HLD (High Level Design)

Adult Census Income Prediction

(With complete CI/CD pipelines)

Revision Number: 2.0
Last date of revision: 04/03/2023

Document Version Control

Date Issued	Version	Description	Author
15/02/2023	1	Initial HLD — V1.0	Rushikesh Chalake
04/03/2023	2	Final HLD — V1.1	Rushikesh Chalake Rajendra Jadhav

Contents

Document Version Control. 2
Abstract. 4
1 Introduction 5
1.1 Why this High-Level Design Document? 5
1.2 Scope. 5
1.3 Definitions 5
2 General Description. 6
2.1 Product Perspective 6
2.2 Problem statement 6
2.3 PROPOSED SOLUTION 6
2.4 FURTHER IMPROVEMENTS 6
2.5 Data Requirements 6
2.6 Tools used. 7
2.7 Hardware requirements 8
2.8 Constraints 8
2.9 Assumptions 8
3 Design Details 9
3.1 Process Flow. 9
3.1.1 Components of ML pipelines 9
3.1.2 Coding flow for building project structure 9
3.1.3 Complete pipeline flow 10
3.2 Event log 10
3.3 Error Handling 10
4 Performance. 11
4.1 Reusability 11
4.2 Application Compatibility 11
4.3 Resource Utilization 11
4.4 Deployment. 11
5 Conclusion 11

Abstract

The Adult Census Income Prediction project involves building a machine learning model to predict an individual's income level based on their demographic and socioeconomic characteristics. The project uses the Adult Census Income dataset, which contains information about individuals' age, education, occupation, marital status, and more. The project aims to build a model that can accurately predict an individual's income level, which can be used for various purposes such as targeted marketing, policy-making, and social research. The project also involves creating an end-to-end solution, including data preprocessing, feature engineering, model selection, and model training. The performance of the model will be evaluated on a test set, and the best model will be deployed in a production environment. The project will also incorporate continuous integration and continuous delivery (CI/CD) pipelines to automate the building, testing, and deployment of the machine learning model. The end result is a comprehensive solution that can accurately predict an individual's income level and can be quickly and reliably deployed in a production environment.

1 Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) 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 the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes

  like: o Security
  - o Reliability
  - o Maintainability
  - o Portability
  - o Reusability
  - o Application compatibility
  - o Resource utilization
  - o Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the 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.

1.3 Definitions

Term	*Description*
CI/CD	Continuous Integration / Continuous Deployment
ASIP	Adult Census Income Prediction
IDE	Integrated Development Environment

2 General Description

2.1 Product Perspective

The ultimate goal is to build a model that can accurately classify individuals based on their income level, with the aim of aiding decision-making processes in various industries and sectors.

2.2 Problem statement

The problem we are trying to solve is to classify individuals based on their income level, which can be used for a variety of purposes such as targeted marketing, policy-making, and social research. Therefore, the objective of this project is to develop a machine learning model that can accurately predict an individual's income level based on their demographic and socioeconomic characteristics. The Goal is to predict whether a person has an income of more than 50K a year or not. This is basically a binary classification problem where a person is classified into the >50K group or <=50K group.

2.3 PROPOSED SOLUTION

Using all the standard techniques used in the life-cycle of a Data Science project starting from Data Exploration, Data Cleaning, Feature Engineering, Model Selection, Model Building and Model Testing and also building a frontend where a user can fill their information in the form input and get the output instantly.

2.4 Data Requirements

Data requirement completely depend on our problem statement.

- We need data that is balanced and must have details about person's demographic and socioeconomic characteristics.
- Data can be ingested from the Cassandra database.

2.5 FURTHER IMPROVEMENTS:

- The ACIP can be easily embedded inside any website or an application and can be used to find out whether a person earns more than $50K annually or not and can be used by various governmental / non- governmental / private agencies around the world.
- Accurately predicting an individual's income level can also help organizations to identify potential customers for their products or services or identify individuals who may be in need of financial assistance.

2.6 Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, are used to build the whole model.

- **Visual Studio Code** is used as IDE.
- For visualization of the plots, **Matplotlib, Seaborn** are used.
- **Railway** is used for deployment of the model.
- **Apache Cassandra** is used to retrieve, insert, delete, and update the database.
- Front end development is done using **HTML/CSS**
- Python **Flask** is used for backend development.
- **GitHub** is used as version control system.
- **Scikit-learn** was used to cross validate and compare different models.
- **GradientBoostingClassifier** is used to build the final model.

2.7 Hardware Requirements

- Windows Server, Linux, or any operating system that can run as a webserver.
- Minimum 1.10 GHz processor or equivalent.
- Between 1-2 GB of free storage
- Minimum 512 MB of RAM

2.8 Constraints

The Adult Census Income Prediction system must be user friendly, as automated as possible and users should not be required to know any of the workings.

2.9 Assumptions

The main objective of the project is to implement the use cases as previously mentioned (2.2 Problem Statement) for new dataset that comes through the UI. It is assumed that all aspects of this project have the ability to work together as the designer is expecting and also the data on which our model is trained is as correct as possible.

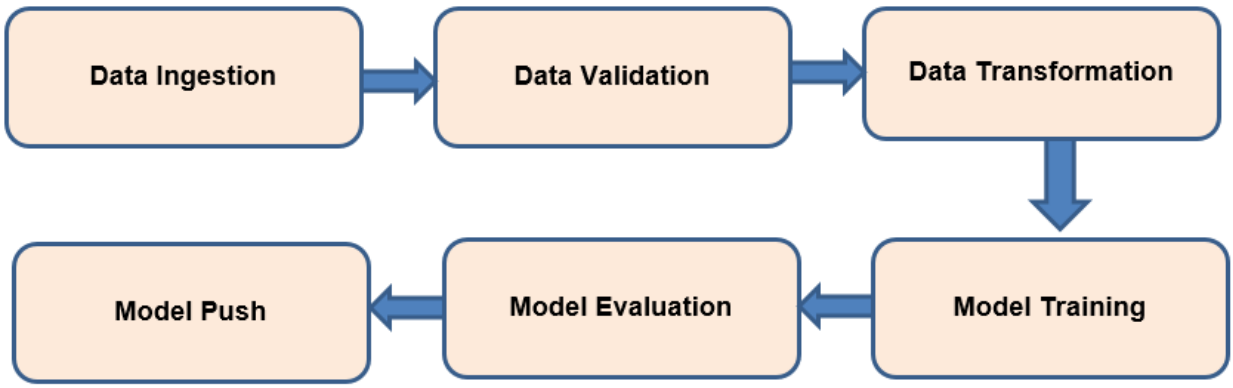
3 Design Details

3.1 Process Flow

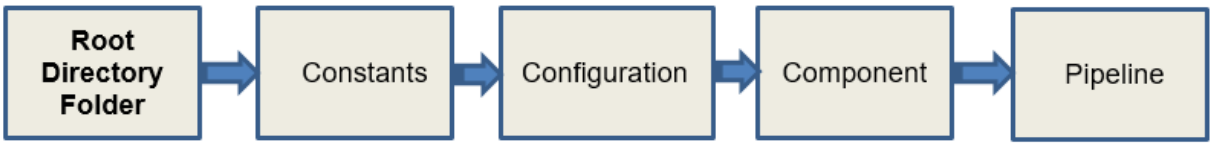
For identifying the different types of anomalies, we will use a deep learning base model. Below is the process flow diagram is as shown below.

Proposed methodology

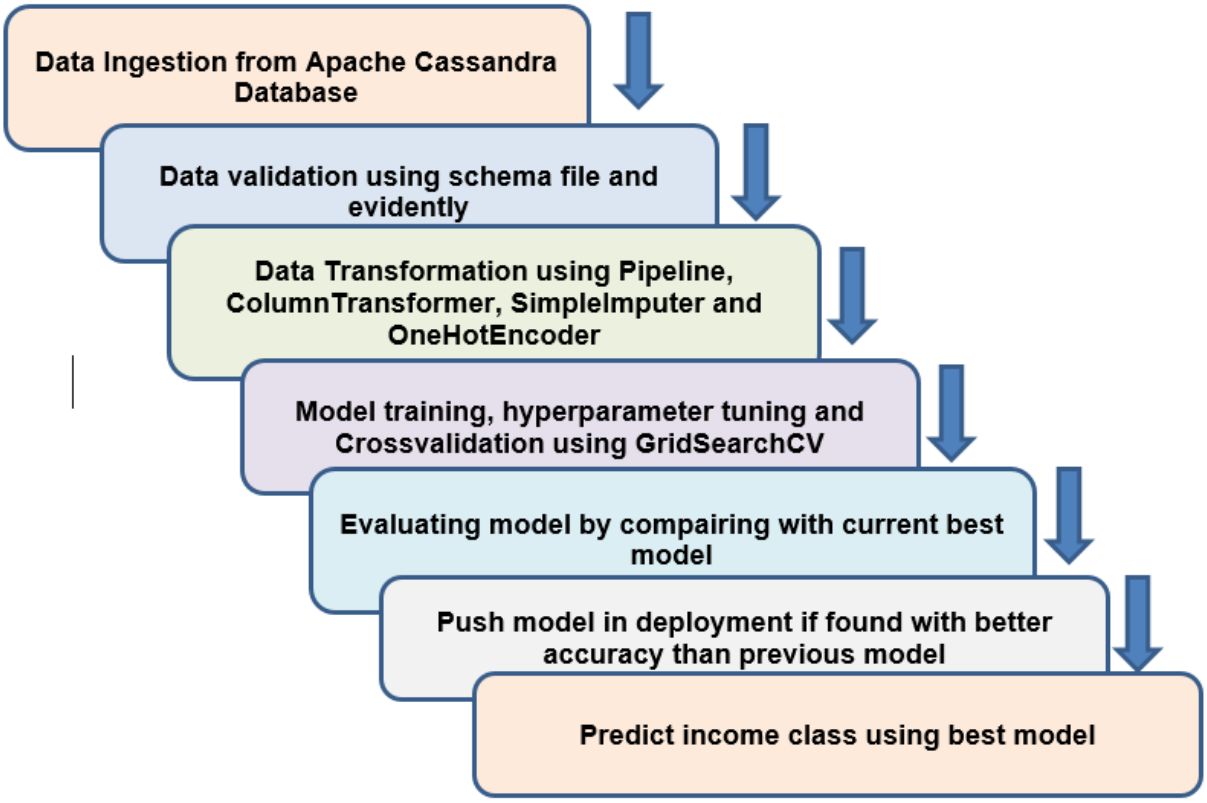
3.1.1 Components of Machine Learning Pipeline



3.1.2 Coding flow for building project structure



3.1.3 Complete pipeline flow



3.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Description:

1. The System should be able to log each and every system flow.
2. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

3.3 Error Handling

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

4 Performance

The Income Prediction tool is used to predict whether a person earns above or below 50K dollars per annum or not. So this is made keeping in mind that if it will 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 the 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 until that function is finished.

4.4 Deployment

5 Conclusion

The Income Prediction will give the income predictions of a person instantly and has the potential to help various organisations, agencies, companies, etc around the world in various tasks.

References

1. https://github.com/avnyadav/machine_learning_project
2. https://www.google.com
3. iNeuron FSDS Bootcamp course lectures