**Architecture**

**INCOME PREDICTION**

Revision Number – 1.0

Last Date of Revision – 16-05-2023

**SHAHIN ANJUM**

**Liabaries Requirment**

Flask

seaborn

pandas

numpy

scikit-learn

**Content**

# **Abstract -----------------------------------------------------------------------------------4**

**Introduction ---------------------------------------------------------------------------- 4**

**Why the Architecture Design Documents? ------------------------------ 4**

**1.Architecture ------------------------------------------------------------------------ 4**

**2.Architecture Design ------------------------------------------------------------- 5**

**2.1 Data gathering from main source --------------------------------------- 5**

**2.2 Data description -------------------------------------------------------------- 5**

**2.3 Upload data inti Cassandra ----------------------------------------------- 5**

**2.4 Export data from database ----------------------------------------------- 5**

**2.5 Data pre-processing ------------------------------------------------------- 5**

**2.6 EDA ------------------------------------------------------------------------------ 6**

**2.7 Modelling ---------------------------------------------------------------------- 6**

**2.8 UI integration ---------------------------------------------------------------- 6**

**2.9 Data from user --------------------------------------------------------------- 6**

**2.10 Data validation -------------------------------------------------------------- 6**

**2.11 Rendering the results ---------------------------------------------------- 6**

**2.12 Deployment ------------------------------------------------------------------ 6**

**Abstract**

­ For this Internship, I examine the Census Income dataset available at the UC Irvine Machine Learning Repository. I aim to predict whether an individual’s income will be greater than $50,000 per year based on several attributes from the census data.

**1 Introduction**

The US Adult Census dataset is a repository of 48,842 entries extracted from the 1994 US Census database. In our first section, we explore the data at face value in order to understand the trends and representations of certain demographics in the corpus. We then use this information in section two to form models to predict whether an individual made more or less than $50,000 in 1994. In the third section, we look into a couple papers written on the dataset to find out what methods they are using to gain insight on the same data. Finally, in the fourth section, we compare our models as well as that of others in order to find out what features are of significance, what methods are most effective, and gain an understanding of some of the intuition behind the numbers.

Why this Architecture Design documentation?

The main objective of the Architecture design documentation is to provide the internal logic understanding prediction code. The Architecture design documentation is designed in such a way that the programmer can directly code after reading each module description in the documentation.

1 Architecture

A picture containing text, screenshot, font, diagram

Description automatically generated

**2 Architecture design**

This project is to create an interface for the user to know the heating cooling amount, in addition to this, in need of getting the real time project experience we are importing the gathered data into our own database and then start the project from the scratch.



**2.1 Data gathering from main source**

The data for the current project is being gathered from kaggle, the link to the data is:

<https://www.kaggle.com/datasets/overload10/adult-census-dataset>

**2.2 Data description**

The Census Income dataset has 48,842 entries. Each entry contains the following information about an individual:

● age: the age of an individual

○ Integer greater than 0

● workclass: a general term to represent the employment status of an individual

○ Private, Self­emp­not­inc, Self­emp­inc, Federal­gov, Local­gov, State­gov, Without­pay, Never­worked.

● fnlwgt: final weight. In other words, this is the number of people the census believes the entry represents.

○ Integer greater than 0

● education: the highest level of education achieved by an individual.

○ Bachelors, Some­college, 11th, HS­grad, Prof­school, Assoc­acdm, Assoc­voc, 9th, 7th­8th, 12th, Masters, 1st­4th, 10th, Doctorate, 5th­6th, Preschool.

● education­num: the highest level of education achieved in numerical form.

○ Integer greater than 0

● marital­status: marital status of an individual. Married­civ­spouse corresponds to a civilian spouse while Married­AF­spouse is a spouse in the Armed Forces.

○ Married­civ­spouse, Divorced, Never­married, Separated, Widowed, Married­spouse­absent, Married­AF­spouse.

● occupation: the general type of occupation of an individual

○ Tech­support, Craft­repair, Other­service, Sales, Exec­managerial, Prof­specialty, Handlers­cleaners, Machine­op­inspct, Adm­clerical, Farming­fishing, Transport­moving, Priv­house­serv, Protective­serv, Armed­Forces.

● relationship: represents what this individual is relative to others. For example an individual could be a Husband. Each entry only has one relationship attribute and is somewhat redundant with marital status. We might not make use of this attribute at all

○ Wife, Own­child, Husband, Not­in­family, Other­relative, Unmarried.

● race: Descriptions of an individual’s race

○ White, Asian­Pac­Islander, Amer­Indian­Eskimo, Other, Black.

● sex: the biological sex of the individual

○ Male, Female

● capital­gain: capital gains for an individual

○ Integer greater than or equal to 0

● capital­loss: capital loss for an individual

○ Integer greater than or equal to 0

● hours­per­week: the hours an individual has reported to work per week

○ continuous.

● native­country: country of origin for an individual

○ United­States, Cambodia, England, Puerto­Rico, Canada, Germany, Outlying­US(Guam­USVI­etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican­Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El­Salvador, Trinadad&Tobago, Peru, Hong, Holand­Netherlands. ● the label: whether or not an individual makes more than $50,000 annually. ○ <=50k, >50k

**2.3 Import data into Cassandra**

Created an api for the upload of the data into the Cassandra database, steps performed are:

* Connection is made with the database.
* Created a database with name Energy Efficiency.
* Cqlsh command is written for creating the data table with required parameters.
* And finally, a cqlsh command is written for uploading the dataset into data table by bulk insertion.
* .

2.4 Export data from database

In the above created api, the download url is also being created, which downloads the data into a csv file format.

2.5 Data pre-processing

Steps performed in pre-processing are:

* First rename all columns
* The data types are being checked and found all columns are of type integer.
* Checked for null values as there are no null values.
* Checking outliers as looking box plot we don’t need to worry about outliers.
* Scaling is performed for required data.
* And the data is ready for passing to the machine learning algorithm.

2.6 EDA

I first plot bar graph of targets columns with all independent columns. By these graphs I concluded many things which is written in EDA Jupyter notebook.

**2.7 Modelling**

The pre-processed data is then visualized, and all the required insights are being drawn. Although from the drawn insights, the data is randomly spread but still modelling is performed with different machine learning algorithms to make sure we cover all the possibilities. And finally, as expected random forest regression performed well.

**2.8 UI integration**

Both CSS and HTML files are being created and are being integrated with the created machine learning model. All the required files are then integrated to the app.py file and tested locally. Note I copy the code from internet and make the CSS and HTML File.

**2.9 Data from user**

The data from the user is retrieved from the created HTML web page.

**2.10 Data validation**

The data provided by the user is then being processed by app.py file and validated. The validated data is then sent for the prediction.

**2.11 Rendering the results**

The data sent for the prediction is then rendered to the web page.

**3 Deployment**

I didn’t deploy.

6

**ScreenShot of the App Interface which I will deploy**



