Architecture Document

Adult Census Income Prediction

Revision Number : 1.0

Last date of revision : 20/06/2022

Ritesh Naik

Darshan Naik

Document Version Control

Date Issued	Version	Description	Author
26-June-2022	1.0	Initial Release	Ritesh Naik

Contents

Abstract

Inequality in wealth and income is a major source of worry, particularly in the United States. One reasonable motivation to lessen the world's rising level of economic disparity is the possibility of reducing poverty. The notion of universal moral equality promotes long-term development and improves a country's economic stability. Governments in several countries have been working hard to address this issue and find the best answer possible. The goal of

this project is to demonstrate how machine learning and data mining techniques can be used to solve the problem of income inequality. The Adult Dataset from UCI was utilized for this.

Based on a set of attributes, classification has been done by building a web app to forecast whether a person's annual income in the United States fits into the income categories of greater than 50K Dollars or less than 50K Dollars.

1. Introduction

The goal of this High-Level Design (HLD) Document is to provide all of the pertinent information regarding this project. This document gives overall understanding about the project.

1.1 Purpose of the Document

The purpose of this document is to predict whether a person earns more than $50K/year or not based on the following features.

- age
- workclass
- education
- marital-status
- occupation
- relationship
- race
- sex
- capital-gain
- capital-loss
- hours-per-week
- country

We are using the dataset given by iNeuron company which is “1994 Adult Census Dataset”

1.2 Scope

This software system will be a Web application. This system will be designed to understand the salary pattern when we provide features to it. Model will be trained on the “1994 Adult Census Dataset”. This system is designed to predict whether the person earns more than $50K or not based on the inform information such as native-country, hours-per-week, race, relationship etc

1.3 Constraints

Dataset we are using is old & cannot be used to understand the present industry salary prediction.

2. Technical Specification

2.1 Dataset

You can find the Dataset here
UCI Machine Learning Repository: Adult Data Set

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	country	salary
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	United-States	<=50K
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	United-States	<=50K
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	United-States	<=50K
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	United-States	<=50K
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	Cuba	<=50K

This dataset consist of 15 features & 32561 rows

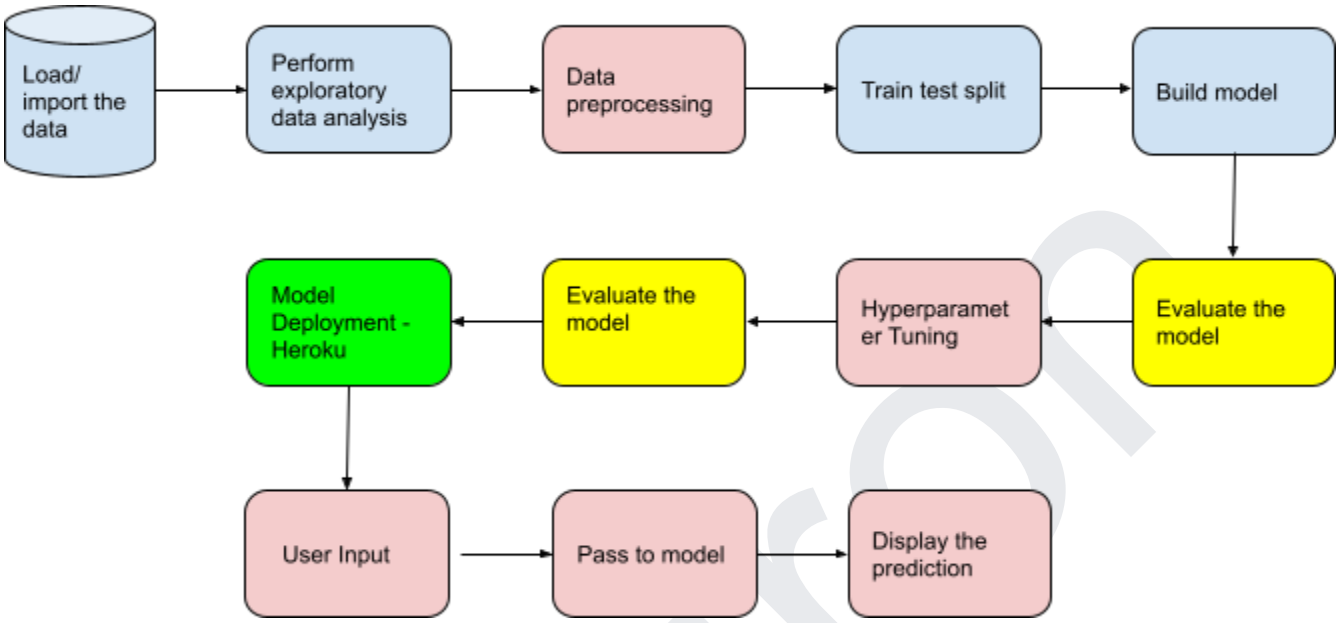
2.2 Predicting income category

- The webapp shows the form that needs user input.
- The user chooses the option from the markdown and fills in the information as needed.
- The user then selects the "Predict" button after giving out the necessary information.
- The model will be fed with the processed user input after the user input has been processed in the backend.
- From the input, the system should be able to determine whether the income exceeds $50,000 or not.

2.3 Deployment

We are using heroku to host our app. Heroku ,which is a platform as a service (PaaS) that allows users to create, launch, and manage all of their apps in the cloud.

3. Process Architecture



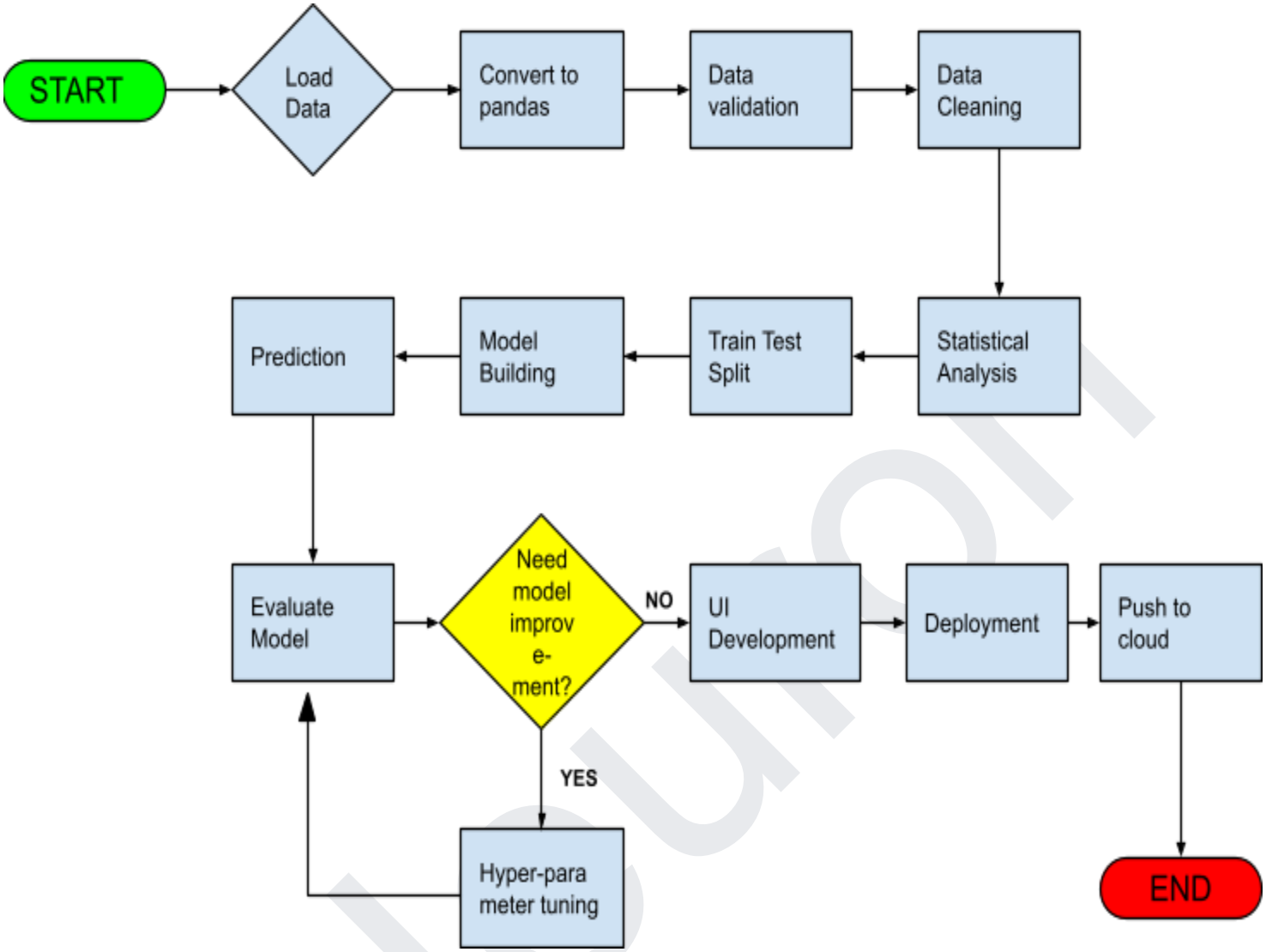
4. Technology Stack

Front End	HTML & CSS
Back End	Python - Flask
Deployment	Heroku

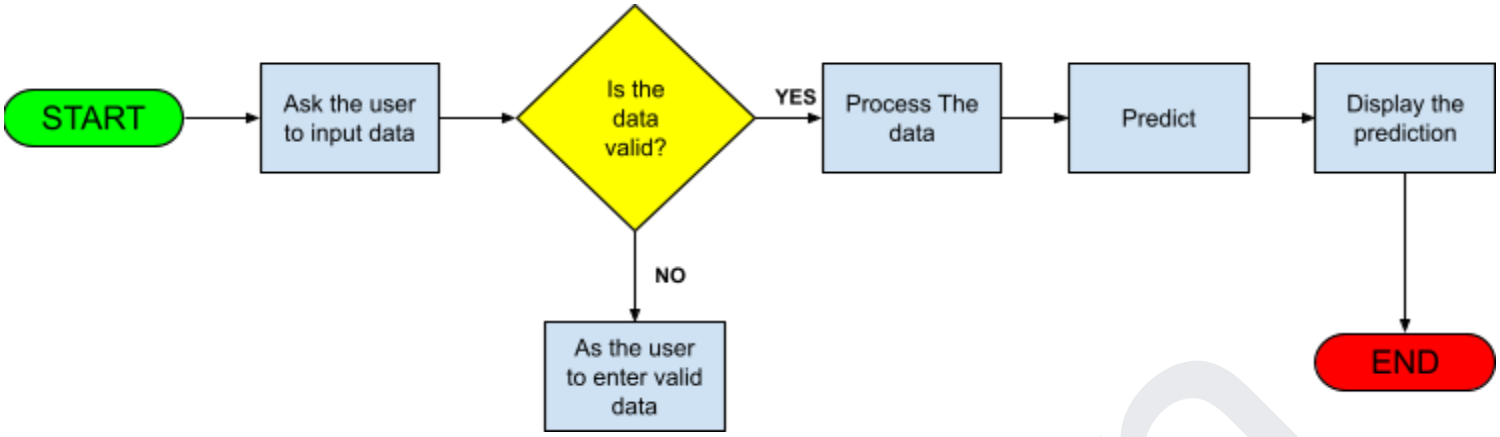
5. Proposed Solution

Of all the 4 models we have built, the **XGboost** model not only has the highest accuracy but also the highest precision and F1_score. Hence we are using XGBoost as our final model.

6. Model training/ Validation workflow



7. User Input/ Prediction flow


START
Ask the user to input data
Is the data valid?
YES
Process The data
Predict
Display the prediction
END
NO
As the user to enter valid data