**Test Plan**

Introduction

This document provides a test plan for the validation and testing of the Mushroom Classification Model as referenced in the Baseline Implementation and Iterative Development stages. This document presents a set of tests that the model must pass to be considered as complete and valid for classifying mushrooms as either edible or poisonous.

More specific information on what exactly will be testing during each Use Case is specified in the next document, the Test Cases document.

Use Cases

1. *Use Case ‘Create Workspace’ (Critical Importance)*
   1. The environment in which the program has been created on should be loaded without error.
2. *Use Case ‘Load Data’ (Critical Importance)*
   1. The environment should be able to load the dataset without errors
3. *Use Case ‘Modelling Phase’ (Critical Importance)*
   1. The data should have been cleaned using appropriate pre-processing techniques
   2. The data should be suitably split into train and test data
   3. The algorithm should return the best predictions possible according to the algorithm behaviour
   4. The overall system performance and accuracy revealed after modelling of the data
4. *Use Case ‘Exploratory Phase’ (Important)*
   1. The data should be able to be adequately explored via different data visualisation techniques such as tables and graphs to draw accurate conclusions from the tested data
5. *Use Case ‘Saving Data’ (Important)*
   1. The system should be able to be saved to the workspace, so that a saved version can be loaded into at a future date.
   2. The system should provide the ability to export the suitably trained and tested data into a data frame, and be outputted in a suitable format, e.g. a csv file
6. *Use Case ‘Adversarial Attacks’ (Not Important)*
   1. The system will be tested to ensure it can handle extraneous, unwanted data in the form of noise. Noise will be added to the dataset in the form of images and tested to identify whether the system can handle that noise so it is not included in Training or Testing data.

Testing Approach

The system is going to be tested via the traditional unit testing approach where all bugs will be tested. After this has been completed, integration testing will occur at the component and system levels. These tests will ensure that in the data used, there are no features missing, all anomalies have been detected before carrying out the modelling stage of the data, and there is a definite conclusion that can be drawn from the data from the post-modelling stage that can be easily understood and is unambiguous.

Testing Criteria

Each Use Case will be considered successful when each section of the test has run correctly without errors, and a relevant output has been delivered when needed (e.g. like in the Exploratory and Exporting phases). This is also true for the Training Phase, but for this phase to be deemed successful, the model must classify the mushrooms with a great amount of accuracy (70%+).

In the event of a failed Use Case, they will be reported in the test log along with a description of how it has failed, and testing will be halted whilst recommendations are made on how to fix that specific Use Case’s problem, and how it can be more successful overall. Recommendations might also be made if a Use Case has technically Succeeded the testing, but there are ways in which it can be improved.

Test Delivery

The test log will include the results of the test (Pass/Fail), as well as descriptions of failures, the date and time in which the test was conducted, and any recommendations.