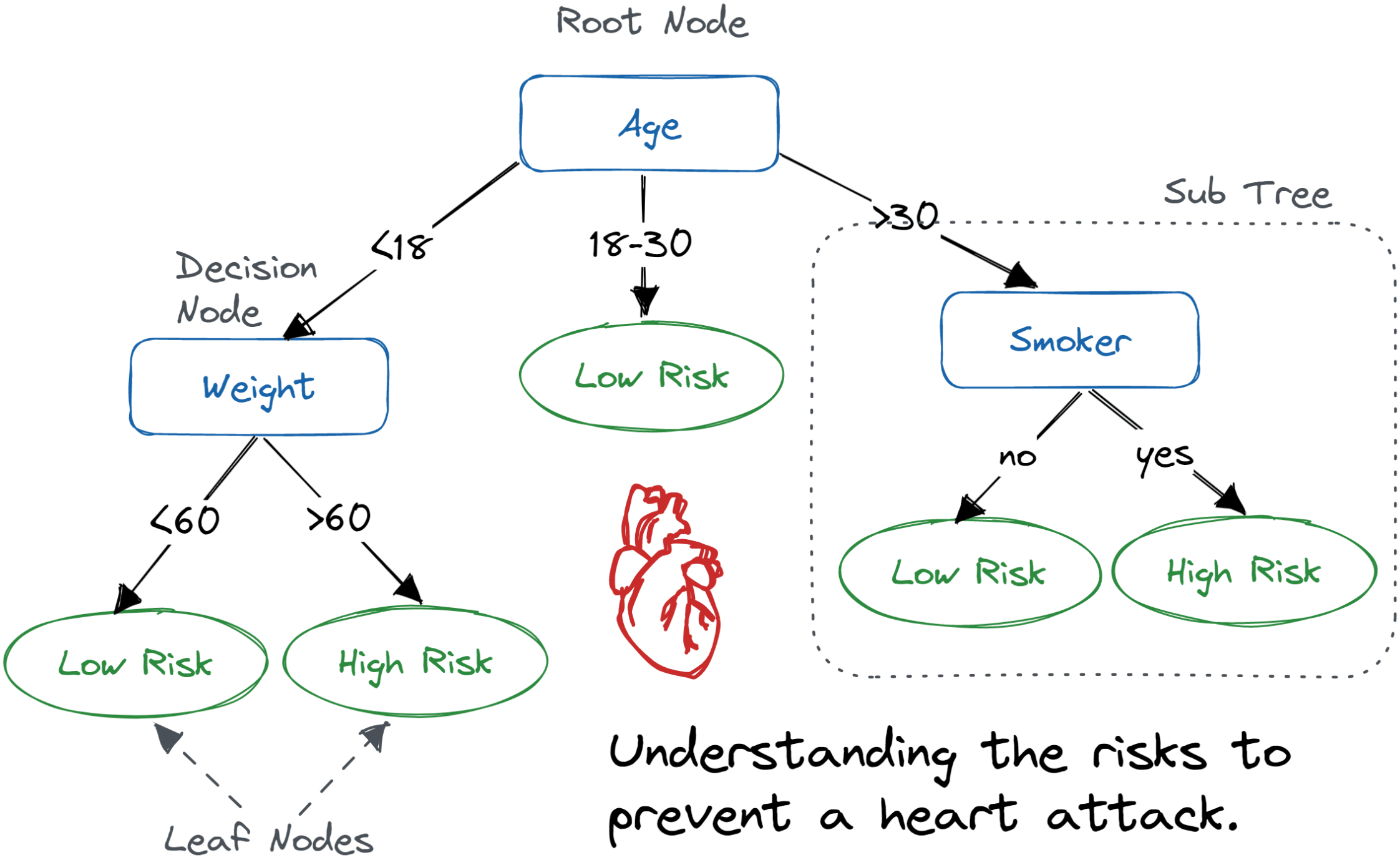
Classification is a two-step process; a

learning step and a

prediction step.

In the learning step, the model is developed based on given training data. In the prediction step, the model is used to predict the response to given data.

A Decision tree is one of the easiest and most popular classification algorithms used to **understand and interpret** data. It can be utilized for both classification and regression problems.


Root Node
Age
<18
18-30
>30
Decision Node
Weight
Low Risk
Sub Tree
Smoker
<60
>60
no
yes
Low Risk
High Risk
Low Risk
High Risk
Leaf Nodes
Understanding the risks to prevent a heart attack.

How Does the Decision Tree Algorithm Work?

1.Select the best attribute using Attribute Selection Measures (ASM) to split the records.

2.Make that attribute a decision node and breaks the dataset into smaller subsets.

3.Start tree building by repeating this process recursively for each child until one of the conditions will match:

1. All the tuples belong to the same attribute value.
2. There are no more remaining attributes.
3. There are no more instances.