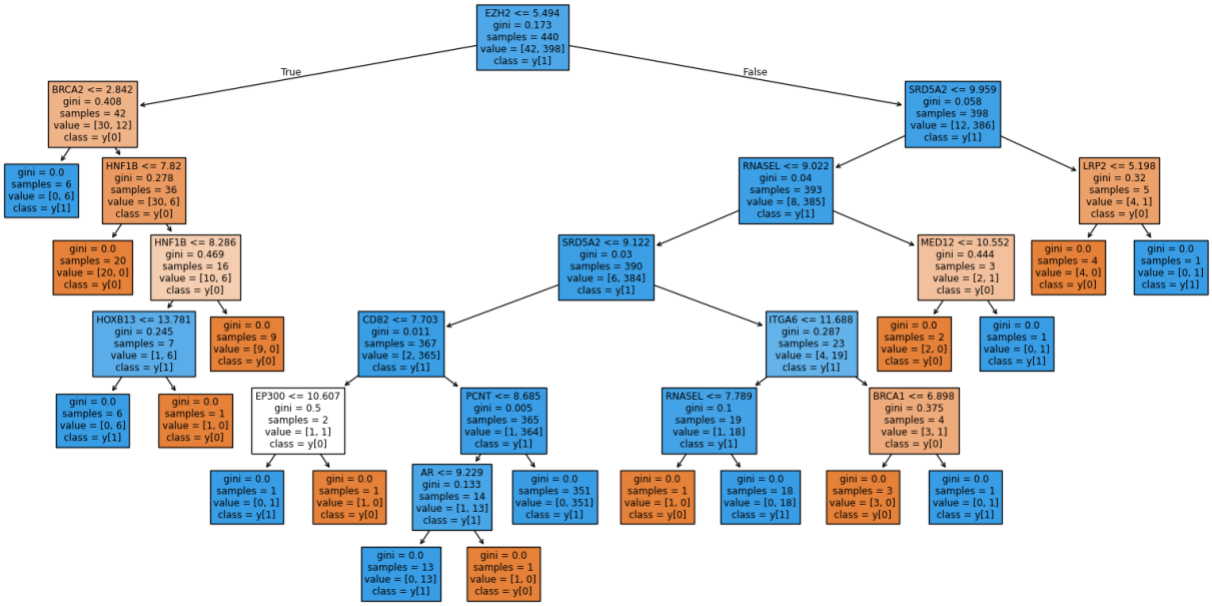
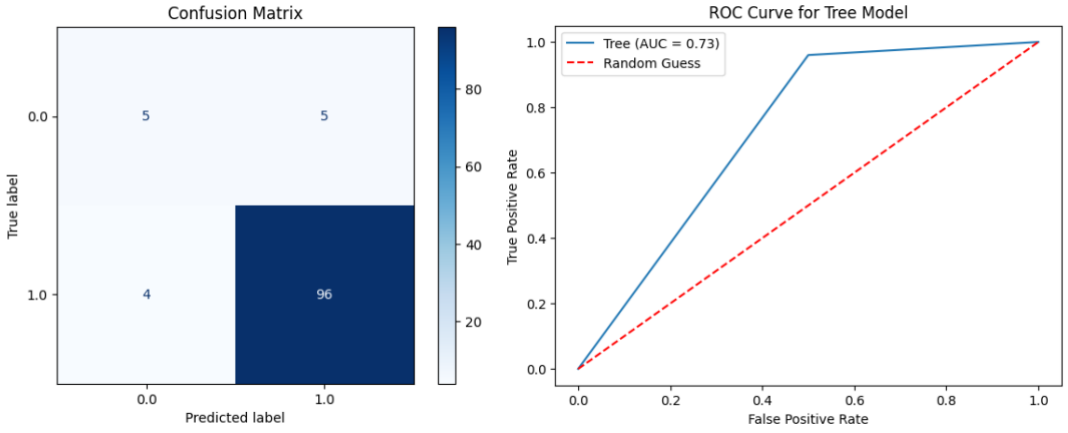
Part 1: Decision Tree

After applying a stratified split to the dataset, allocating 80% of the data for training and 20% for testing, we developed a decision tree classifier to model the data. The model was implemented using the DecisionTreeClassifier class from the sklearn.tree module. This decision tree was trained on the 80% training subset, and its performance was evaluated on the remaining 20% of the data.



The resulting decision tree achieved an accuracy of 91.82%. The confusion matrix and ROC curve based on the test data are pictured below.



Part 2: Nueral Network

We implemented a basic neural network to perform binary classification using a sigmoid activation function. We used the given dataset that contained three input features. The training set included three samples, while the testing set contained two unlabeled examples. We initialized the model with weights [0.3, 0.1, 0.3], set the learning rate to 0.01, and used a prediction threshold of 0.42. The number of training epochs was chosen randomly between 5 and 10.

After completing 8 epochs the weights are [0.31153311899147107, 0.11436181188732898, 0.2923212119157178]. The weights after each epoch are as pictured.

```
Weights after epoch 1: [0.3014384114368487, 0.10178629726772621, 0.29903791384322287]
Weights after epoch 2: [0.3028777637450858, 0.10357516119629698, 0.2980764620155537]
Weights after epoch 3: [0.30431804775852617, 0.10536658415332344, 0.2971156470214078]
Weights after epoch 4: [0.3057592542593427, 0.1071605584298978, 0.296155471361086]
Weights after epoch 5: [0.3072013739784043, 0.10895707624080332, 0.29519593753069745]
Weights after epoch 6: [0.3086443975956205, 0.11075612972473187, 0.294237048022082]
Weights after epoch 7: [0.31008831574029255, 0.11255771094450936, 0.2932788053227326]
Weights after epoch 8: [0.31153311899147107, 0.11436181188732898, 0.2923212119157178]
```

Both test sample 1 and test sample 2 produce an output of 1 when evaluated on the trained model.