# High Level Design

## Adult Census Income Prediction

| **Written By** | **Sachin Kumar** |
|---|---|
| Document version | 0.1 |
| Last Revised date | 08-06-2022 |
| | |

## Document Control

Change Record:

| Version | Date | Author | Comments |
|---|---|---|---|
| 0.1 | 08-06-2022 | Sachin Kumar | HLD — V1.0 |

# Contents

# Abstract

The use of machine learning models to improve prediction problems and handle increasingly large datasets is a rising trend in economics. Prediction plays a particularly important role in applied economics because it provides critical insights to assess market outcomes. This study builds to showcase the relative power of these modelling methodologies in economics through the prediction of income. This research utilizes data from the Current Population Survey from 2017 – 2020, containing 32562 observations and 15 variables. 2017- 2018 data served as training data for the models and 2019-2020 served as data for the two testing sets. The results show that machine learning models performed better than traditional prediction approaches in predicting individual total income. The high performance of the machine learning models supports that these methodologies should be utilized alongside more traditional techniques to assist in economic research focusing on prediction. With further development, these models could be used with great effect to assist in both the public and private sectors.


## 1. Introduction

### 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
    - Security
    - Reliability
    - Maintainability
    - Portability
    - Reusability
    - Application compatibility

- Resource utilization
- Serviceability

## 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.

## 3. Definitions

| Term | Description |
|---|---|
| AWS | Amazon Web Services |
| Cassandra Database | Collection of all the information monitored by this system |
| IDE | Integrated Development Environment |

## 4. General Description

The Income Prediction Surveillance solution system is a machine learning project which will help to improve prediction problems and handle increasingly large datasets is a rising trend in economics. this project could be used with great effect to assist in both the public and private sectors.

## 5. Problem statement

To build a classification methodology to predict the individual income based on given training data.

## 6. PROPOSED SOLUTION

The machine learning models (K-Mean, Naïve bayes, XGBoost) can be used to do the individual prediction.

The model with the based accuracy will be used further for making predictions.

This approach is much more accurate compared to traditional prediction approaches.

## 7. FURTHER IMPROVEMENTS

Adding more data to the models will make the model more reliable.

Adding high computations computers will help to run the project in less spam of time.

This project can be implement in various public and private sectors.

Private sectors like – E-commerce companies, this will be help to make rule to provide offers to certain customers based on their income.

Govt sectors – Help to build policy to reach out to individuals having income lower than a certain range and help to deliver different schemes like Pradhan Mantri Jeevan Jyoti Bima Yojana (PMJJBY), Pradhan Mantri Mudra Yojana.etc.

## 8. Data Requirements

The data used in this paper is from the Current Population Survey, which is conducted and sponsored jointly by the India Census Bureau and the India Bureau of Labor Statistics (U.S. Census Bureau). The data was extracted from IPUMS and includes the data from 2017-2020. Data from 2017 and 2018 is used to train the models, while 2019 and 2020 data is used to as the validation set to test the models. This uses typical methodology from machine learning, in which the models are trained on one subset of data and then evaluated on the predictions produced for a different subset called the validation data. The actual and predicted values for the validation data are compared, allowing for reasonable comparison of the models. The CPS is a voluntary survey conducted each month for approximately 60,000 households. The survey was chosen for its high dimensionality, large number of observations, completeness of data, and its reputation and usage in the field.

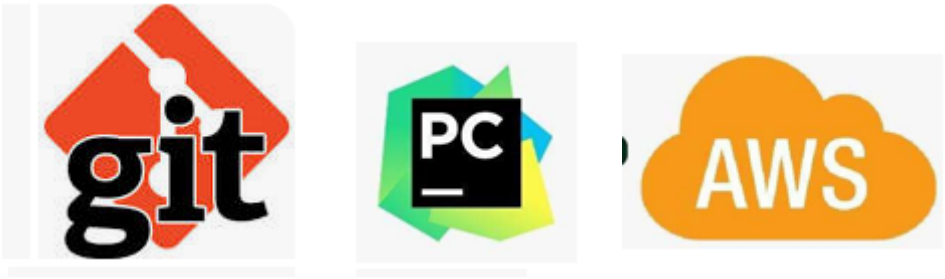
The Current Population Survey contains a plethora of information from each respondent and their household. The approximately 150 variables extracted capture a variety of characteristics for individuals from categories such as work, income, education, ethnicity, tax status, poverty

These original variables were recoded to create 264 unique variables that were used in the analysis. These new variables created include a large number of dummy variables created for categories such as state of residence, and occupation of the individual.

The response, or target, variable for the analysis is the natural log of real income for individuals.

## 9. Tools used

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


- PyCharm is used as IDE.
- For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
- AWS is used for deployment of the model.
- Cssandra DB is used to retrieve, insert, delete, and update the database.
- Front end development is done using HTML/CSS
- Flask is used for backend development.
- GitHub is used as version control system.

## 10. Constraints

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

## 11. Assumptions

The main objective of the project is predict the income of individuals based on a given data. Adding more data will make the model more reliable.

# 12. Design Details

Application control flow:-

**Step1** – Enter point for the application is main.py/app.py file.

**Step2** – There will be different routes which will be called inside it.

**Step 3** – Build flask application

**Step4** – Inside the main.py file there will different routes and control will be transferred according to the route getting called.

There are three routes.

Home

Training

Prediction

**Home** – This route is the application home page, ones the model gets trained, we can do the prediction from the web UI as well.

Home route just calls the home page of the application.

**Training** – The training route will be triggered ones the users gets the location of the file, or the location of the folder where the training files are kept.

Now the training will be divided into 2 parts.

**A) Validation**

**B) Actual Training**

A) **Validation** – will check whether the no of columns are correct or not.

Whether the file name format is correct or not

Replacing Na with Null

Replacing single quotes with double quotes.

These all steps will be done in validation and finally all the data will be segregated and store into a training database.

Output of the training database is Inputfile.CSV file and this CSV file will act as a training data.

B) **Actual Training** –Here we are going to do data preprocessing, will perform several steps like feature selection or column selection.

Now will do missing value imputation.

We might try to normalize or data via standard scaler.

Convert categorical values into numerical values.

Once the data is pre-processed will perform clustering ones the clustering is done and appropriate number of clusters created for the machine learning model are selected.

Now we are going to the actual machine learning model training

Now we will select the model and perform the hyper parameter tuning for each of the models and final outcome of this is going to be saved models for individuals clusters.

Once successfully model is saved training will end here.

Once the training is completed the user can do predictions.

**Prediction** – For predictions same things are going to happen, like there will be some data preprocessing and even before doing data preprocessing we need to do data validations.

This data validations will be having same approach like we will check whether the number of columns are correct or not

Whether the file name format is correct or not

Replacing Na with Null

Replacing single quotes with double quotes.

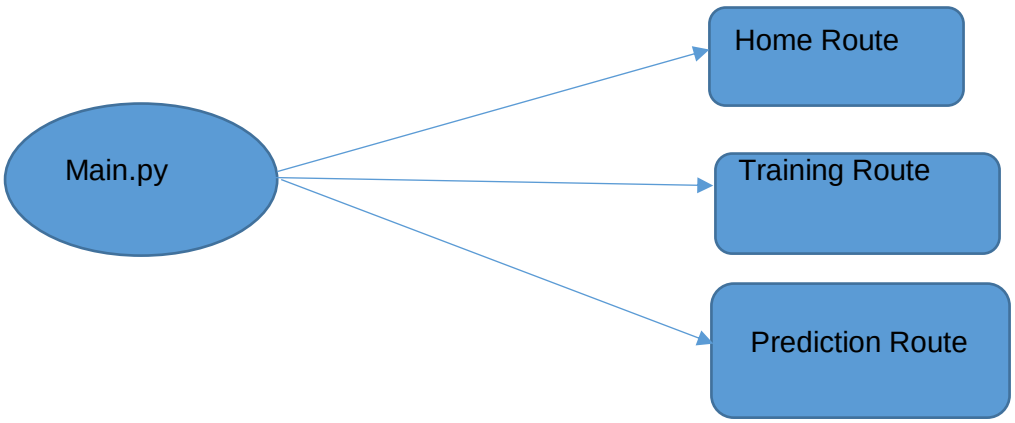
Post validation is done will combine everything whatever user has sent for prediction will combine it and store it in DB. And from this DB will export a CSV file, and this CSV file will act as a input for our actual prediction.

**Actual prediction** – Here will be doing preprocessing and then clustering post will do prediction.

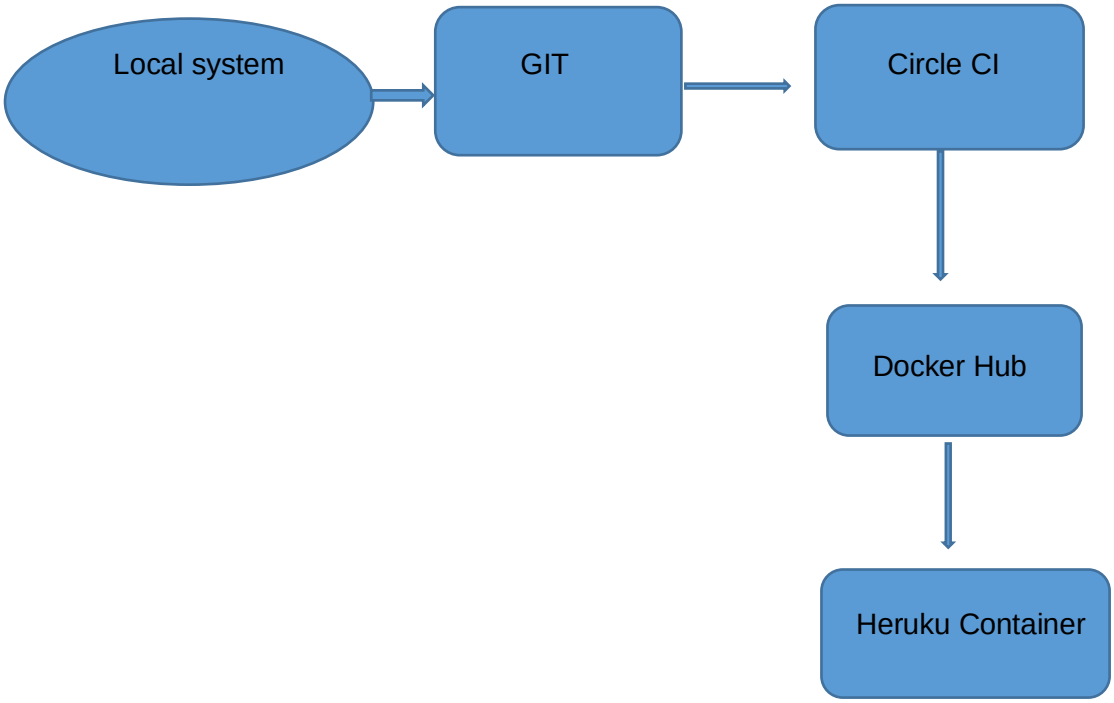
**Preprocessing** – likewise training here again we will select features, missing values imputations, Normalize the data via standard scaler, balance and imbalance data handling etc.

**Clustering** - Here we will feed our data to pre trained clustering model and based on the row will give different cluster numbers, in simple terms will tell us which row belongs to which clusters. Post we going to segregate the data based on clusters and for individual clusters we have saved individual models, now those individual models for the cross ponding clusters will be used to do the final prediction.

Again final prediction will be saved into CSV file, which will be final output of our model prediction.



## 13. Deployment process



## 14. Event log

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

*Initial Step-By-Step Description:*

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. Developer can choose logging method. You can choose database logging/ Filelogging as well.
4. System should not hang even after using so many loggings. Logging just becausewe can easily debug issues so logging is mandatory to do.

## 15. Error Handling

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

## 16. Performance

The Income prediction for individuals are used in public and private sectors both.

This help to launch difference schemes/policy based on income of individuals.

Adding more data and high computational computer the process will be much more simpler and prediction will more accurate.

### *A. Reusability*

The code written and the components used should have the ability to be reused with noproblems.

### *B. 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.

### *C. Resource Utilization*

When any task is performed, it will likely use all the processing power available until that function is finished.

## 17. Deployment




# 18. CONCLUSION

This research finds that machine learning methodologies outperformed the traditional OLS model with variable selection from literature in the area of prediction. The OLS regression performed worse across all three metrics relative to the machine learning models for prediction. This demonstrates that machine learning methodologies could be effective to supplement other models to assist with research focusing on prediction. However, it is important to note that the lower performance of the OLS regression also comes as a result of key differences between OLS and machine learning methods. The machine learning models were able to consider and utilize more features (or independent variables) relative to the OLS regression. The mere fact that more variables were potentially included in the machine learning models is one reason why they outperformed the OLS regression. However, the OLS regression was chosen as the baseline comparison with this in mind. The traditional OLS regression still has many benefits, including its interpretability and lower computing needs, but often requires individuals to create several regressions and self-select variables to be included based on previous literature or other factors. There are methods like stepwise regression that can help with the variable selection, but oftentimes variables are selected by the researcher and fewer variables are considered. The more traditional techniques in economics and machine learning actually complement each other rather than serve as substitutes for one another. The improved prediction power of machine learning methods can be used with tested techniques to further advance research and lead to new outcomes.

This approach will help private and public sectors to launch different schemes directing directly to the individuals based on income.