Low Level Design(LLD)

Insurance Premium Prediction

Written By	Apurv Bhusari
Document Version	1.2
Last Revised Date	17/02/2023

Document Version Control

Version	Date	Author	Comments
1.0	1/01/2023	Apurv Bhusari	Abstract Introduction Architecture
1.1	15/01/2023	Apurv Bhusari	Architectural Design
1.2	17/02/2023	Apurv Bhusari	Deployment Unit Test Cases

Contents

1. Introduction

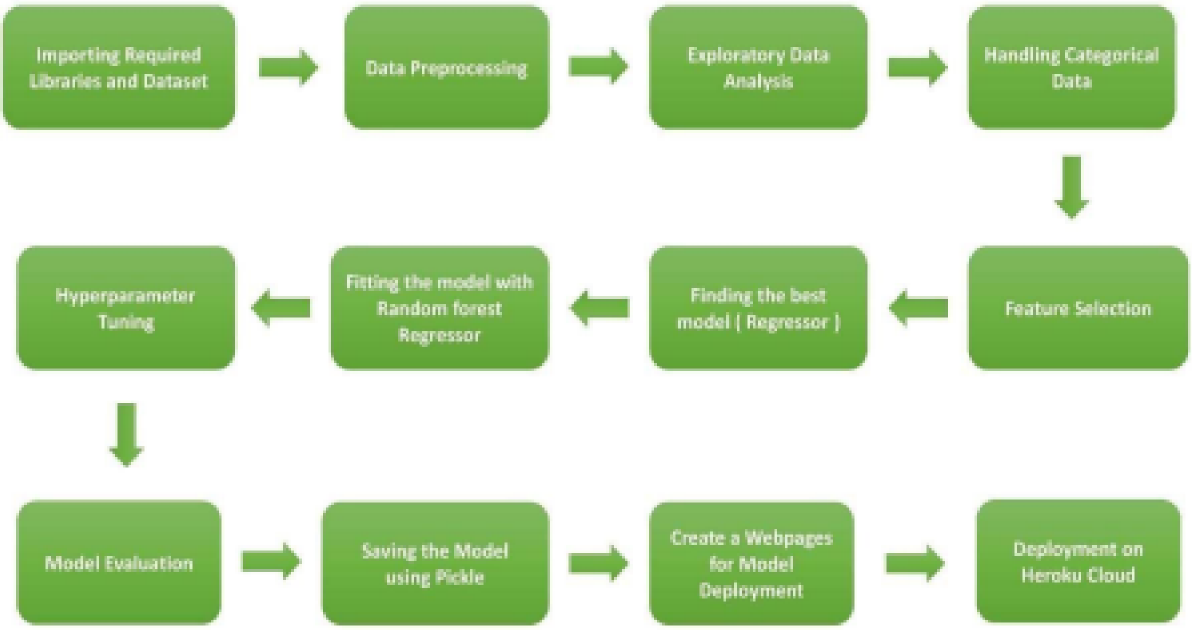
1.1. What is Low-Level design document?

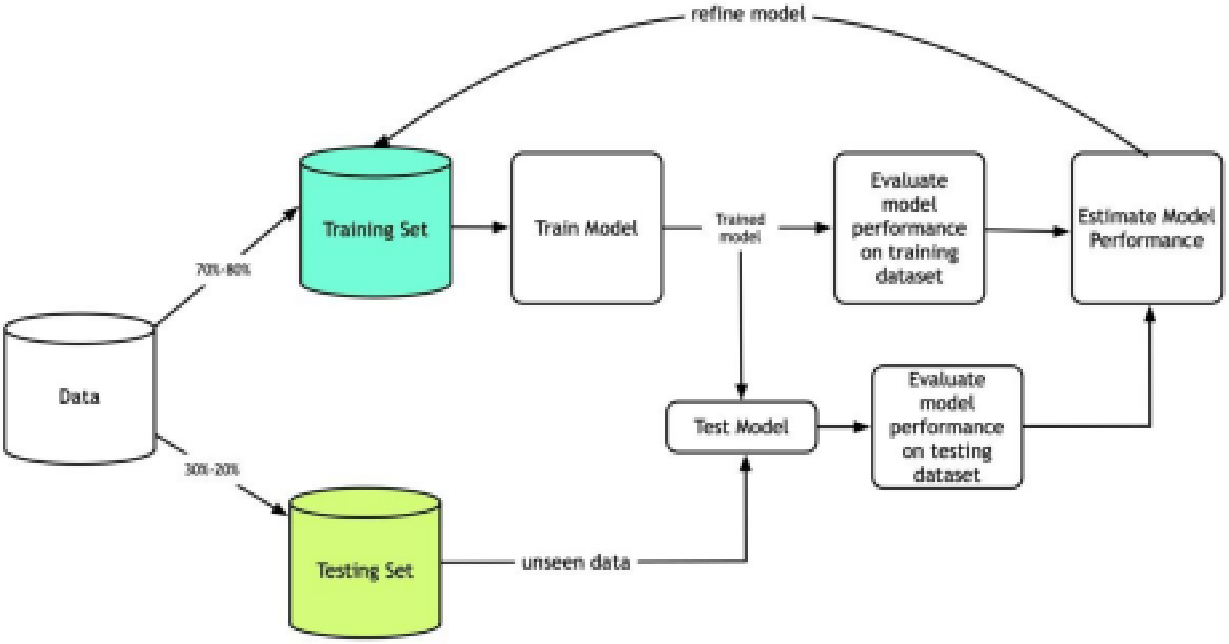
The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Food Recommendation System. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can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

2. Architecture


Importing Required Libraries and Dataset
Data Preprocessing
Exploratory Data Analysis
Handling Categorical Data
Feature Selection
Finding the best model (Regressor)
Fitting the model with Random forest Regressor
Hyperparameter Tuning
Model Evaluation
Saving the Model using Pickle
Create a Webpages for Model Deployment
Deployment on Heroku Cloud


refine model
Training Set
Train Model
Trained model
Evaluate model performance on training dataset
Estimate Model Performance
70%-80%
Data
30%-20%
Test Model
Evaluate model performance on testing dataset
Testing Set
unseen data

3. Architecture Description

3.1. Data Description

This dataset contains 1000 records along with 6 columns like age, sex, gender, region, bmi etc. This dataset is available in .zip format.

3.2. Data Ingestion

In the Ingestion Process, we will convert our original dataset which is in Zip format to csv format. After that we will split them into train and test dataset.

3.3. Data Validation

In Data validation steps we could use Null value handling, outlier handling, Imbalanced data set handling, Handling columns with standard deviation zero or below a threshold, etc.

3.4 Data Transformation

In this step we will transform out data. We will use standard scaler for numeric data and we will convert categorical data into numeric data using one-hot encoding technique so that machine can understand it.

3.5 Model Training

Here we will build the Machine Learning model using all regression algorithms.

3.6 Model Evaluation

Here model evaluation will be done on the model which we got in the model building stage. We can define base accuracy of the model and if model accuracy is higher then base accuracy, then only our model will accept otherwise it will be rejected.

3.7 Model Pusher and Deployment

Here model will be deployed to Heroku cloud platform

Unit Test Cases

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 user is able to see input fields on application	1. Application is accessible 2. User is logged into the application	User should be able to see input fields on application
Verify whether user is able to edit all input fields	1. Application is accessible 2. User is logged in to the application	User should be able to edit all input fields
Verify whether user gets Submit button to submit the inputs	1. Application is accessible 2. User is logged in to the application	User should get Submit button to submit the inputs
Verify whether user is getting predicted results on clicking submit	1. Application is accessible 2. User is logged in to the application	User should be presented with predicted results on clicking submit