High Level Design (HLD)

INCOME PREDICTION

Revision Number: 0

Manojprabakaran

Document Version Control

Date Issued	Version	Description	Author
12th Dec 2023	1.0	First Submission	Manojprabakaran

Contents

Abstract

Income prediction is a fundamental task in the field of machine learning, with significant realworld applications. In this study, we focus on the predictive analysis of individuals' incomes, aiming to classify whether a person's income falls below or above the 50k threshold. Our research utilizes a dataset consisting of several key features, including sex, education, workclass, capital gains, capital losses, and the type of work performed. The primary objective of this project is to design a robust and accurate income prediction model, which can provide valuable insights for various socio-economic applications.

By leveraging algorithms and data analysis, we aim to deliver a tool that can assist policymakers, businesses, and researchers in making informed decisions related to income disparities, education, and employment. The insights gained from this research can contribute to a better understanding of socio-economic factors affecting individuals' incomes, ultimately helping create a fairer and more equitable society.

1 Introduction

1.1 Why this High-Level Design Document?

The purpose of this document is to present a detailed description of the Income Prediction . It will explain the features of the system, the interfaces of the system and what the system will do. This document is also intended to help detect contradiction prior to coding, and can be used as a reference manual for how the module interact at high level.

The HLD will :

- Present all 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 App compatibility o Resource utilzation o Serviceability

1.2 Scope

- The HLD documentation presents the structure of the system, such as the database architecture, application layers, application flow (navigation), and technology architecture.
- The HLD uses non-technical to mildly-technical terms which should be understandable to administrators of the system.
- This software system will be a Web application, this system will be designed to predict income and make better and easy.
- It aims to provide comprehensive income information, including expected salary, workclass, gender ,income loss and income gain

2 General Description

2.1 PROBLEM STATEMENT

In the realm of socio-economic analysis, income prediction stands as a vital component with far-reaching implications. As a society, we strive for equitable resource distribution and seek to address income disparities, create informed policy decisions, and enhance the quality of life for all citizens. Accurate income prediction models are essential tools for achieving these objectives.

The problem at hand is to develop a robust income prediction model that leverages a rich set of features, including age, workclass, sex, education, occupation, race, capital loss, capital gain, hours worked per week, and country of residence. The model should effectively categorize individuals' incomes into two groups, namely those earning below 50k and those earning above 50k.

2.2 PROPOSED SOLUTION

Income prediction project aims to deliver an advanced and precise system for forecasting individuals' income levels. By utilizing a comprehensive set of attributes, we intend to construct a predictive model that accurately categorizes incomes as either below 50k or above 50k.

Our proposed solution focuses solely on income prediction, and its primary objective is to provide an efficient and reliable tool for assessing income disparities. Through the application of machine learning and data analytics, our model will empower governments, businesses, and researchers to make well-informed decisions related to income inequalities. This system is dedicated to addressing the specific challenge of income prediction, contributing to a more equitable society by offering a better understanding of the factors influencing income levels and enabling data-driven decision-making.

2.3 FURTHER IMPROVEMENTS

In the realm of income prediction, our project can be expanded to provide even more valuable insights and applications. Beyond the core income classification task, we envision the following enhancements:

Feature Expansion: We can further enrich the dataset with additional attributes, such as location, industry, or personal financial habits, to improve the predictive accuracy of our income model.

Financial Planning Tools: Developing tools and resources that help individuals plan their budgets, savings, and investments based on their income category. This can empower users to make more informed financial decisions and improve their financial security.

Social and Policy Impact Analysis: Expanding the project to assess the social and policy impact of income disparities, providing insights for policymakers to design more effective interventions and programs.

By incorporating these improvements, our income prediction project can evolve into a multifaceted tool, not only for accurate income classification but also for offering valuable financial guidance, supporting individuals in achieving their financial goals, and contributing to a fairer and more financially secure society.

2.4 TECHNICAL REQUIREMENT

To use the system technical requirement is web browser. In some cases, the requirements could be the credentials.

2.5 Data Requirement

However, data requirement completely depends on our problem statement

- Age .
- Sex.
- Education .
- Workclass.
- Race.
- Captial Gain.
- Capital Losses.
- Working Hour Per Week.
- Country.

2.6 Tools Used

- Python programming language
- Web framework flask.
- IDE : Visual Studio Code (VS Code)
- MongoDB Atlas : for remote database
- Scikit-Learn for analysis
- Pandas for Data Ingestion
- Numpy
- For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
- AWS is used for deployment of the model.
- GitHub is used as version control system.
- MLflow for Tracing.

2.7 Constraints

- Security measures will be in place to protect user data and system integrity.
- System must be user friendly, as automated as possible and users should not be required to know any of the workings..

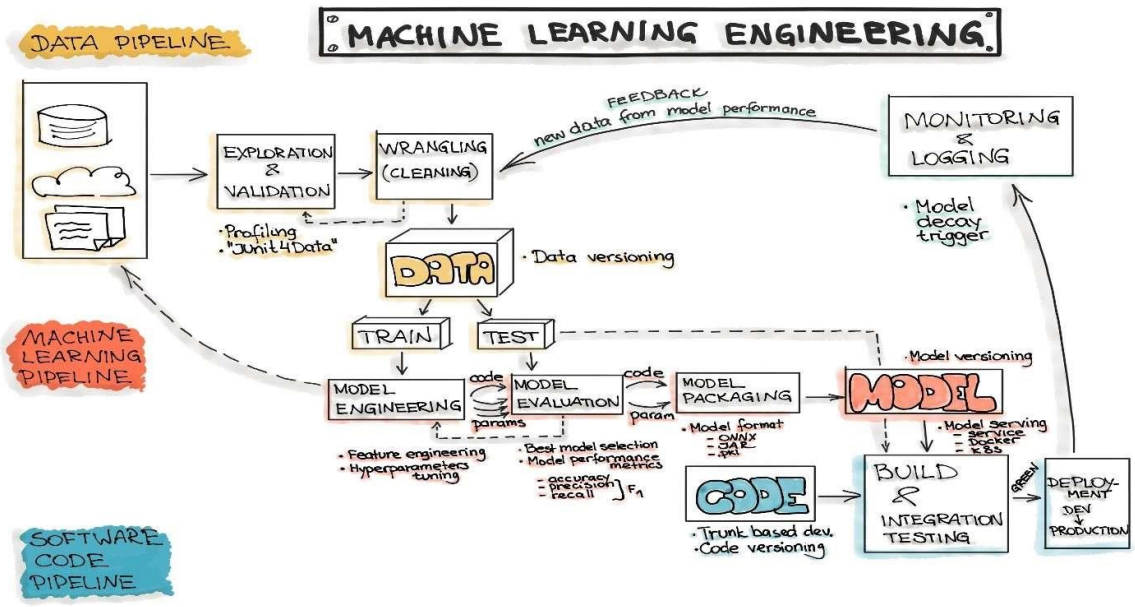
2.8 Assumptions

- Hardware Compatibility: Users' devices, including smartphones, tablets, and computers, are assumed to be compatible with the BIMS web and mobile applications.
- The system's scalability assumptions include the ability to handle increased user traffic and data volumes as the user base grows.

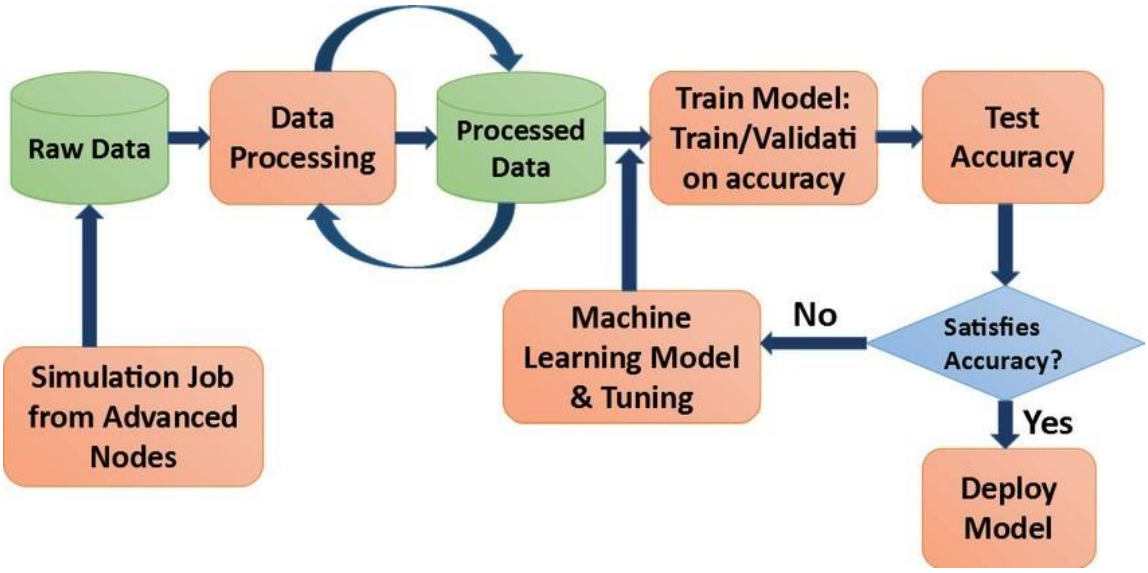
3 Design Details

3.1 Process Flow

Below given is all the process flow that should happen in the application



3.2 Model Training and evaluation



3.3 Deployment

- Deployment on cloud infrastructure (i.e., AWS) for scalability and availability.
- Continuous integration and continuous deployment (CI/CD) pipelines for automated updates.

3.4 Event Log

- The System identifies at what step logging required
- The System should be able to log each and every system flow.
- Developers can choose logging methods. You can choose database logging/ File logging as well.

3.5 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.

4 Performance

4.1 Reusability

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

4.2 Scalability and Load Handling

1. Scalable Machine Learning Models: Implement machine learning models that can scale
reasing prediction tasks. As the number of users and data points grows, the model should rem
2. Infrastructure: omputing resources to ensure the system's
y to allocate additional computational resources during
to handle spikes in prediction requests effectively.
3. Data Storage and Processing: mploy scalable data storage and processing solutions to
cess the data. Distributed databases and data processing
andle increased data volumes.
4. Parallel Processing: arallel processing techniques to speed up model training
e multiple requests simultaneously without compromising performance.
5. Load Balancing: echanisms to distribute incoming prediction
alanced workload and preventing overload on any single server.
6. Monitoring and Optimization: nuously monitor system performance and optimize
onsiveness and handle increasing loads efficiently.

5 Conclusion

In conclusion, In the pursuit of addressing income disparities and fostering data-driven decision-making, our income prediction project represents a significant step forward. Through a meticulous analysis of various attributes, including age, workclass, sex, education, occupation, race, capital loss, capital gain, hours worked per week, and country of residence, we have developed a predictive model capable of accurately categorizing incomes as below or above 50k.

This project extends beyond mere data analysis; it stands as a testament to the power of machine learning and data-driven insights. By creating a reliable income prediction model, we empower governments, businesses, and researchers to make informed decisions related to income inequalities, labor market dynamics, education strategies, and social welfare programs.

The path to a fairer and more equitable society begins with understanding the factors that influence income disparities. Our income prediction model illuminates these factors, offering a better-informed perspective on socio-economic challenges. As we move forward, we can refine and expand this model, integrating additional attributes and offering financial guidance, further contributing to a society where opportunities and resources are distributed more fairly, ultimately improving the quality of life for all.