Low Level Design

Mice Protein Expression

Contents

**1. Introduction……………………………………………………………………………………..3**

**1.1. What is Low-Level design document ?..........................................3**

**1.2. Scope………………………………………………………………………………………….3**

**2. Architecture………………………………………………………………………………………4**

**3. Architecture Description……………………………………………………………………5**

**3.1. Data Description………………………………………………………………………….5**

**3.2. Data Preprocessing……………………………………………………………………..5**

**3.3. Data Visualization……………………………………………………………………….5**

**3.4. Feature Selection………………………………………………………………………..5**

**3.5. Data Classification………………………………………………………………………5**

**3.6. Model Training……………………………………………………………………………6**

**3.7. Data Input…………………………………………………………………………………..6**

**3.8. Deployment………………………………………………………………………………..6**

**4. Unit Test Case……………………………………………………………………………………6**

1. Introduction

1.1 What is Low-Level design document ?

The goal of LLD or a low-level design document (LLDD) is to give the internal design of the actual program code for detection of trisomic mice. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

2. Architecture

Data Preprocessing

Data Description

Start

End

Output

Data Classification

Classifies Accordingly

Model Training

Data Analysis

Feature Selection

Data Input (sample)

Model Start

Data Visualization

3. Architecture Description

3.1 Data Description

The data set includes the expression levels of 77 proteins/protein changes that generated measurable signals in the cortex’s nuclear fraction. There are 72 mice in all, with 38 control mice and 34 trisomic mice (Down Syndrome). In the experiments, 15 measurements of each protein per sample/mouse were recorded. As a result, there are 38 × 15, or 570 measurements for control mice and 34 × 15, or 510 measurements for trisomic mice. There are 1080 measurements per protein in the dataset. Each measurement can be thought of as a separate sample/mouse.

3.2 Data Preprocessing

Data preprocessing steps we could use are imputation of missing values, label encoding, elimination of duplicate values, etc.

3.3 Data Visualization

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs and maps. Data visualization tools provide an accessible way to see and understand trends, outliers and patterns in data. In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.

3.4 Feature Selection

Feature selection (variable selection, attribute selection or variable subset selection) is a process where you automatically select those features in your data that contributes most to the prediction (output) it simply means that reducing the number of input variables which do not contribute towards the model. It enables the machine learning algorithms to train faster it also improves the accuracy of model if the right subset is choosen.

3.5 Data Classification

Here the data is classified as independent (X) and dependent (Y) feature as 80 and 20 percent to train and test the model respectively.

3.6 Model Training

The model training is process in which the model is trained using training data by some of algorithms.

3.7 Data Input

Input data of particular mice/sample is given the model system processes the input data as trained and classifies it as normal or trisomic.

3.8 Deployment

We will be deploying the model to AWS.

4. Unit Test Cases

|  |  |  |
| --- | --- | --- |
| Test Case Description | Pre-Requisite | Expected Result |
| Verify whether the system is user friendly | 1. The system should be user friendly | The system should be user friendly |
| Verify whether the Application URL is accessible to the user | 1. Application URL should be defined | Application URL should be accessible to the user |
| Verify whether the Application loads completely for the user when the URL is accessed | 1. Application URL is accessible  2. Application is deployed | The Application should load completely for the user when the URL is accessed |
| Verify whether the user is able to sign up in the Application | 1. Application is accessible | The user should be able to sign up in the application |
| Verify whether user is able to successfully login to the Application | 1. Application is accessible  2. User is signed up to the application | User should be able to successfully login to the application |