# **Task\_4: Explain what you analyzed in the testing of the below code using pytest.**

## Output:Text Description automatically generated

Firstly, the titanic data is evaluated using decision tree and random forest models and below test cases have been created to test the model using pytest. By using the command pytest after changing the directory path to the path of the project folder, all the test cases that have been defined will be executed.

## Test\_decision\_tree\_1pre.py

Firstly, we test the ginni impurity and ginni gain of the decision tree and assert if the conditions given satisfy them. The third test case is to predict if the predicted model shape is the same as the shape of the labels. If the condition is false, then we assert that the ‘DecisionTree output should be the same as training labels.’. The fourth test case is to test if the output ranges from 0 to 1 inclusive. The fifth test case is to check if the model overfits. The sixth test case is to check if the accuracy and AUC ROC increase as the depth of the decision tree increases. The seventh test case is to check if the test data is leaked into the training data. If any of the test cases fail, we assert the error message.

Text

Description automatically generated with medium confidence

Text

Description automatically generatedText

Description automatically generated

## Test\_decision\_tree\_2post.py

We have two test cases here, one is to check if the outputs remain unchanged if we change certain inputs like name, ticket, embarked, etc., and the second test case is to check if the outputs remain unchanged if we change certain inputs like gender, Pclass, Fare, etc.

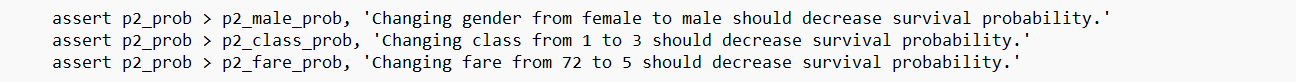
Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

## Test\_decision\_tree\_3eval.py

The first test case is to evaluate the decision tree model using the accuracy score and ROC AUC score. We check the accuracy scores with the value given, if the condition fails, we assert the error message. The second test case checks the training time at the 95th percentile, it should not be greater than 1 sec, if the condition fails, we assert the error message. The third test case checks the serving latency at the 99th percentile, it should not be greater than 0.004 sec, if the condition fails, we assert the error message.

Text

Description automatically generatedText

Description automatically generated

## Test\_random\_forest\_1pre.py

The first test case is to check if the model overfits by checking the labels and predicted outputs. The second test case is to check if the accuracy and AUC ROC increase as the number of trees increases. The third test case is to compare the accuracy scores of both the decision tree and random forest to check if the random forest model is better.

Text

Description automatically generated

Graphical user interface, text

Description automatically generated

## Test\_ random\_forest \_2post.py

We have two test cases here, using the random forest model, one is to check if the outputs remain unchanged if we change certain inputs like name, ticket, embarked, etc., and the second test case is to check if the outputs remain unchanged if we change certain inputs like gender, Pclass, Fare, etc.

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

A picture containing text

Description automatically generated

## Test\_ random\_forest \_3eval.py

The first test case is to evaluate the random forest model using the accuracy score and ROC AUC score. We check the accuracy scores with the value given, if the condition fails, we assert the error message. The second test case checks the training time at the 95th percentile, it should not be greater than 3 sec, if the condition fails, we assert the error message. The third test case checks the serving latency at the 99th percentile, it should not be greater than 0.018 sec, if the condition fails, we assert the error message.

Graphical user interface, text, application

Description automatically generatedText

Description automatically generated