Detailed Project Report of Income Prediction

Ojective:

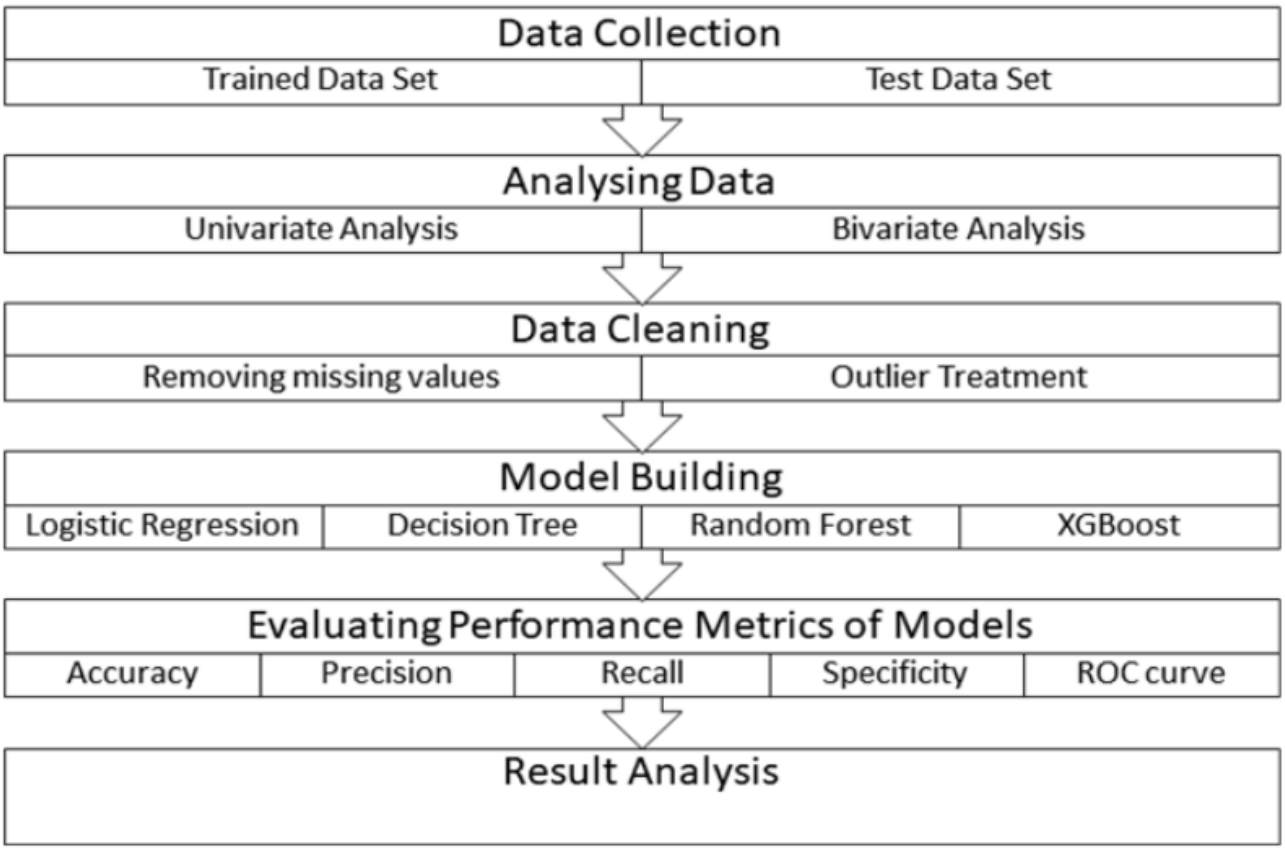
The purpose of this project is to use data transformation and machine learning to create a model that will predict a salary when given years of experience, job type. The purpose of this project is to use data transformation and machine learning to create a model that will predict a salary when given years of experience, job type.

Benefits:

- Used To Predict Salaries Years Experience: How many years of experience Information .

- This model can be used as a guide when determining salaries since it shows reasonable predictions when given information on years of experience.

Methodology:



Data Sharing Agreement:

- Sample file name (ex incomeprediction_20162021_112)
- Length of data stamp(8 digits)
- Length of time stamp(6 digits)
- Number of Columns
- Column names
- Column data type

Features:

- **Age :** continuous. It denotes the age of the person
- **Workclass**: It denotes the working class of the person.
- **Fnlwgt**: continuous.
- **Education**: It denotes the educational qualification of the person.
- **Education**-num: continuous. It denotes the quantitative values with reference to education.
- **Marital-status**: It denotes the marital status of the person.
- **Occupation**: It denotes the occupation of a person.
- **Relationship**: It denotes the people present in the family.
- **Race**: It denotes the person’s origins.
- **Sex**: It denotes the person's gender.
- **Capital-gain**: continuous. It denotes the monitory gains by the person
- **Capital-loss:** continuous. It denotes the monitory loss by the person.
- **Hours-per-week:** continuous. It denotes the number of working hours per week by the person.
- **Native-country:** It denotes the country to which the person belongs

Data Validation and Data Tranformation:

- **Name Validation:**

  We validate the name of the files based on the given name in the schema file. We have created a regex pattern as per the name given in the schema file to use for validation.

  If all the values are as per requirement, we move such files to "Good_Data_Folder" else we move such files to "Bad_Data_Folder."

- **Number of columns:**

  We validate the number of columns present in the files, and if it doesn't match with the value given in the schema file, then the file is moved to "Bad_Data_Folder."

- **Name of columns:**

  The name of the columns is validated and should be the same as given in the schema file. If not, then the file is moved to "Bad_Data_Folder".

- **The data type of columns:**

  The data type of columns is given in the schema file. It is validated when we insert the files into Database. If the data type is wrong, then the file is moved to "Bad_Data_Folder"

- **Null values in columns:**

  The data type of columns is given in the schema file. It is validated when we insert the files into Database. If the data type is wrong, then the file is moved to "Bad_Data_Folder"

Data Insertion into Database:

- **Database creation and connection:**

  Create a database with the given name passed. If the database has already been created, open a connection to the database.

- **Table creation in the database:**

  Table with name - "Good_Data", is created in the database for inserting the files in the "Good_Data_Folder" based on given column names and datatype in the schema file.

- **Insertion of files in the table:**

  The files in the "Good_Data_Folder" are inserted in the above-created table. If any file has invalid data type in any of the columns, the file is not loaded in the table and is moved to "Bad_Data_Folder".

Model Training:

Data Export from db:

The data in a stored database is exported as a CSV file to be used for model training.

Data Preprocessing:

a) Drop the columns not required for prediction.

b) Remove the unwanted spaces in data.

c) For this dataset, the null values were replaced with '?' in the client data. Those '?' have been replaced with NaN values.

d) Check for null values in the columns. If present, impute the null values using the categorical imputer.

e) Replace and encode the categorical values with numeric values.

f) Scale the numeric values using the standard scaler.

g) Handle the imbalanced dataset using oversampling.

Clustering:

Kmeans algorithm is used to create clusters in the preprocesed data.The optimum number of clusters is seletected by plotting the belwo plot, and for the dynamic selection of the number of clusters, we are using KneeLocator function. The idea behind clustering is to implement diffrent algorithms.

Model selection:

- Preforming EDA to get insight of data like identifying distribution, outlier, trend, among data etc.
- Check for null values injw the columns. If present impute the null values.
- Encode the categorical values with numeric values.
- Perform standard scaler to scale down the values.

Prediction:

- The Client will send the data in multiple sets of files in batches at a given location. Data will contain the annual income of various persons.
- Apart from prediction files, we also require a "schema" file from the client, which contains all the relevant information about the training files such as
- Name of the files, Length of Date value in FileName, Length of Time value in FileName, Number of Columns, Name of the Columns and their datatype.

Q & A :

Q1). What is the source of data?

The data for training provided by the client in multiple batches and each batch contain multiple files.

Q2). What was the type of data?

The data was the combination of numerical and categorical values.

Q3). What is the complete flow you followed in this Project?

Refer slide 3 for better understanding

Q4). After the file validation what you do with incomplete file or files which didn’t pass the validation?

Files like these are moved to the achive folder and list of the files has been shared with the client and we removed the bad data folder.

Q5). How logs are managed?

We are using different logs as per the steps that we follows in validation and modeling like file validation log, Data insertion, Model Training log, Prediction log.