

Prediction of LC50 value using

Quantitative structure activity relationship models (QSAR models)

High Level Design

Domain: Machine Learning

Author : Debasish Mohanty

Date: 01.06.2023

**Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Date issued | Version | Description | Author |
| June 1st, 2023 | 1.1 | First Draft | Debasish Mohanty |

CONTENTS

[Abstract 3](#_Toc110595993)

1. [Introduction 4](#_Toc110595994)

1.1 [What is High-Level Design Document? 4](#_Toc110595995)

1.2 [Scope 4](#_Toc110595996)

2. [General Description 5](#_Toc110595997)

2.1 [Definitions 5](#_Toc110595998)

2.2 [Product Description 5](#_Toc110595999)

2.3 [Problem Statement 5](#_Toc110596000)

2.4 [Proposed solution 6](#_Toc110596001)

2.5 [Data Source 6](#_Toc110596003)

2.6 [Tools used 6](#_Toc110596004)

2.7 [Hardware Requirements 8](#_Toc110596005)

2.8 [Constraints 8](#_Toc110596006)

2.9 [Assumptions 8](#_Toc110596007)

3. [Design Details 9](#_Toc110596008)

3.1 [Process Flow 9](#_Toc110596009)

3.2 [Event Log 1](#_Toc110596010)0

3.3 [Error Handling 1](#_Toc110596011)0

4. [Performance 1](#_Toc110596012)1

4.1 [Reusability 1](#_Toc110596013)1

4.2 [Application Compatibility 1](#_Toc110596014)1

4.3 [Resource Utilization 1](#_Toc110596015)1

4.4 [Deployment .....................................................................................................................................1](#_Toc110596015)1

5. [Dashboards 1](#_Toc110596016)2

5.1 [KPIs (Key Performance Indicators) 1](#_Toc110596017)2

6. [Conclusion 1](#_Toc110596018)3

7. [References 1](#_Toc110596018)4

# Abstract

The prediction of LC50 value is an essential part of ecotoxicology research. This study explores the use of machine learning models for the prediction of LC50 value. The QSAR models leverage a wealth of scientific vocabulary and computational tools to construct a predictive model that relates the molecular structure of a compound to its LC50 value. The predictive power of the QSAR models is driven by a suite of physicochemical descriptors, which describe the molecular structure of the compound, and machine learning algorithms, which learn to predict the LC50 value based on the physicochemical descriptors. The study includes the selection and preparation of datasets, feature engineering and selection, model training and validation, and the analysis and interpretation of the results. The performance of the machine learning models is evaluated based on metrics such as correlation coefficients, root mean square errors, and mean absolute errors. The results demonstrate the potential of machine learning models as a valuable tool for predicting LC50 values and suggest further opportunities for improving model performance through the selection of appropriate descriptors and algorithms.

# Introduction:

## 1.1 What is High-Level Design Document?

The goal of this HLD or a high-level design document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

* Present all of design aspects and define them in detail
* Describe all user interfaces being implemented
* Describe the hardware and software interfaces
* Describe the performance requirements
* Include design features and architecture of the project
* List and describe the non-functional attributes such as security, reliability, maintainability, portability, reusability, application compatibility. resource utilization, serviceability

## 1.2 Scope:

The HLD documentation presents the structure of the system, such as database architecture, application architecture (Layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly technical terms which should be understandable to the administrators of the system.

# General Description

## 2.1 Definitions:

|  |  |
| --- | --- |
| **Term** | **Description** |
| LC50 | exposure concentration of a toxic substance lethal to half of the test animals |
| QSAR | Quantitative Structure Activity Relationship Models |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## 

## 2.2 Product Description:

The product of this project is fully automated end-to-end Machine Learning model capable of predicting the LC50 value. This refers to the concentration of a substance that can cause the death of 50% of a test batch of fish within a 96-hour duration, using 6 predetermined molecular descriptors.

## 2.3 Problem Statement :

The problem addressed in this project is to develop robust and accurate QSAR models for predicting the LC50 value, a measure of the concentration of a substance that causes 50% mortality in test organisms, of potentially hazardous chemical compounds.

The project aims to leverage the power of machine learning algorithms and chemical informatics tools to construct predictive models that can help in the identification of potential risks associated with chemicals, and thereby support the design of safer chemicals and products.

## 2.4 Proposed Solution :

The proposed solution for the problem involves the development and evaluation of QSAR models based on molecular descriptors to predict LC50 values. The models will be trained on a dataset of compounds with known LC50 values and molecular descriptors. The performance of the models will be evaluated using statistical measures such as correlation coefficient, root mean square error, and coefficient of determination. The resulting models can be used to predict the toxicity of new compounds, which can be useful in environmental risk assessment, and chemical safety evaluation.

## 2.5 Data Source1:

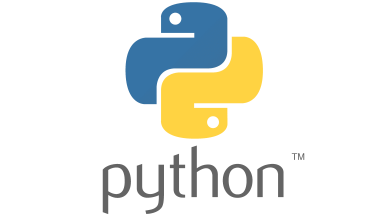
This study utilizes in vivo test data on fish for numerous chemical substances obtained from the ECOTOX database of the US Environmental Protection Agency.

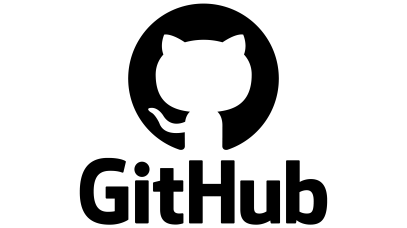
* CIC0
* SM1\_Dz(Z)
* GATS1i
* TNdsCH
* NdssC
* MLOGP

Above parameters are the required molecular descriptors to feed into the model.

## 2.6 Tools Used:

The process of developing the model involves using Python programming language and frameworks like NumPy, Pandas, Scikit-learn, and Optuna. React JavaScript is used for frontend development, and FastAPI is utilized for backend API development. Additionally, other tools are used to deploy the model.

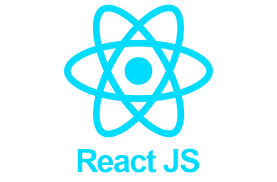












* For visualization tasks, matplotlib, seaborn were used.
* NumPy and Pandas were used to clean and interpret data.
* Scikit-learn was used for data preprocessing and model selection.
* Optuna was used for efficient hyper-parameter optimization
* Kaggle was used for gpu cloud training of the models.
* ReactJS and FastAPI were used for building the web application.
* GitHub was used as version control system
* VSCode and NeoVim were used as primary IDE.
* Amazon EC2 instance is used for model deployment.

## 2.7 Hardware Requirements :

* Windows Server, Linux, or any operating system that can run as a webserver, capable of delivering HTML5 content.
* Minimum 1.10 GHz processor or equivalent.
* Between 1-2 GB of free storage
* Minimum 512 MB of RAM
* 3 GB of hard-disk space

## 2.8 Constraints :

The front-end must be user friendly and should not need any one to have any prior knowledge in order to use it.

## 2.9 Assumptions :

The primary goal of this project is to apply the previously described use case to new datasets received via the user interface. It is assumed that all components of the project are able to function seamlessly as intended by the designer and that the training data for the model is as accurate as possible.

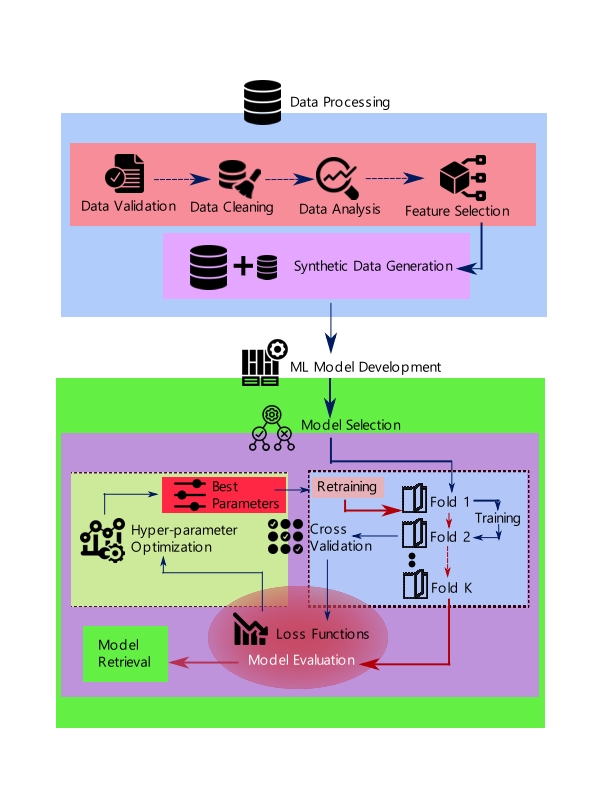
# 

# Design Details :

## 3.1 Process Flow :

We utilized various machine learning models to complete the task, and the model development workflow is illustrated below.

**Model Development Work-Flow:**



## 3.2 Event Log :

The system should log every event so that the user will know what process is running internally. Initial step-by-step description:

1. The system identifies at what level logging is required
2. The system should be able to log each and every system flow
3. Developer can choose logging method.
4. System should not hang even after so many loggings. Logging just because we can easily debug issues, so logging is mandatory to do.

The event logging for the machine learning development process is facilitated by the implementation of a logging library, which is utilized consistently throughout the codebase for each individual model. The event logs, encompassing valuable information regarding the execution and progress of the models, are securely stored within the code repository, ensuring their availability for future reference and analysis.

## 3.3 Error Handling :

Errors should be encountered, an explanation will be displayed as to what went wrong ? An error will be defined as anything that falls outside the normal intended usage.

# Performance:

This tool could be used to predict whether the toxicity condition is adverse, suitable or managable for fish environment by providing accurate regression of LC50 value . It can be used by various governmental/ non-governmental/ private agencies then it is supposed to be as accurate as possible. So that it doesn’t mislead authorities. Also, model retraining is very important to further enhance its performance.

## 4.1 Reusability :

The code written and the components used should have the ability to be reused with no problems.

## 4.2 Application Compatibility :

The different components for this project will be using Python as an interface between them, each component will have its own task to perform, and it is the job of Python to ensure proper transfer of information.

## 4.3 Resource Utilization :

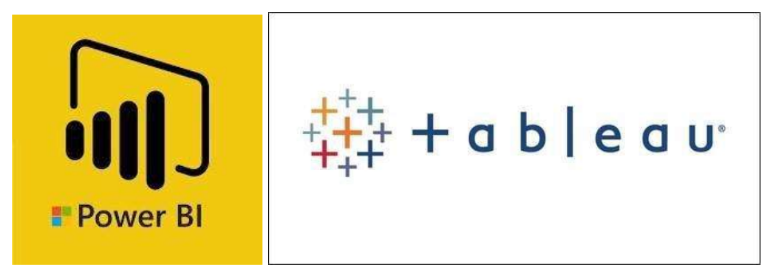
When any task is performed, it will likely use all the processing power available to it until finished.

## 4.4 Deployment :



## 5. Dashboard:

Immersive dashboards shall be implemented to visually present and signify the LC50 value alongside pertinent molecular indicators, encapsulating an expansive range of parameters. These comprehensive dashboards serve as an indispensable resource, alerting and highlighting critical thresholds that, if left unattended, may precipitate species extinctions within specific regions, potentially leading to unprecedented ramifications on biodiversity.



Upon the initiation of chemical data acquisition for a user, the system will seamlessly integrate dynamic dashboards to present temporal charts showcasing the evolving progress across diverse chemical factors.

## 5.1 KPIs (Key Performance Indicators):

1. Latency or response time for model prediction
2. Monitoring of LC50 values for toxicity detection
3. Insights for balancing molecular descriptors in required conditions
4. Information retrieval to corresponding autority
5. Integrating with disaster management in fishery enviroment
6. The processing power required to run

# Conclusion:

The devised model will possess the capacity to forecast the LC50 value based on six molecular descriptors, thereby delineating the magnitude of toxicity within an ecosystem. By employing Quantitative Structure-Activity Relationship (QSAR) models, this capability is harnessed through the utilization of meticulously formatted and pristine data. The applications of this project encompass the determination of environmental conditions and the monitoring of fluctuations in the LC50 value, as per the specified requirements.

References:

1.https://www.epa.gov/chemical-research/ecotoxicology-ecotox-knowledgebase