

HIGH LEVEL DOCUMENT

Insurance Premium Prediction

| | |
|--------------------------|------------------------|
| Written By | Mohammad Sohail Parvez |
| Document Version | 0.1 |
| Last Revised Date | |

Content

Abstract

1. Introduction

1.1 Why this High Level Design Document?

2. General Description

2.1 Product Perspective

2.2 Problem Statement

2.3 Proposed Solution

2.4 Technical Requirements

2.5 Data Requirements

2.6 Tools Used

2.7 Constraints

3. Design Details

3.1 Process Flow

3.2 Deployment Process

3.3 Event Log

3.4 Error Handling

4. Performance

4.1 Re-usability

4.2 Application Compatibility

4.3 Deployment

5. Conclusion

Abstract

For this project, we examine the “Insurance Premium Prediction” dataset available at the kaggle. Our goal is to predict the premium.

1. Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the important details about this project. Through this HLD Document, I'm going to describe every small and big thing about this project.

2.General Description

2.1 Product Perspective

The ML project using Supervised learning(regression) .

2.2 Problem statement

The goal of this project is to give people an estimate of how much they need based on their individual health situation. After that, customers can work with any health insurance carrier and its plans and perks while keeping the projected cost from our study in mind. This can assist a person in concentrating on the health side of an insurance policy rather than the ineffective part.

2.3 Proposed Solution

The dataset does not contain any more preprocessing. We have to encode the data to numerical and pass it to the model.

2.4 Technical Requirements

In this project the requirements to get income classify various platforms. For that, in this project we are going to use different technologies. Here are some requirements for this project.

- Model should be exposed through API or User Interface, so that anyone can test model.
- Model should be deployed on the cloud..

2.5 Data Requirements

The dataset is downloaded from Kaggle. The dataset contains 1300+ records for training The train dataset contain three columns

2.6 Tools Used

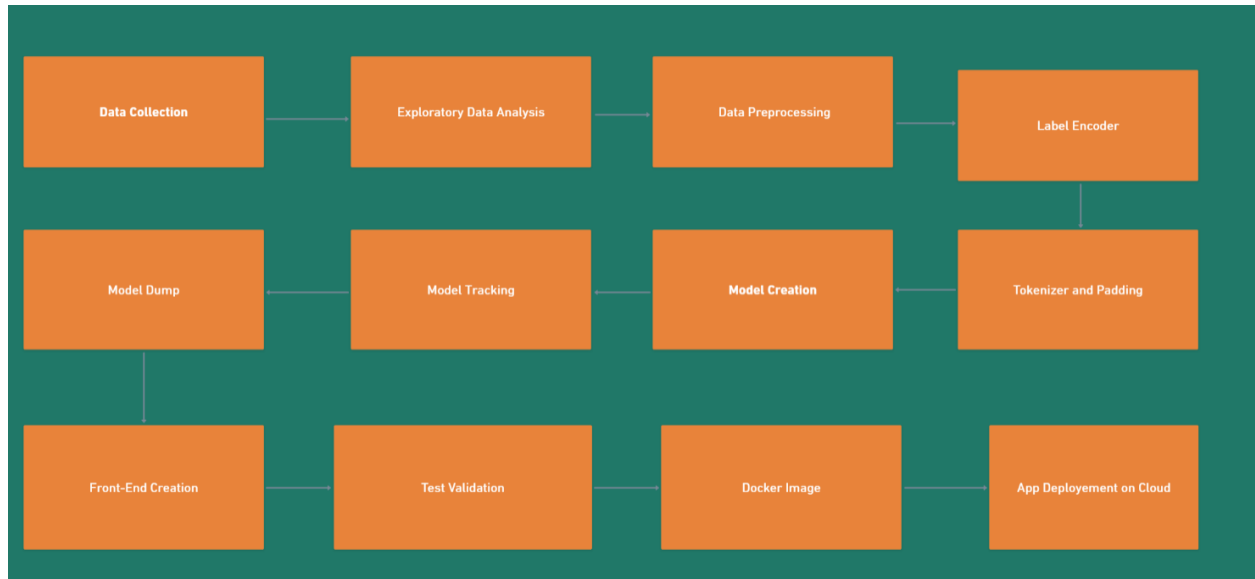


2.7 Constraints

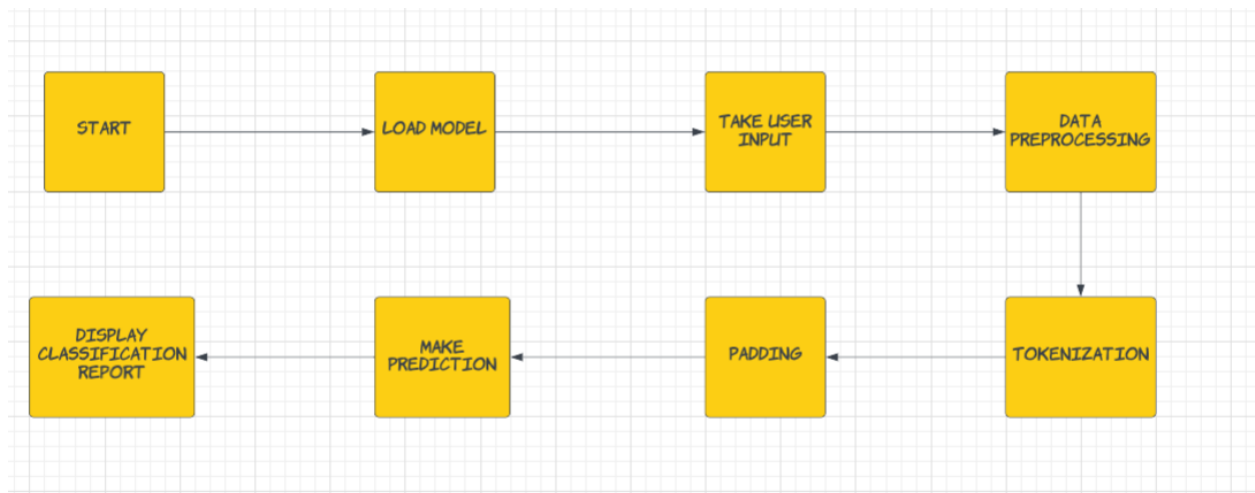
The Insurance premium prediction system must be user friendly, error free and users should not be required to know any of the back-end working.

3 Design Details

3.1 Process Flow



3.2 Deployment Process



3.3 Event Log

In this project we are logging every process so that the user will know what process is running internally.

Step-By-Step Description:

- In this project we defined logging for every function.
- By logging every function in the Exploratory Data Analysis.
- Then logs all the data preprocessing functions and code.
- Logs every model algorithm to our data.

3.4 Error Handling

The project is designed in such a way that, at any step if error occurs then our application should not terminate rather it should catch the error and display that error with proper explanation as to what went wrong during process flow.

4. Performance

Solution of Machine Learning is used to classify in advance, so it should be as accurate as possible so that it should give as much as possible accurate classification. That's why before building this model we followed the complete process of Machine Learning. Here are summary of complete process:

1. First we cleaned our dataset properly by removing all null value and duplicate values present in the dataset
2. Then we explored our dataset using visualization using EDA preprocessing. After EDA we prepared a XGBRegressor model for model prediction and saved it.
3. We build an end-to-end machine learning pipeline for re-training and for new dataset.
4. We build two machine learning pipeline mainly for batch prediction and other is for normal data prediction.
5. For every input and output we had generated outputs using artifact entity and for input we have used config entity.

4.1 Re-usability

The programming is done in such a way for this project that it should be reusable. So that anyone can add and contribute without facing any problems.

4.2 Application Compatibility

The difference module of this project is using Python as an interface between them. Each module has its own job to perform and it is the job of the Python to ensure the proper transfer of information.

4.3 Deployment

We have deployed this on cloud and also dockerized this.

5. Conclusion

The Machine Learning model will predict the premium