Prediction Of Income Levels Of Individuals Based On Demographic Attributes

Chanpreet Singh

(School of Information Technology)

Abstract— The behavior of an unbalanced ratio of wealth and income is a very challenging problem in an economy of a particular country or region raising a lot of ups and downs in the stock market as well as the fear to create an economy that has been subject to conflicting forces. The likelihood of diminishing poverty is one valid reason to reduce the world's surging level of economic inequality. A lot of countries have set up government departments, which are working on research and study to control this problem for future increasing economy databases. This study aims to show the usage of Decision trees, GBM, ANN, Naive Bayes, Random Forest, machine learning and other data mining techniques in providing a solution to the income equality problem as well as the comparative analysis among them to study the better selection of algorithmic models to develop a better prediction under what factors are affecting the major parts and helps in learning the ambiguity of data caused by the data category under missing or invalid inputs in minor parts. The UCI Machine Learning repository Adult dataset [42] has been used for the purpose. The classification has been done to predict whether a person's yearly income in the US falls in the income category of either greater than 50K Dollars or less equal to 50K Dollars category based on a certain set of attributes.

Index Terms— Data Mining, Adult Data set [42], R, Machine Learning

Keywords—Decision trees, Naive Bayes, Random Forest, GBM, Logistic regression

I. INTRODUCTION

The Dependence of humans on data collection and analysis in any field that involves economy has grown on a tremendous scale. The scalability to perform utility functions on gathering missing information, detection of frauds, processing data for prediction in various fields which involves Artificial intelligence, Data Mining, machine learning is becoming challenging as the size of data gets bigger, the ambiguity for missing values also gets increased. Hence, making prediction of future events gets more difficult. In recent years, the problem of income inequality has been a great concern. For example a country like the US, the economy has traditionally been understood in terms of its geography. The physical attributes of each region have influenced the development in various aspects from the beginning of economy cycle with the arrival of the first colonists. The importance of human capital has been considered a vital aspect in modern economics profession.

As such, population and education levels had a great significance in a region's economic output [59].

Meaningful statistics about the direction of the economy start with the major market indexes that provide information for Stock and stock futures markets, Bond and mortgage, interest rates, the yield curve, Foreign exchange rates, as well as Commodity prices, especially gold, grains, oil, and metals.

Data analysis helps in Economic Forecasting, referred as a technique using a combination of widely followed features in current trends within the methodology for predicting the future outcomes of the economy. The future growth rate of gross domestic product (GDP) is basic charge to run under the economic forecasting which includes statistical modeling to be built up using various parent key variables as well as different types of indicators such as interest rates, worker productivity, industrial production, inflation, consumer confidence, unemployment rates and retail sales etc. [57][59].

This research will focus on the prediction of the income level of an individual by using various classifier methodologies using programming language R. The dataset provides information about individuals in different demographic regions under which the main target aim of this paper will be to find the deviation of income of an individual to be less or more than 50,000.

The demographic information scale objective acts as predictors of income variable. In this paper, a comparative analysis of data will be performed using study of the performance of Naive Bayes, Decision Tree, Random Forest, logistic regression, GBM and other classification techniques for the dataset.

II. MOTIVATION

The process of guesstimating data information using analytical and logical reasoning to inspect each component of the data information provided. Data information from various sources is gathered, reviewed, and then analyzed and investigate to form some sort of finding or conclusion as denouement. There are a variety of specific data analysis method, some of which include data mining, text analytics, business intelligence, and data visualizations.

Predictive Analytics Work on five stages that form the core of every efficient predictive analytics which includes Identifying Information as Data Outcomes, Determining Data Required to Train, Training the System, Validating the Results, and Using the Insights. For example predictive analysis in Banking and Financial Services where Banking and other Economical institutions are implementing new techniques to make sure that they are able to provide better customer services and schemes which benefits the bank to

prevent frauds and aware about the secure income level of their customers in advance from the prediction model technique described in this research. Data processing is the central hub for machine learning and predictive analytics. Predictive analytics act as a branching for machine learning in extended version to enable the actions performing in recognizing and understanding patterns and Automatized-gaining knowledge from data [60][11][40].

Such an analysis will help to set focus on the important areas which can significantly improve the income levels of individuals using better prediction models. The scope of analytical approach and unique methodology to be used in future as per the dataset methods applied already by other users does not involve the implementation of other algorithms which are not conducted in deployment of technologies through which we can improve accuracy and get better prediction outputs by performing data modeling under categories such as logistic regression, gradient boosting machine (GBM), Artificial Neural networking (ANN), K-Nearest Neighbors, Learning Vector Quantization, Linear Regression, as well as Support Vector Machines. An example of the classification purpose, many classifiers like naive Bayes will be applied that will classify data with respect to income level. So for the future goal improvements to be made in a way initializing with the data cleansing, the transformation of variables into vector form and then applying various algorithms [54][11][40].

The data cleaning process includes removal of punctuation marks, impute or deal missing values, handling tokens that help analytical data modules to study easily and increase their efficiency of the output and then the data will be transformed into the matrix form for further calculations. Then high-performance statistical values can be graphically plotted and visualized, using several visualization libraries in R, that indicates easily what are the influencing factors and helps to access the statistical relationship or correlation between variables [54]. For example, analyzing the range of income according to ages with use of boxplot.

The data matrix is then used for operations on which we apply various classification algorithms, data modeling and analytical techniques to get a more accurate prediction of the income level based on the demographic attributes of an individual. The comparison among these techniques can help us understand which model better suits the indicated data set. The key attributes in the research include Age, Workclass, education, marital status, occupation, relationship, race, sex, capital gain, capital loss, hours per week, native country, income level. The prediction task associated with this data set is to predict whether or not a person makes more than $50K a year using census data. The dataset contains a total of six continuous and eight nominal attributes. Where the basic key constraint is that the age, occupation, and education play a major role in deciding an individual's income[11][40].

III. DATASET DESCRIPTION

The dataset used for this paper is available at UCI Machine Learning repository as the [42] set and is extracted from the 1994 census database. The dataset consists of 15 attributes including income level. The dataset has 48842 records in it, which will be further split into training data and test data by following 75:25, 70/30 or 80/20 rule to apply prediction algorithms.

IV. LITERATURE REVIEW

This section will represent the related work done in the previous research papers which worked under the utility of algorithms performed using UCI repository to utilize Adult dataset [42] which formulates from an article in 1975 by Walter E.Simonson et al [25] on a DBMS for the U.S bureau of Census. To Begin with, the main element of this section focuses on the dynamic real-life usage as well as outcomes from the previous methodologies. The outputs on the adult census dataset from various different types of analytical techniques and models help to study different schemes and proportions of the dataset to be studied in a pattern which helps to produce a variety of results in factors such as accuracy, sensitivity or recall and specificity as well as other basic measures of diagnostic accuracy. The main targets of the algorithms, which cover the best output, depend upon housing information because it involves a direct or indirect relationship to the financial status of individuals. To be more depth study of this idea research in 1979 shows an estimator method known as Estimates of Income for Small Places: An Application of James-Stein Procedures to Census This estimator was applied to the places having a population of 1,000 or less. The estimates formed a basis for the allocation of revenues across various states [1][2][54].

In 1997 Kriger reflect usage of social and economic elements to be determined by using various strategies. The data is accumulated in various manners to support the socioeconomic position of the children and adulthood as well. This work intends to bridge between the economic conditions and the health aspects [3]. In 2017, S. Latif and Z. Z. Lecturer comes up with research on "Customer annual income prediction using a resampling approach," which helps to study the records in the prediction of demographic attributes. The domain of work done is to study and properly classify a large, highly dependent and complex dataset to predict customer income range from other demographic attributes [4].

Sometimes for a large number of highly correlated attributes makes the dataset suffering under evaluation phase confronting with over fitting problem. In order to improve this classification, performance resampling method can be utilized [4]. To evaluate and validate the models' various useful measures have been considered. The positive branching of this resampling approach was that it identifies the limitations of supervised classification problem for those datasets which consist of highly dependent feature vectors and incorrect class information [4]. A supervised learning framework by Zhong et al [5] was proposed to predict users' demographic attributes based on mobile data because demographic prediction plays an important role in

user profile modeling. In 2006, A. Lazar and R. Zaremba worked upon an Income Prediction Study for which implementation was done using Support Vector Machines Optimization. This approach shows increases in efficiency as well as helps to improve classification accuracy under a systematic analysis of the grid parameter search, training time, accuracy, and a number of support vectors. An accuracy value as high as 93% is allowed by proper identification of the relevant features to be obtained for specific problems against a test population while reducing the total computational. In addition to it, Tailoring computational methods around specific real data sets is critical in designing powerful algorithms [6][7]. Under which they described that various feature selection through principal component analysis is employed to increase the efficiency of support vector machine (SVM) methods [6][7].

Various research papers have used python in their research projects using data from Adult Census Income and have produced accuracy up to 88 % using the Xgboost and producing set of controls such as single cell estimate of the population 16+ for each state, controls for Hispanic origin by age and sex, as well as controls by race, age, gender. These controls are measured under the category of weights on the Current Population Survey (CPS), which is used to supplement census information between census years. These data types consist of a random sample of persons from the CPS, with information on wages and other characteristics of the workers, including sex, number of years of education, years of work experience, occupational status, a region of residence and union membership. This production of analysis is carried over by R programming for better analysis tools available and graphical representation on a visual scale but as such in this program under python the xgboost command is used as to regenerate and produce a powered version of accuracy by using all three sets of controls in the weighting program and "rake" through them 6 times so that by the end the result come back to all the controls used. CPS helps to retrieve population totals which is mentioned as estimate in the above paragraphs. This action is done by collection of socio-economical characteristics of people in formation of weighted tallies from CPS [61].

Moving further for this section a large number of datasets were studied and categorized which results in a good research work already done to be compared and learn to modify for future but as an output for conclusion is that 80% of papers were not being published in any good journals or made available online in form of research papers to validate the comparisons for real-time practical authentication of results. In fact, some researchers have definitely proven the accuracy to be better than other algorithms but they lack in a lot of other variable transformations that could create a novel idea for graphical representation or any new model generation on which the same program could not be reutilized again because they made the dataset to be program dependent. This situation leads to a condition where we can use only certain types of variables as to input otherwise the accuracy will decrease or the output will receive an error. Machine learning involves two major techniques, one is classification which is used to assign each dataset to predefined sets and another one is the prediction which is used to predict continuous valued function. The intention of categorization is to precisely calculate the objective class for every access in a certain data set. To create and assess income prediction data in light of the Current Population Survey given by the Census Bureau of U.S methodologies such as SVM and PCA [8] were utilized to increase the productivity and even enhance classification precision.

According to Y. Bengio [9] for information-rich situations networks are suitable connectors which are usually utilized for retrieving implanted facts in the structure of regulations, quantitative assessment of these regulations, clustering, self-organization, categorization and regression. They have an improvement, over other kinds of machine learning algorithms for scaling. An emphasis about the major category of categorization algorithm includes C4.5, k-nearest neighbor classifier, Naive Bayes, SVM, and IB3 was proposed in research by S.Archana et al [10] where they also explained a common survey of dissimilar categorization algorithms and benefits as well as their drawbacks. Researchers have used various machine learning models for predicting income levels enhancing the future scope utility of the results optimistically. Several Machine Learning Models like Logistic Regression, Stepwise Logistic Regression, Naive Bayes, Decision Trees, Extra Trees, k-Nearest Neighbor, SVM, Gradient Boosting and 6 configurations of Activated Neural Network were explored and analyzed by Chockalingam et. Al. [11] performed on the Adult dataset [42] which shows better results than the Principal Component Analysis (PCA) and Supports Vector Machine methods implemented in a research by A. Lazar [6][7] to generate and evaluate income prediction data based on the Current Population Survey provided by the U.S. Census Bureau.

A new approach using the Random Forest Classifier algorithm to predict income levels of individuals implemented as a decision trees model was presented in research by Bekena [12]. Where Topiwalla [13] with the results of 87.53% made the usage of complex algorithms like XGBOOST, Random Forest and stacking of models for prediction tasks under which Logistic Stack on XGBOOST and SVM Stack on Logistic were also included for scaling up the accuracy.

5. COMPARATIVE ANALYSIS

A comparative analysis shows the types of algorithms that were used by previous researchers to perform analysis on adult dataset [42]. The rows represent the methods used such as different types of algorithms performed as well as the columns represent the output in terms of accuracy, specificity, sensitivity and recall.

Table 1 : Comparison among various algorithms within specific output.

TYPES OF ALGORITHMS	ACCURACY (ARY)	SENSITIVITY (SPY) OR RECALL	PRECISION
Naive Bayes			
	81.20%[19]		
XGBOOST (Using Python)			
	87.53 [13]		
PCA and Support Vector Machine	84.92% [7]		
Gradient Boosting Classifier			
	86.29 % [11]		
Logistic Regression	85.11% [17]	88.118% [17]	92.83% [17]
Random Forest			
	82.27% [19]		
Decision Tree			
	83.89% [19]		
KNN			
	79.5% [11]		
ANN			
	85% [11]		

Table 2 : Feature Selection in compared research papers.

FEATURES USED	RESEARCH PAPER REFERENCE					
USE ALL FEATURES	[7]					[17]
REMOVE EDUCATION NUMBER		[11]	[13]			
REMOVE FINAL WEIGHT	[19]			[20]		
REMOVE NATIVE COUNTRY		[11]		[20]	[21]	
REMOVE RACE					[21]	

When it comes to classification tasks there exist several important features which were used to optimize the complexity in different machine learning models shown in research by Lemon et. al. [14]. On the other hand, Bayesian Networks, Lazy Classifier, Decision Tree Induction, and Rule-Based Learning Techniques were replicate by Deepajothi et. al. [15] to present a comparative analysis of the predictive performances for the Adult dataset [42]. In addition to it, Logistic Regression as the Statistical Modelling Tool and 4 different Machine Learning.
Techniques were presented in research by Haojun Zhu [16] which includes Neural Network, Classification and Regression Tree, Random Forest, and Support Vector Machine for predicting Income Level.

An article in 1976 by Jay-Louise et al [24] discuss about various problems under which it focus on the management of census data base on terminologies such as data description, acquisition and manipulation. To add to it Jay-louise [27] also publishes an article in 1975, which shows implementation strategies for manipulating functions on census data base. Moreover, an article in 2017 by Christian clausner et al. [26] explains a topic on “Unearthing the recent past: Digitizing and Understanding Statistical Information from Census Tables “which describes Censuses as a wealth of information on a national scale that allow governments and the public to have a detailed glimpse of how people live under geographical distribution and characteristics. In addition to underpinning socio-economic research, the study of historical Census statistics provides a unique opportunity to understand several characteristics in a country and its heritage. This helps in presenting an overview of a complete blueprint of the implemented preprocessing, recognition background, challenges and post-processing pipeline, as well as the information-rich results obtained through a pilot digitization project on the 1961 Census of England and Wales which helped in portray the resulting methodology to be used for digitizing and understanding tabular information in a large variety of application scenarios [26].

As Artificial neural network (ANN) is trending to explore and automatize the power of machine learning to learn datasets and help in making strategic decisions based on hidden information from existing datasets. To validate this statement a research published in 2010 showing Revenue prediction Using Artificial Neural Network was proposed by Christine Sanjaya et al. [27] which was conducted based on the following phases: business and data understanding, data preparation, modeling, evaluation and deployment. Techniques such as benchmarking regression algorithms are also used by some researchers for predicting income of customers from banks, which involves various methodologies such as regression algorithms which are most commonly used for example CART, ANN, MARS, LS-SVM and other regression models that include beta regression, robust regression, ridge regression as well as multiple techniques that are being combined together under two-stage models [62][28]. However, the digital sharing economy has introduced opportunities for economic

growth, productivity, and technological innovation. However, the adoption of sharing economy applications may be inaccessible to certain demographics, including older adults, low-income adults, and individuals who are not college educated. This factor leads researchers to investigate how the demographic factors: trust, computer self-efficacy, and perceived ease of use, impact participation in the sharing economy [29]. In 2018, Douglas J.Kennard [30] helped to produce an effort, which improves both volunteer participation and transcription accuracy on utilizing his research conducted on Computer-Assisted Crowd Transcription of the U.S. Census with Personalized Assignments for Better Accuracy and Participation [30].

6. BASIC CONCEPTS AND TERMINOLOGIES

To begin with a basic description of machine learning concepts is necessary to understand methods followed to analyze and predict information. Machine learning and predictive analytics are synonymous and constitute the 'modeling' portion of this research. A discussion of machine learning as a concept and the tools used are discussed further.

Machine Learning: "Machine learning is a method of data analysis that automates analytical model building. Using algorithms that iteratively learn from data, machine learning allows computers to find hidden insights without being explicitly programmed where to look."[22] Machine learning can create algorithms for adapting knowledge from past data and makes predictions that are accurate. The general idea behind most machine learning solutions is that a computer learns to perform a task by studying a training set of examples. The computer then repeats the task with fresh data that it has not trained on before. [23] Algorithms can add newly predicted information to their training data and learn from this updated data, thus modifying the model itself without human intervention. This feature is important because some tasks can be difficult to define except through the use of examples (for e.g., grouping viewers based on movies they have watched in the past, and predicting what they may prefer in future)[51].

Various methodologies for analytics defined as following:

Predictive Analysis: In the predictive analysis, data is fed into any of the statistical algorithms by performing the steps like data cleaning and transformation, in order to predict some meaningful insights from the data. The prediction algorithms include regression, support vector machines, multiclass classification, etc.

Neural Network: It works on the basis of the connections in neurons that logically represent the working of the human brain. The data values are represented by the neurons and the link is shown by synapses in this process. A layer architecture, where the input is fed into the first layer, then there are middle layers also called hidden layers which perform operations, and then the final output layer. Other than these two layers, the model also has intermediate layers called hidden layers. Weights are assigned to each layer and with multiple iterations; weights are re-calculated to find the optimal values of those weights that fit the function [51][33].

Support Vector Machine: A Support Vector Machine (SVM) is a discriminative classifier help to distinguish between data types by a separating hyper plane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyper plane, which categorizes new types of resulting utilities. In two-dimensional space this hyper plane is a line dividing a plane in two parts where in each class lay in either side [34][51].

Linear Regression: Its working is based on the assumption of hypothesis which itself is a linear function and its goal is to find the optimal parameters for the assumed function, that fits the data. It works using the linear function, so the graph obtained by performing this is usually a straight-line graph.

Logistic Regression: Like all regression analyses, the logistic regression is a predictive analysis. It is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables [35][51].

Decision Trees: A structure that uses a branching method to represent every possible output result of a problem in context of a decision is called a decision tree. It starts from the root node, which splits into further nodes and leaf nodes (the branch which further does not split). Leaf node represents the end output of the branch with particular inputs. It represents the relationship between different features clearly and easy to understand their importance upon each other resulting in various factors upon different possibilities [51].

Random Forest: It is a supervised learning algorithm. A more defined version of Decision Trees comprises of the forest in this most of the time trained with the "bagging" method, a combination of learning models that enhances the end results is presented. It helps to modify multiple decision trees to a single set of database in a stable architecture and completely merges them for higher accurate performance [51].

AdaBoost: It is a general ensemble method, which creates a automatic classifier collection upon the basis to develop a stronger network in a single classifier version from a number of weak classifiers. It is generally used to boost the performance of decision trees on binary classification problems [53][54].

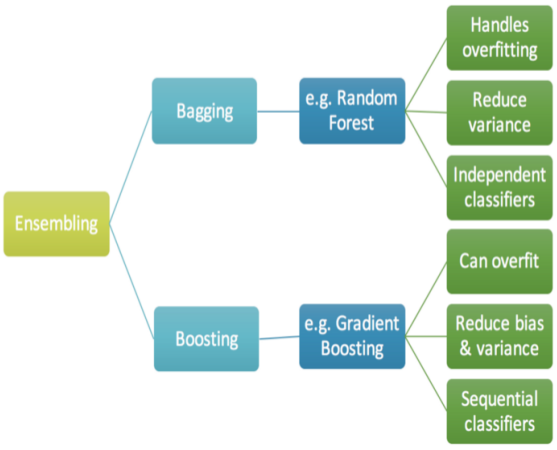


Fig 1.1 Types of boosting based on different problems in the dataset [58].

AdaBoost helps to improve the performance of any machine-learning algorithm to boost the results using more iterations and features. It is best used with weak learners which achieve accuracy using in case of adaboost slightly more than compared to the expected versions from other algorithms [53][54].

Naive Bayes: Naive Bayes classifiers are a collection of classification algorithms that use Bayes Theorem for its application. It does not comprise a single algorithm but a family of algorithms where all of them share a common assumption, i.e. every pair of predictors being classified is independent of each other. It means that it assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. It is one of the easy methods to predict the class of test dataset. It performs well in multi-class classification.

Nearest Neighbor: K-nearest neighbour can be used for both classification and some types of regression problems as well. It basically works by predicting the k (any numerical value) neighbours of a particular input value. It firstly assigns the input into a group, and with the number of increasing iterations, it improves the assignment of data values into a group, basically finding the k nearest neighbours of any particular input [36].

Gradient Boosting Machines: It comprises of statistical framework called Adaptive Reweighting and combining algorithms helps to recast AdaBoost and related algorithms on a higher scale. This structure on a larger scale with quantitative and qualitative concept is called gradient boosting or gradient tree boosting. It usually works using three elements, to optimize the loss function, a weak learner to make the prediction, and to minimize the loss when using an additive model. Stochastic gradient boosting is very commonly used which at sudden point draws subsample at each iteration in context of the training set, and this random subsample is used to fit the base learner.

VII. PROBLEM STATEMENT

Machine learning techniques to identify the likelihood of future outcomes based on historical data with the use of data modeling and statistical algorithms are called Predictive analytics. The aim is beyond knowing what has happened to providing a best result of what will happen in the future. Over the past few years, Predictive analytics has become much more vital in all aspects from business to government sectors and many more. It helps banks in approving credit card or identifying risk factors, Security issues, suspicious activities, e-mail providers in filtering spam, as well as retail marketing in predicting customers' likelihood to churn out or purchase products. But predictive analytics is a complex multi tasking process, and therefore implementing in real time gets complicated if the user lacks under analytical as well as statistical background making it more challenging [37][38][63]. Due to larger datasets as well as more features in a data with missing values the traditional approach for predictive analysis is very hard to perform the same actions due to less flexibility in the user interface model or updates in the library packages in various languages such as r and python that ultimately taking more time and still getting less accurate results in traditional methodology [37][38][63]

Common Predictive Analytics Challenges are Expertise, Adoption, and Empowering End Users [38]. This is because for predictive analysis as well as data mining under machine learning there is a need for professionals who have a deep understanding of r programing as well as python language within the concepts of how and where to use and best fitting of models which is very hard to get without experience which makes expertise act as a gap. In case of adoption, It is very hard to just in time adopt a new technology by learning it from scratch—and predictive analytics solutions are rapidly changing with the day by day changes in technology and data types in real time making it difficult to get a consistency in learning as well as practicing. This is because they typically live as limited tools, which means users have to switch from their main interest of technical; background in order to improve any sort of quantitative or qualitative concept in their business [37][38].

To add to it, traditional predictive tools are complex to scale and deploy, which makes new versions and improvements in future a tough effort. As a solution to this part, Predictive analytics is most effective when it's embedded inside the applications, which are commonly used under daily life like Microsoft office tools. Inbuilt mechanism of deep learning and AI inside an application gives a high range advantage over the competition—and gives the end users a strategic advantage for their businesses[37][38][54]. On the Other hand, The information gathered under the predictive analysis is always hidden under layers of statistical as well as programming backgrounds which forms a bridge gap between the user as

well as the data scientist performing actions from which the end user is unable to make sudden actions and decisions on with their own prospective this makes it hard to empowering end users [38]. As discussed above, if users want to act on the data, they have to jump to yet another application, ultimately wasting time and interrupting their workflow. The data analysts require performing a list of steps over any predictive classification problem. But the major obstacle is that for every update and release, the whole steps gets more complex in context to the implementation procedure along with different types of datasets and methodologies to be performed by manipulating certain attributes [37][38].

These steps include the following pattern [37][38][54][64].

1. Preparation of data information
2. Cleaning of dataset
3. Identifying important features
4. Recognizing and reordering correlations between features
5. Understanding how different algorithmic statistics work on different features
6. Choosing the right algorithm for the right problem
7. Deciding the right specifications for the algorithm
8. To be sure that the data format is correct
9. Understanding the results of the algorithm iterations
10. Re-training the algorithm with new dataset
11. Treatment with imbalanced data
12. Deploying/re-deploying the algorithmic model
13. Predicting in real time/batch
14. User action can initialized at the time of predictive analysis embedding with the application to carry the data building blocks for insights acknowledged, which are integrated with the primary application [54][64].

The real time models are built to improve in the assessment of the economic as well as financial consequences associated with the issues that are related to uncertainty or missing data values. This requires understanding the past situations of the results from older datasets to study how the newer versions of dataset with new problems will be solved and provide a validation to solution that rebuilds the prediction model to provide as accurate results as possible. [64]

This paper contributes to similar type of issues in the specific context of income modeling in terms of prediction. In this research paper, the main focus is to predict the income level of a person from the adult census dataset. The study will help us to learn and train from the past dataset inputs of financial as well as other attributes to predict the future trends of financial output in terms of potential scale accuracy of predicting the future income of a person is above or below a certain limit.

The challenges for this paper are to perform Analysis of Census Data to determine certain trends. The Prediction task is to determine whether a person makes less than or over 50K a year. The implementation part involves to Analyze the accuracy and run time of different machine learning algorithms to identify how salary is affected by Different demographics. To help Uncover the socio-economic factors such as (age, workclass, education, marital status, occupation, native country, etc.) affecting high income (separate causes from consequences and confounders).

The Data Set has 48842 records with non identified, missing, duplicate and conflicting instances to be removed for further analysis in data cleaning process. The data set involves 15 attributes having both continuous and discreet-valued. The connection between the numerical and the categorical values such as the age and the income respectively makes a gap to study the dataset analysis onto regression algorithms such as liner regression as well as logistic regression because they process one type of data variable to be numerical or categorical which makes it harder to study consuming more time and increasing error rate output as less accuracy. Some problems from the previous papers studied in the past involves a concept of Feature scaling which helps to visualize and manipulate algorithms on the dataset results positively while using certain algorithms and have a minimal or no effect in others. To understand this a procedure is to be followed that helps to learn selecting from a variety of scaling methods and when we should scale features [39].

Most of the times, the dataset will contain features highly varying in data types, units and range. But since, most of the machine learning algorithms use Euclidean distance between two data points in their computations for which it acts as a problem. If left alone, these algorithms only take in the magnitude of features neglecting the units. The results would vary greatly between different units for example 5kg and 5000gms [39].

The distance will be changed with low to high magnitude features. Therefore in some cases, to decrease this effect, all features require to deal with the same level of magnitudes by the scaling method. In this research paper, the database used to analyze prediction modeling called “adult census dataset” includes a lot of missing values in some rows and columns which includes a number of less significant attributes that does not contribute in utility functioning of the model to train or test features as per different functions carried over by algorithms described in the methodologies used for a systematic analytical procedure.

VIII. GAP ANALYSIS

In previous research papers the major focus is given to find out the accuracy to predict the results for whether the income level of an individual is less or more than $50k. Other than this, there is a very minimal amount

of research done for what factors contribute to this change in results. Some researches taken over the results using 14 attributes and in some they used 15 attributes after cleaning or preprocessing over which the best results were given when 2 attributes out of 16 are removed that have no utility in manipulating any changes to the data analysis. There is no doubt between the result outcomes crossing a highest to around 88-90% with an ROC curve value near to 1 but still it does not provide essential contribution for future data scientists to utilize their results in efficient manner. So here is a proposed structure, which will help to learn from this research paper by answering all the research questions gathered from the study done under literature review. In further study, we will discuss about the hypothesis to be involved for selecting the overall possibilities in the proposed solution plan.

This paper involves the study to find the solution for the following problem statements:

1. Does education play a major role in salary and what is the minimum level of education needed to ensure a high salary?
2. Will marital status affect the salary of a person?
3. With all other factors being the same will sex of a person determine him/her getting a higher salary?
4. Will the age of a person play a significant role in defining his salary?
5. Will the race of a person be significant factor in defining his salary?

IX. PROPOSED SOLUTION PLAN

Various hidden questions were remain unanswered due to lack of methods and algorithmic statistics involved in dealing with ambiguous and missing values in a dataset that could solve to answer various concepts which could be helpful for future researchers to solve new problems of the same kind in context to prediction improvement from learning the mistakes done in previous researches. As the technology has been advanced in several methods and deep learning as well as Artificial intelligence using machine-learning methods [65]. This paper includes all the solutions to those problem statements which were not answered before using different methods as well as visualizations which validates the solutions to a large scale further verified by accuracy results achieved in this paper. One of the most important part of providing information from census data is predicting data like income on the basis of different demographics and then providing solutions upon which factors the income gets vary and to show which factors influence the income of a person the most [65]. Cleaning and preparing the data are one of the most vital components of Machine learning that should be deal under the classifiers for the model to train [65]. The different techniques in the field of analysis will be discussed further such as preprocessing the data for each classifier, training the model and the performance evaluation [54][65]. A number of classifiers helps to classify and reveal the background hidden information behind any data on the basis of landmarks based upon previous benchmark made and already existing data [16]. In order to learn, how well someone can predict whether an individual's annual income exceeds $50,000 or not using the set of variables in this data set [16][40]. The question is retrieved by following in two separate methodologies – traditional statistical modeling and machine learning techniques [16]. For binary outcome a statistical modeling tool called logistic regression is used. Other machine-learning techniques such as – Neural network, classification and regression techniques, random forest, and support vector machine, are also used to answer the same question [16]. There exists a possibility that these learning methods may not always work on each type of data set or may not be able to perform properly under the context of statistical modeling [40][16]. In this research the main focus tends to do a comparative study of these different commonly used machine learning classifiers like Decision Tree, Naïve Bayes, Random Forest, Logistic regression and Support Vector machine as well as Neural Networks. To Benefit for selection of the most accurate machine learning algorithm in order to derive a solution for the problems of classification and prediction of information for future outputs as the most important concept of machine learning which depends on the data type as well as different datasets to be used [65].

So for higher efficiency and effectiveness of system satisfaction in context of accuracy achieved will be done by comparing these classifiers by their different evaluation measures like Confusion matrix, ROC curve and graphical visual plotting the features used in results [54][65].

Data Partitioning will be done under which data will be divided into Training Set in 75% and the Validation Set in 25%. The further process will go step-by-step structure work to be done to including Data Cleaning, Data Transformations, Dataset Preprocessing and Exporting Transformed Datasets. For better understanding of variables to determine their functions and their importance in the dataset, some attributes of the dataset explained as follows:

Age = the age of the individual in years, workclass = the classification of the individual's working status (does the person work for the federal government, work for the local government, work without pay, and so on) education = the level of education of the individual (e.g., 5th-6th grade, high school graduate, PhD, so on), marital status = the marital status of the individual, occupation = the type of work the individual does (e.g., administrative/clerical work, farming/fishing, sales and so on), relationship = relationship of individual to his/her household, capital gain = the capital gains of the individual in 1994 (from selling an asset such as a stock or bond for more than the original purchase price), capital loss = the capital losses of the individual in 1994 (from selling an asset such as a stock or bond for less than the original purchase price), hoursperweek = the number of hours the individual works per week, native country = the native country of the individual, over50k = whether or not the individual earned more than $50,000 in 1994 [42][41].

Table 3. The following table provides details about the 14 attributes that will be used to train and test the outcome (outcome is ≤ $50000 or > $50000) [41][42][45].

No.	DATA SET	DATA TYPE	DATA SIZE
1	Age	Ordinal attribute denoting age of the person	Range from 17 to 90
2	Workclass	Nominal attribute denoting the working class of a person with values	Private, Selfemp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
3	. Fnlwgt :	Ordinal attribute denoting the survey weight	Range from 12285 to 1490400
4	Education	Nominal attribute denoting the education level of a person with values	Bachelors, Some-college, Masters, HS-grad, Prof-school, Assoc-acdm, Assoc-vo 1st-4th, 5th-6th 7th-8th, 9th, 11th, 12th, 10th, Doctorate, Preschool.[45]
5	Education-num	Ordinal attribute denoting the educational number given to each education level -	Range from 1 to 16.
6	. Marital-status	Nominal attribute denoting the marital status of a person with values	Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AFspouse.
7	Occupation	Nominal attribute denoting occupation level of a person with values	ech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty Handlers-cleaners, Machine-opinspct, Adm-clerical, Farming-fishing, Transport moving, Priv-house-serv, Protective-serv, ArmedForces.[45]
8	Relationship	Nominal attribute denoting the relationship status of a person with values	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
9	Race	Nominal attribute denoting race of a person with values	White, Asian-Pac-Islander, AmerIndian-Eskimo, Other, Black.
10	Sex	Nominal attribute having two values	male or female.
11	Capital-gain	Ordinal attribute denoting the capital gain contributed by the person	Range from 0 to 99999
12	Capital-loss	Ordinal attribute denoting the capital loss caused by the person	Range from 0 to 4356
13	Hours-per-week	Ordinal attribute denoting the total number of hours worked per week by the person	Range from 1 to 99
14	Native-country	Nominal attribute denoting the country of origin of the person with values	United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVIetc), India, Japan, Greece, South, China, Cuba, Iran, Honduras Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad &Tobago, Peru, Hong, Holand-Netherlands.[45]
15	Income	Nominal attribute having two values	> 50k , < 50k

Listing of continuous attributes from the data set :

- Class: >50K, <=50K.
- Age: continuous.
- Fnlwgt: continuous.
- Education-num: continuous.
- Capital-gain: continuous.
- Capital-loss: continuous.
- Hours-per-week: continuous.

Preprocessing-Variables in data includes:

1. MISSING VALUES
- Removed missing values which were denoted by "?"
- Na.omit() will be used to remove these rows.

2. DATA MODIFICATION
- Removed less significant columns ("fnlwgt" & "education_num")
- Data Binning-Grouping Multiple Categories into lesser number of "bins"

X. RESEARCH WORKFLOW [54][45]

A. **Data Description-** It includes the features which are dependent and independent to learn and describe about how much influence could each feature has upon the income prediction as well as defined nature of each feature to understand how well they are inter connected to each other [45].

B. **To Load and program the data to run in the R studio -** Downloading the data directly from the source packed in a csv file and then running on the r studio platform[45].

C. **Data cleaning-** Removing all the biased or missing values as well as ambiguous information, which manipulates the data resulting in false reading accuracy [45].

D. **Explore the different types of dependent as well as independent variables –** A vital stage on which the variable dependency has to be explored in order to check that the newer version of features are interlinked as well as which features has direct influence or indirect influence through other connectivity with co-features in the dataset.[45]

E. **Using training data to initialize a prediction structure model layout (75%) –** A phase on which the weak or the strong classifier have been regulated into a singular manner of variable which helps them to inact in a formation of learning process over which the system uses classifiers to learn from the repeating as well as different non repetitive structural relationships between patterns of features and their respective values which further is stored and performed over the testing phase using various algorithms [45].

F. **To perform actions on testing data which in this case is 25% -** The technique is used on the test dataset on a scale of various models over which it helps the system to perform the prediction tasks learn during the training phase on the data which is new or not processed to predict an output matching the patterns from the comparison as well as during various iterations to run the tasks and provide various accuracy on testing models showing how accurate the results fit the prediction for future[45].

These steps will follow a general roadmap for the solution to be obtained in this research[54][66]:

1. Getting the adult dataset [42] in csv format from the uci repository and storing on local disk.
2. Loading the file – redirecting the working directory in the rstudio to set the boot location for the adult datset in csv format.
3. Data cleaning under exploratory data analysis[66]
   - Removing the missing values.
   - Deleting the non-valid information.
   - Improving the correct type of data values in their respective order. For example age should be in digits – continuous variable as well as cannot be 0 or xyz.
4. Statistical analysis [54][66]
   - To get the model select within use of different feature with a visual impact of which features to be selected in which sort of algorithm
   - To achieve a basic aim in each algorithm with same sort of design ultimately resulting into a target variable which acts as an solution output for each model
5. Manipulating categorical features – because some of the features are not required into the dataset that has least or no influence on the results it is mandatory to use as less features as possible by making corr plots that enables to understand the nature distinguishing between different variable categories[66].
6. Evaluation, Validation as well as verification
   - The results are evaluated using confusion matrix, which helps to get the accuracy, precision, specificity as well as sensitivity with the f-score of the model [66].
7. Data Visualization
   - To show relationship between different features to understand the dataset clearly.
   - Plotting the results to represent the roc curve helps to know which model suits the best [66].

XI. (PROPOSED SOLUTION)

The adult dataset [42] also known as "Census Income" retrived from the 1994 US Census database. UCI Machine Learning Repository [42] has all the detailed structure of the data set within acknowledgement from the papers which have cited the dataset for similar types of researches using US census database [42]. For this research the main aim

was fcus on knowing the hidden information about the demographics within the objective to score a high accuracy in finding a best model which helps to predict that if a person earns more or less than $50,000 using the set of features in this source data [16][42][11][21].

The ultimate goal of this research is to conclude whether a person earns an yearly income to be less or more than 50,000 US dollars and to find which model best suit the this type of dataset problem. This goal is retrieved by the following steps: To begin with, the trained dataset was used to learning process for making a informational layout to train the machine to predict whether a person makes over 50,000$ a year [67]. If data mining algorithm is not performing well to solve missing values as well as binding up with several outliers, then initialize changes in iterations as well as loop wholes by manipulating R program code to clean missing values using na.omit() and outliers. Moving further to next step, the testing dataset used to test whether the model which has been trained by feeding information is able to actually predict if a person makes over 50,000 $ a year[67]. The last step is the validation as well as the experimental evaluation plan under which a comparison of accuracy from several data mining algorithms is done that results as the output from confusion matrix achieved in each model that was utilized[67]. Last but not the least, compare and track the complexity of several implemented data mining algorithms and by the roc curve showing area under the curve value to be nearest reach to 1 by any model as the total to be consider decides which model is best suited for the dataset problem [16][11][21][67].

1. UNDERSTANDING THE DATA SET

Specifications of the Dataset

To predict if a person has an income of more than 50,000 US dollars per year we have used the “adult” dataset
Dataset contains [42][67]:

1. There are 48842 records in total under which we will use 36,631 records for training the data values, and the remaining 12210 for testing data values[67].
2. For a total of 14 features in the dataset 6 features have a continuous numerical data type whereas the remaining 8 features show a nominal nature[67].
3. The objective divided under 2 classes for earning >50k as well as <50k.
4. There were approximately 7.461% missing values in the dataset.

Dataset Training

The independent instances helps to form the training data which works best when the dataset is larger to explore more outputs learn by the models in terms of classifiers. The dataset was slipt into 75:25 under which the 75% is used for training set which holds 36,631 number of records[67].

Dataset Testing

The independent instances with no role or no use in learning process of models. This works best when larger dataset trained to perform on a larger set of test data information results to measure errors more precisely. From the total of data one fourth of the data that is 25% of the domain dataset was utilized as test data containing 12210 records [67].

2. IMPORTING THE DATASET

The dataset can be used by a direct url link from the UCI repository[42]. In this research the dataset was loaded from a local machine. The hardware and Software requirements for the implementation were: Apple MacBook pro i-7 processor 4GB RAM , 512GB HDD
And Rstudio Version 1.1.456.

Algorithm used to import dataset :

```
1. LOAD THE FILE USING
 file_name="/Users/apple/Downloads/adult.csv"
2. READ THE DATA IN CSV FORMAT
adult.data<- read.table(file = file_name, header = TRUE,
sep = ",",strip.white = TRUE, stringsAsFactors = TRUE,
3. CALLING THE FEATURES FROM DATASET
col.names=c("age","workclass","fnlwgt","education","educ
ationnum","maritalstatus","occupation","relationship","race
","sex","capitalgain","capitalloss""hoursperweek","nativeco
untry","income")).
```

Rstudio comes with previous various preinstalled package libraries but for this research we used a lot of visual model tools and algorithms which helps to present the dataset into more interactive manner making it easier to understand for analyzing. We have to install some certain type of Package libraries, which are presented as following:

```
R Packages Used:

# library(e1071), library(ggplot2), library(reshape2), #
library(randomForest), library(rpart), library(rattle),
library(corrplot), library(gridExtra), library(MASS),
library(gbm), library(glmnet), library(nnet), library(class),
# library(xtable), library(parallelSVM), library(FastKNN),
library(caret). And library(tiktok) for time complexity.
```

After loading the dataset as shown in the previous steps data description is given which shows us the loaded version of dataset to attain a summary of data and to get a deep detailed information about the dataset.

3. MISSING VALUES AND DATA CLEANSING

The plot of the response variable shows that yearly income is imbalanced. There are far more individuals (76%) with an yearly income less than or equal to 50K that those (24%) with an yearly income more than 50K.The dataset contains 48842 observations and 16 columns

[16][11][42]. This source of dataset is not balanced properly as out of these 48842 observations, 76% of the individuals have an income ≤ 50k while only 24% individuals have income level <50k which results in accuracy to be biased over the fitting issue to manipulate rearrangement of the sensitivity as well as specificity in results. Fixing the imbalanced data depends on the objective of the analysis. For example, in the banking sector while use of credit cards, it may be more important to correctly identify the customers who will default rather than those who will not default [43].

This imbalanced issue can be resolved by moving data-sets into a balance structure bridge gap value between them, either by oversampling instances of the minority class or under sampling instances of the majority class. This allows us to create a balanced data set that should not lead to classifiers biased toward one class or the other. Oversampling the minority can lead to model over fitting, since it will introduce duplicate instances, drawing from a pool of instances that is already small. Similarly, under sampling the majority can end up leaving out important instances that provide important differences between the two classes [43].

Assumption: For the basic source of this data analysis, we keep the distribution of majority and minority class as it is. The dataset has some missing values but some fields shown "?" in their respective fields [31]. These can be considered equivalent to missing values. Further deep diving to check which fields have these missing values.1465-missing values for column workclass. 1471 missing values for column occupation (All the 1465 missing values in workclasss overlap with occupation). 486 missing values for column native country. Nearly 23 of these missing values overlap with workclasss and occupation. Overall there are around roughly 7.461% (7.5 %approx.) missing observations from the Clean_train 30162 observations.

By performance of clustering methodology using na.omit function as well as classification architecture the missing values were removed for future training on the dataset model. This is important because maybe there exists a chance that the valid information may be neglected in case of deleting the whole row or column in the total dataset so to avoid decrease in number of instances the best case values were filled by seed value and the values which are worst case were dropped. Hence, those missing observations were deleted that have a manipulative effect on the model in terms of predicting better accuracy as well as performance actions.

Algorithm used for missing values and set seed

1. check for missing values col Sums(is.na(adult.data))

2. Missing values in, education(11077) occupation(4066) and native.country(20) str(adult.data)
3. Missing data imputation str(adult.data) , generally it is advisable not to impute the categorical missing values, if they are less than they should be removed.
4. levels(adult.data$income)<-list(less50K=c("<=50K"), gr50K=c(">50K")) library(VIM)aggr_plot < aggr(adult.data, col=c('orange','purple'), numbers=TRUE, sortVars=TRUE, labels=names(adult.data), cex.axis=.7, gap=3, ylab=c("Histo ram of missing data","Pattern")) library(missForest)imputdata <- missForest(adult.data)
5. check imputed values imputdata$ximp
6. Assign imputed values to a data frame adult.cmplt< imputdata$ximp df.master<- adult.cmplt
7. save a copy set.seed(1234) ratio = sample(1:nrow(adult.cmplt), size = 0.25*nrow(adult.cmplt)) test = adult.cmplt[ratio,]
8. Test dataset 25% of total train = adult.cmplt[-ratio,]
9. Train dataset 75% of total dim(train) dim(test) str(train)

4. DATA VISUALISATION

Various plots and graphs are used to represent the correlation between the various attributes and to get an idea of the overall pattern of dataset into a pie chart, bar graph, histogram as well as box plot display output. This will help to get an idea of which algorithm is suitable for certain type of scattering of data as well as which relationships between various attributes are strong and weak so as to decide in the elimination step for what attributes should be removed to get a better accuracy and attain a best model.

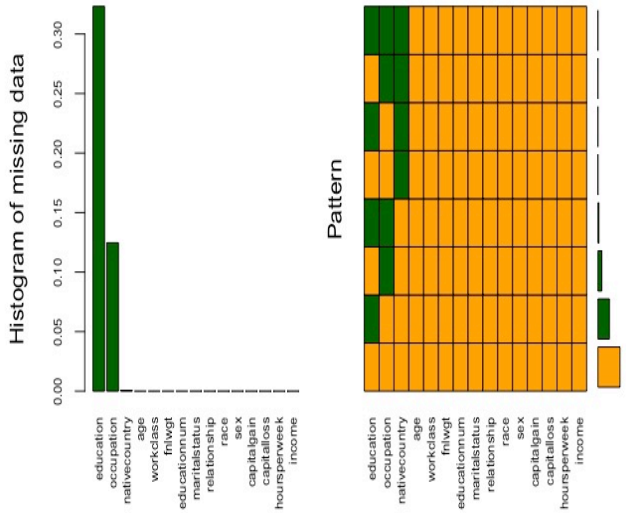


Fig- 2.1 Histogram showing missing values in dataset

Visualization is the most vital step in initializing a dataset because the overall layout helps the analyst to develop a blueprint of biased weight outputs , continuous as well as numerical value feature selection. This helps because when the dataset is divided into different types of values which further have a sub category of division will ultimately gets complex when performing logistic or liner regression algorithms. Therefore to overview the quality of dataset heat maps and various correlation plots and graphs are essential to the steps followed by feature selection.

The variables which have a higher effective relationship between their data type values are represented in this section of research paper which helps to develop a clear layout from the corr plot function making a visual layout showing a Correlation matrix which helps to calculate the relationship amongst two features in terms of weaker or stronger. In some cases the problems for collinearity or multicollinearity may occur due to effect of one relationship factor on the other attribute (both variables can be used to predict each other) that when we count a prediction model using one feature amongst any pair , the model gains the ability to predict the other[44][68].

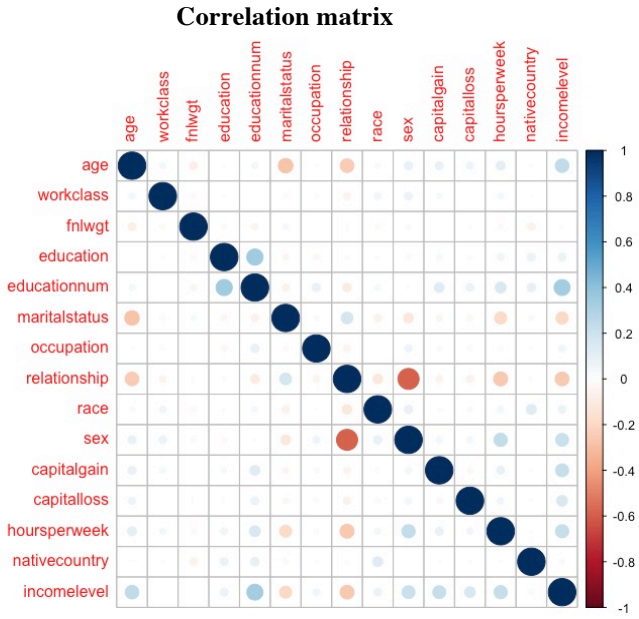


fig-2.2 weak and strong Relationship among attributes

Therefore, its is important to treat collinearity problem. Let us now check, if our data has this problem or not. Again, it is important to note that correlation works only for continuous variables [44]. We can calculate the correlations by using the cor() as shown;

```
1.cor(clean_train[sapply(clean_train,function(x)
!is.factor(x))])
2. X = data.matrix(train) corr_matrix = cor(X)
corrplot(corr_matrix) corrplot(corr_matrix, method =
'ellipse', type = "full")
```

it's proved clearly that none of the predictors are highly correlated to each other which moves to the next stage prediction modeling for the pre processing phase.

R code pattern for pie charts

```
1.p1<_ggplot(clean_train,aes(x=factor(1),
2.fill=maritalstatus))+geom_bar(aes(y=100*(..count..)/su
m(..count..)), width = 1) + coord_polar(theta="y") +
theme(axis.text.y = element_blank(), axis.ticks.y =
element_blank(), legend.title=element_blank()) +
xlab("") + ylab("") + ggtitle("ATTRIBUTE NAME
"XYZ"")
```

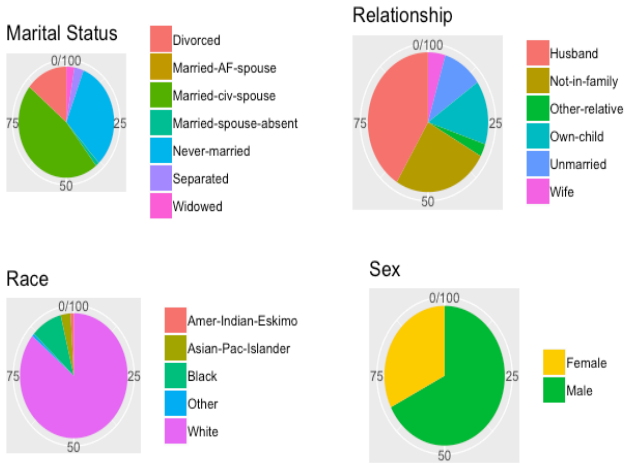


fig –2.3 showing pie chart distribution of dataset

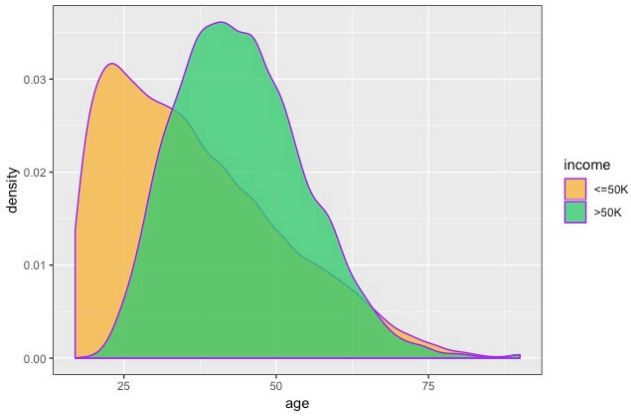


fig –2.4 showing age density distribution of dataset.

The graphs shows age from 25 to 75 and above with a more age group people from 25-40 and less people in age group of 70 and above. We can fix the outliers depending on the business/problem objective. Outliers may be due to random variation or may indicate statistically interesting. One of the slowest but effective way would be to go back the data source to find the underlying reasons for the anomaly [31]. In this case, without any prior information about the population builts a gap which ultimately results in using all these values for 75% ratio for training and remaining 25% testing. The numerical features show very low correlation with each other shown by the corr plot.

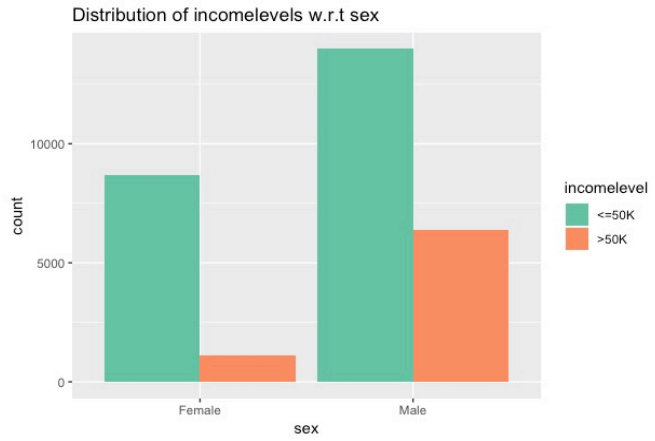


Fig 2.5 Showing distribution of income w.r.t sex

Females show a lower proportion of individuals in the >50K income level compared to that in males

Significance Testing of Continuous Variables

Every attribute is classified for disturbances in scattered information on graphs, distribution of data types , variance between the components such as features which are ambiguous or non-relevant , and predictability that is the target variable to output[45]. After the data cleaning part is done we began to look for which algorithm would be the best suitable for getting the highest roc and accuracy value. Also, because the comparative analysis from previous research papers used GBM has the highest accuracy [45].
So according to it we set feature selection, a non parametric model known as boosting is used because of it's nature that has no requirement pattern for distribution of information statistically. In this research, the skewness or the asymmetry is not being notified for any changes done under the variables. To add to it, the major focus is given to the effectiveness of each variable that will help us to understand it's predictable nature.[13][45][31][40].

A great distribution of value points is being notified under the "Age" feature which follows a broad range of deviations proving that it helps in predicting the objective classes for person earning less or more than 50,000 US dollars per year[45].

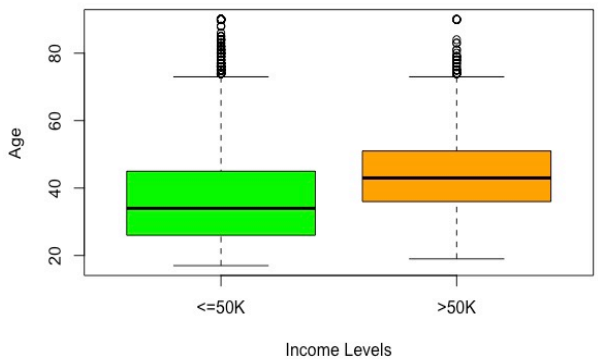


fig 2.6 Age distribution for different income levels

The "EducationNum" driving the category to represent that years of education also contributes to predict that higher the education of an individual , more is the predictability of the model to show that his/her earning is more. [13][45][31][40].

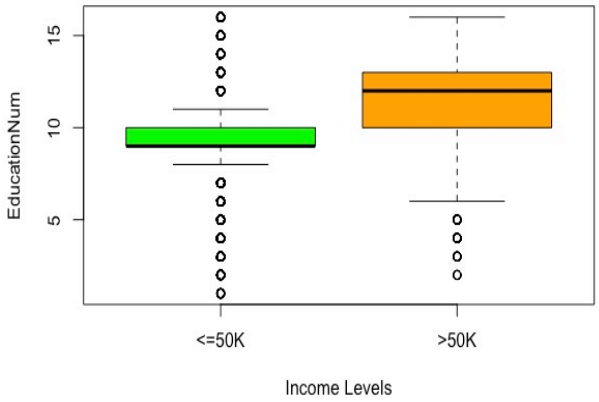


fig 2.7 Highest Level of education distribution for different income level

The graph shows hours per week playing a vital role in understanding that whether a person with more income working on a less hours can be compared with a person of low pay per hour income but working more hours has earning more or less than 50,000 US dollars.

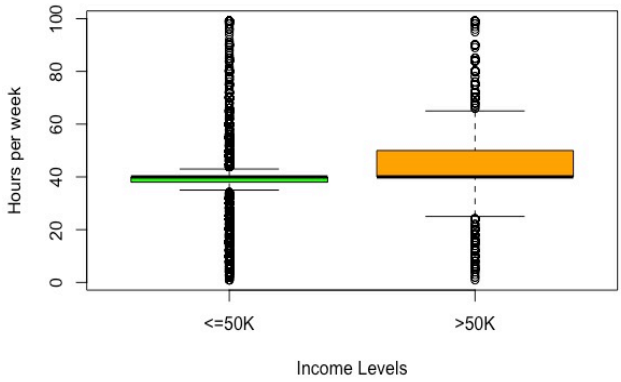


fig 2.8 Hours per week distribution for different levels

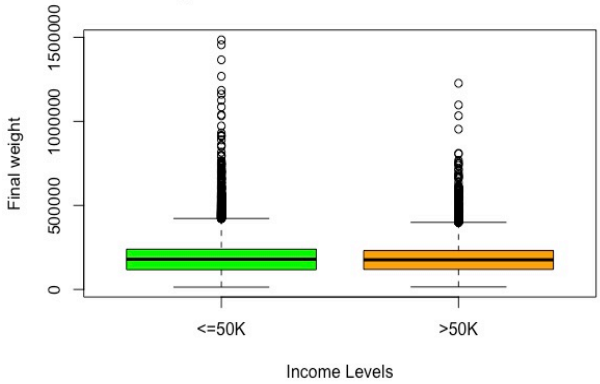


Fig 2.9 Final weight distribution for different income levels

The capital gain and capital loss variables do not show much variance for all income levels from the plots below. However, the means show a difference for the different levels of income. So these variables can be used for prediction[13][45][31][40].

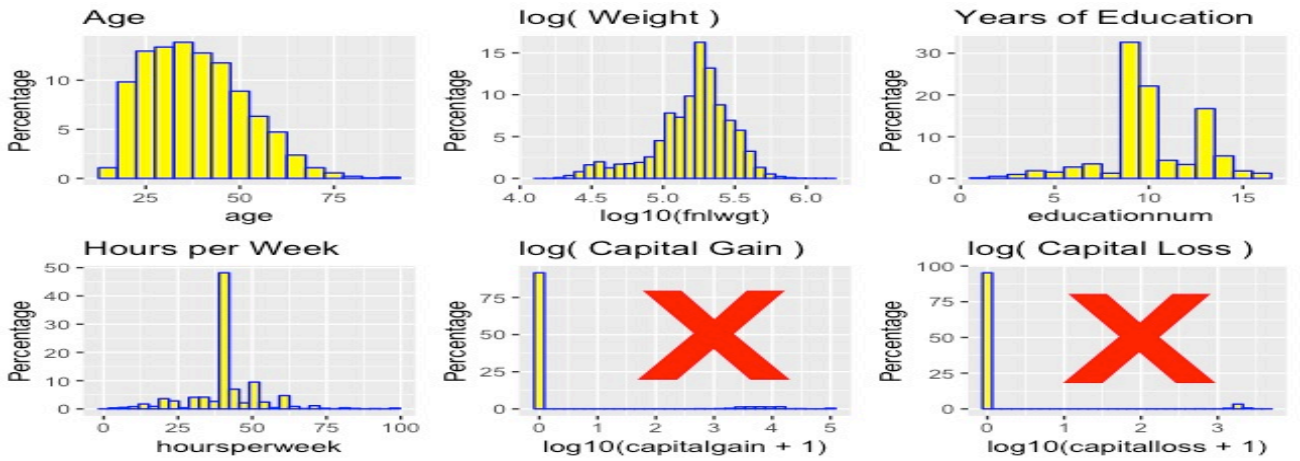


Fig 3.0 Visual comparison of different variables such as Age, fnlwgt, education-number, Hours per week , capital gain & capital loss

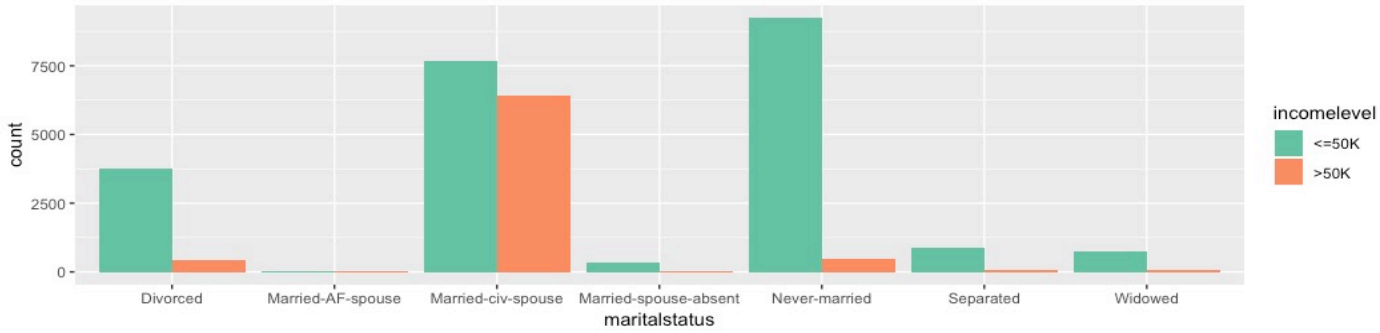


Fig 3.1 Distribution of income w.r.t marital status

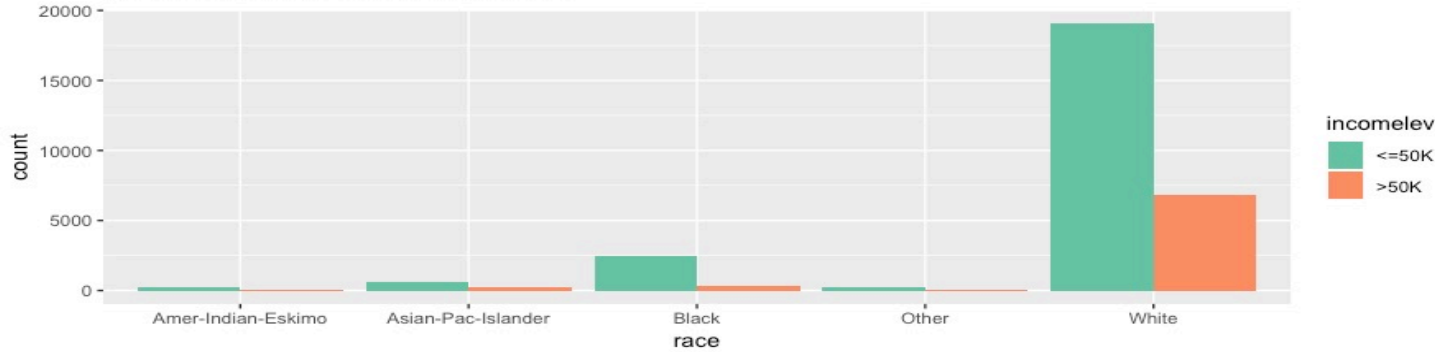


fig 3.2 Distribution of income w.r.t race

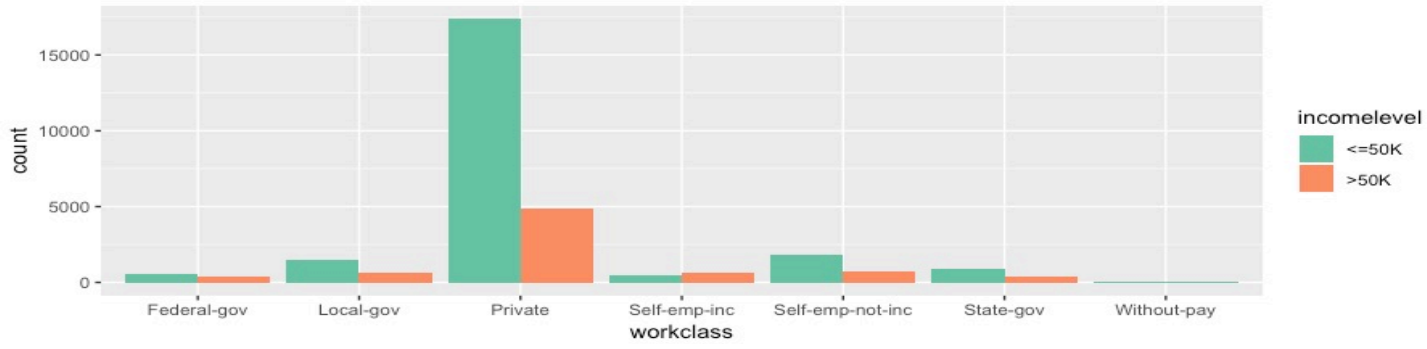
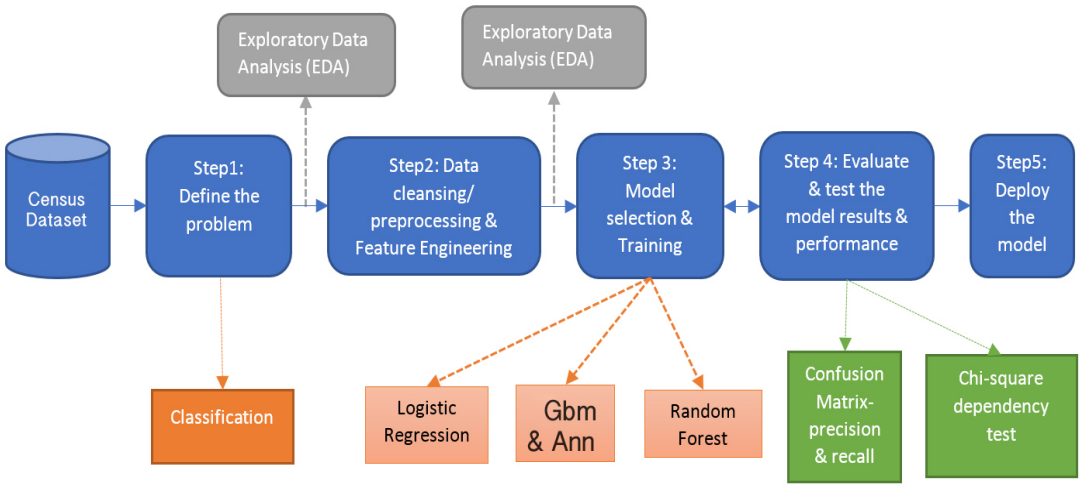


fig 3.3 Distribution of income w.r.t workclass

Figure 3.4 . Shows the Prediction model block diagram of Data Analytics process



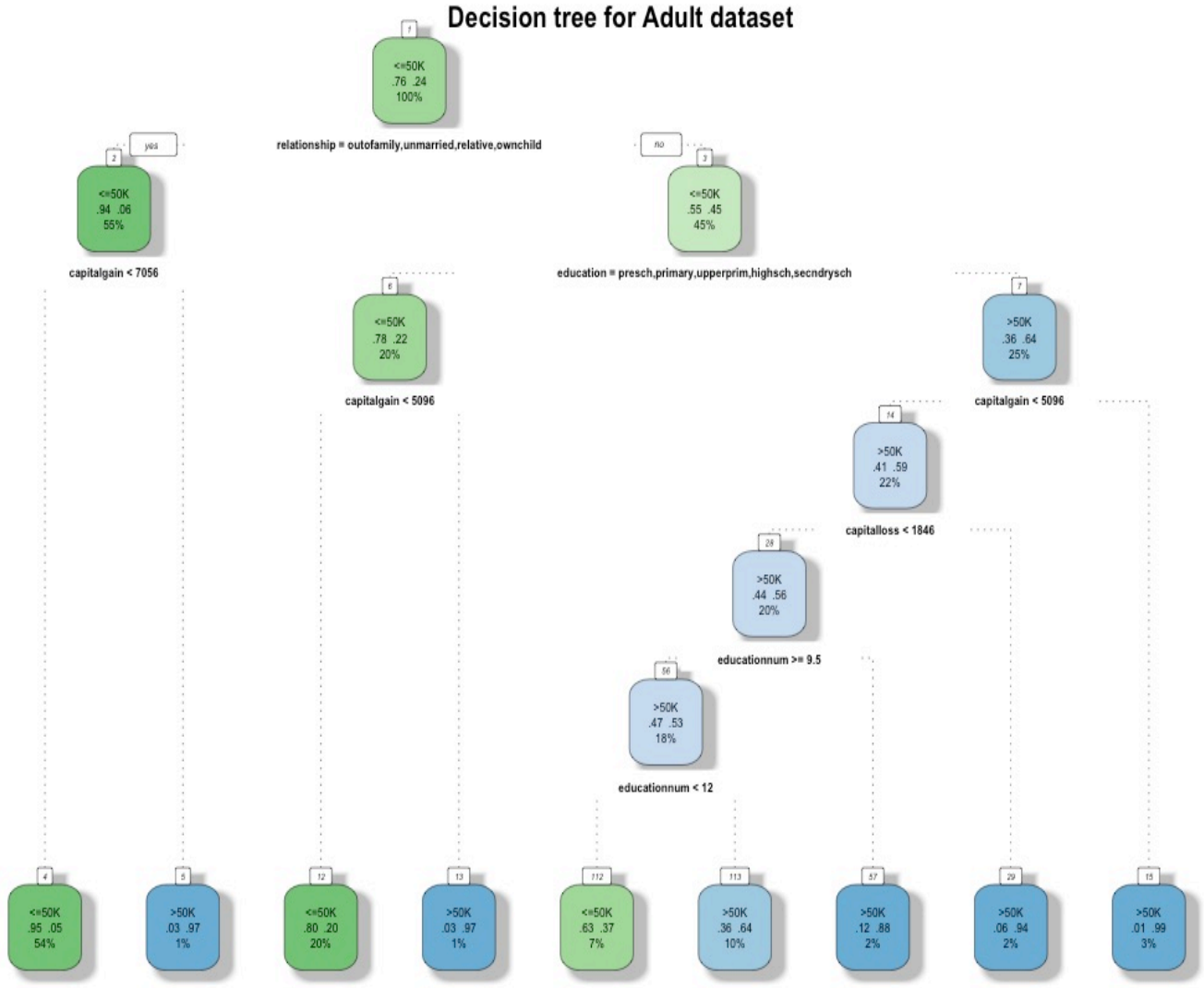


fig 3.5 Decission tree for adult datatset

Different types of graphs are generated in order to learn and think logically for which features play a major important role in order to answer the research questions mentioned in the problem statement which ultimately will help us to learn that which feature has a more significant role in determing the income of an individual from the US census adult dataset [42] in UCI repository[13][45][31][40].

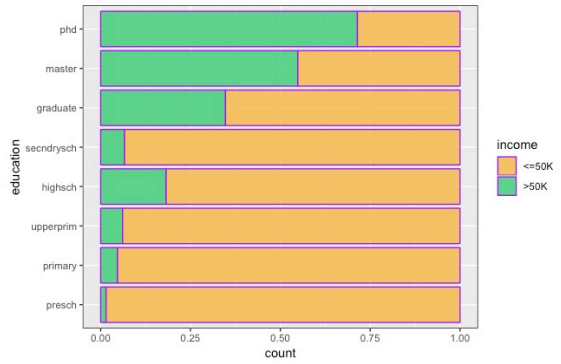


Fig 3.6 Distribution among different level and type of education with income levels

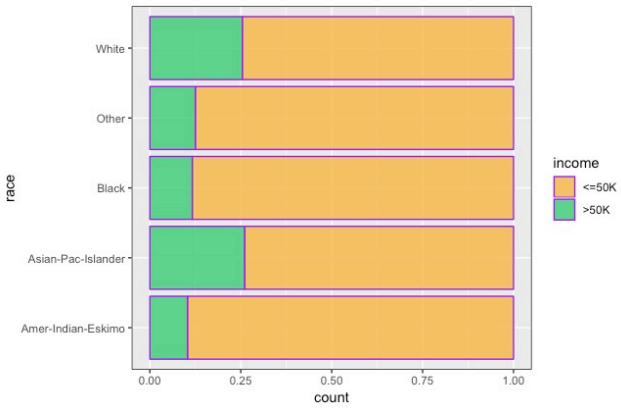


Fig 3.7 Distribution among different races with income levels

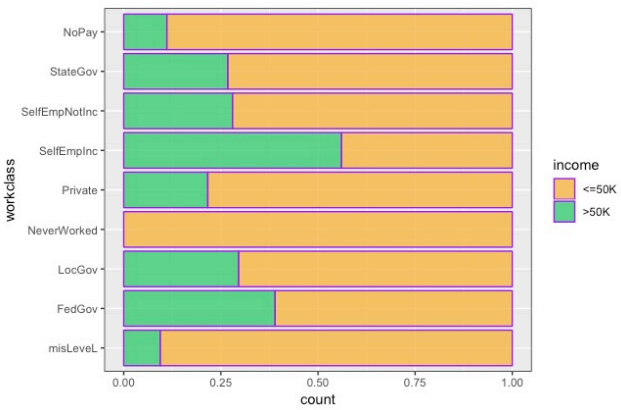


Fig 3.8 Distribution of work-class with respect to income level

The graph shows that Self Employed people have more people with income greater than 50k as well as the people working in private companies have more people with income above 50k.

1. Workclass<-**melt**(WorkclassLevel,id.vars = 'workclass')
2. **ggplot**(Workclass,**aes** (x=workclass,y=value, fill= variable)
3. **geom_bar**(stat = 'identity')+**theme**(axis.text.x = **element_text**(angle = 45, hjust = 1))+ **ggtitle**('Proportion of People with income above 50k')+ **xlab**("Work Class")+ **ylab**("Number of People")

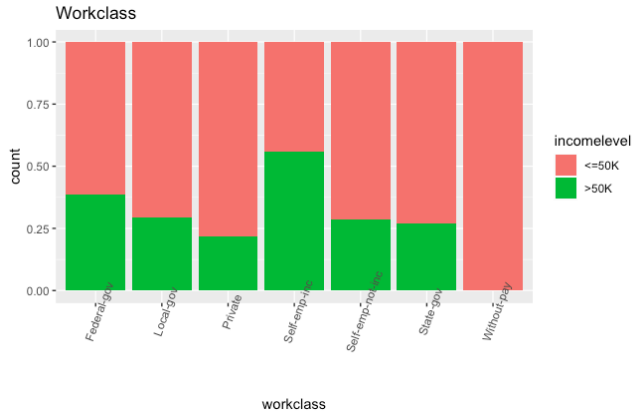


Fig 3.9 Work Class and implication on the income levels

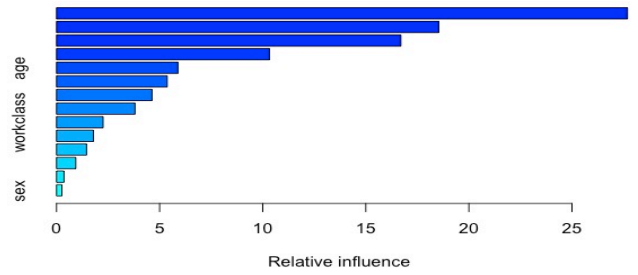


Fig 4.0 Relative influence between variables such as Sex, Workclass and Age.

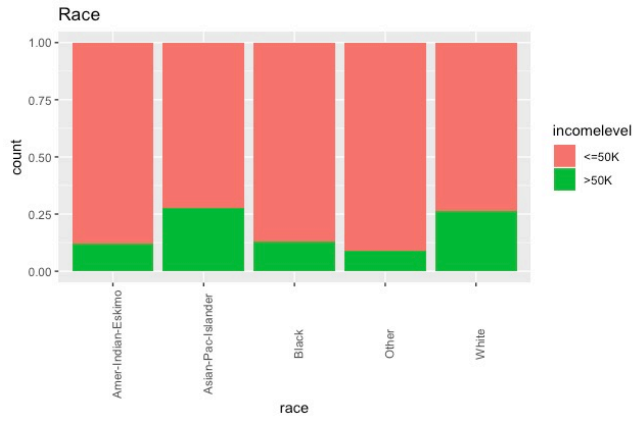


Fig 4.1 The variables work-class, occupation, marital-status, relationship all show good predictability of the income level variable.
The graph shows that the married people have more earning income (>50k) as compared to unmarried.

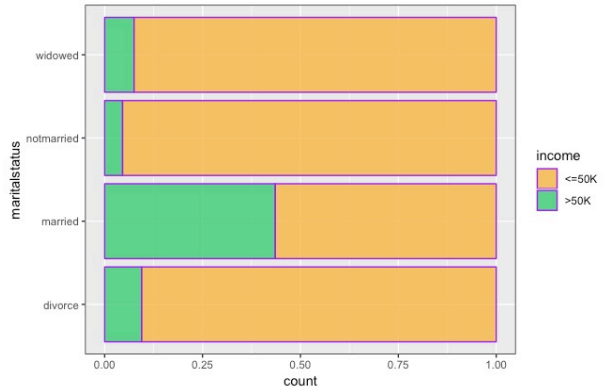


Fig 4.2 Marital Status and implication on the income levels

```
1. Maritalclass <-melt (MaritalLevel,id.vars = 'status')
ggplot (Maritalclass,aes (x=status,y=value,fill=variable))+
geom_bar(stat = 'identity')

2. theme(axis.text.x = element_text(angle = 45, hjust =
1))+  ggtitle('Proportion of People with income above
50k')+  xlab("Gender Class")+  ylab("Number of People")
```

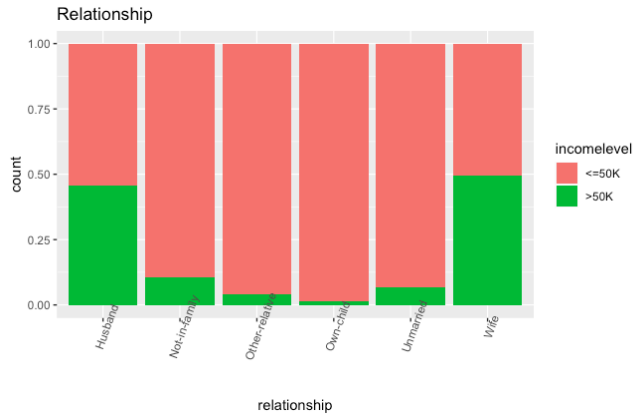


Fig 4.3 Relationship and influence on income levels

XII. LEARNING ALGORITHM

Selection and Building of a model

At first when the data visualization was done various of the features were determined which has less significant role in creating greater accuracy amongst various algorithms. Deal with missing values In this dataset the missing values are denoted by ”?” instead of the generic ”NA” or ”NAN” format. The attributes with missing values are workclass, occupation, and native country. All the three attributes are categorical and the missing observations belong to the majority class i.e ’<=50K’ [13][45][31][40]. Since the majority class has a probability of 76%, we can safely discard the observations with missing values using na.omit() from the data set as this will not affect the class probabilities as such.

Deal with categorical variable As you can see from the table 3, eight of the attributes are categorical variables with string values. We cannot string data to a classifier directly. Using the as.factor() function in R, this type of categorical variables are assigned numerical levels based on the categories [13][45][31][40]. We applied this transformation on the string attributes with distinct levels to enable processing using R.

Types of Feature Selection for various models :

1. LOGISTIC REGRESSION :
   - All Features in first processing.
   - Final weight remove in second run.
   - Final weight and native country removed in third run.

2. GBM :
   - All features used.

3. RANDOM FOREST :
   - All Features used in first run.
   - Removed native country in second.

4. SVM :
   - All features in first run.
   - Final weight and race removed in second.

5. NEURAL NETWORKS :
   - removed final weight.
   - Will try with removing other features and then select on comparatative basis.

6. NAÏVE BAYES :
   - We used all features in first run to attain higher accuracy.

Since, Each algorithm has a valid pattern to which we can implement the proposed system for predicting our desired results. But the fact that matters is to select the appropriate dataset and features to be selected as well as it's ratio into which it has been trained and will be tested. The solution proposed for the questions under the problem statement section will just be proved by the results from these proposed algorithms as well as half of the clarification is presented through visual graphs and plots.

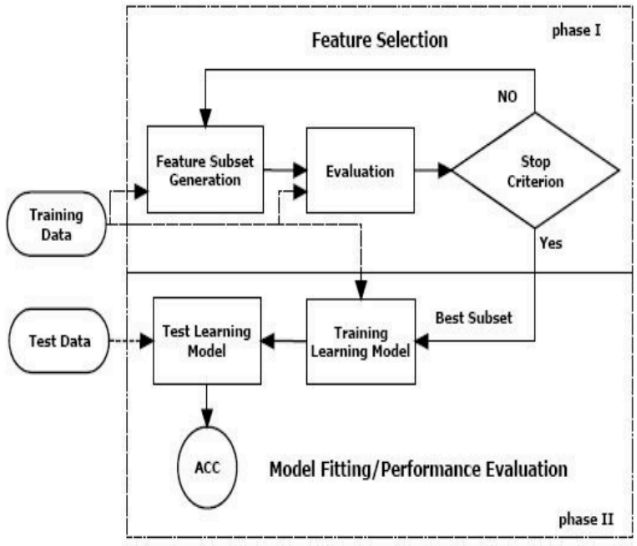
To add to it, the result phase in future approach is just a few steps away from deciding which algorithm will suit the research questions in this paper to be solved by following the steps involved under Prediction model block diagram of Data Analytics process.

A.Creating the train and test dataset

Visualize the tree and each node, we will find that:

(1) People with more capital gain tend to have more income also to add to it, people who makes planning for investing of their income also gets a share of profit shown by the visualization.
(2) It is hard to tell if a person has a high income through only viewing capital-loss information.
(3)people who are never married or divorced have lower income.
(4) people with no high school graduation tends to have a lower income as compared if a person has skills to get a same job can also be visualize through the graphs in previous sections.

Figure 4.4 showing Model used for feature selection [46][54][56]



The dataset has been divided into 75:25 ratio under which for training it is the 75% of total records whereas the 25% remaining will be used in testing. A RMS method for evaluation in model testing that is the function calling of root mean square was also calculated [68].

```
1. >ratio=sample(1:nrow(adult.cmplt),
size=0.25*nrow(adult.cmplt)) > test.data =
adult.cmplt[ratio,]

2. Test dataset 25% of total > train.data = adult.cmplt[-
ratio,]

3.Train dataset 75% of total > dim(train.data) [] 24421
dim(test.data) [] 8140
```

```
library(VIM)
aggr_plot <- aggr(adult.data, col=c('orange','purple'),
numbers=TRUE, sortVars=TRUE,
labels=names(adult.data), cex.axis=.7, gap=3,
ylab=c("Histogram of missing data","Pattern"))
library(missForest)
imputdata<- missForest(adult.data)
# check imputed values
imputdata$ximp
# assign imputed values to a data frame
adult.cmplt<- imputdata$ximp
df.master<- adult.cmplt # save a copy

set.seed(1234)
ratio = sample(1:nrow(adult.cmplt), size =
0.25*nrow(adult.cmplt))
test = adult.cmplt[ratio,] #Test dataset 25% of total
train = adult.cmplt[-ratio,] #Train dataset 75% of total

dim(train)
dim(test)
str(train)
```

Running classifier algorithms

Fitting a Logistic Regression Model
1. Logistic regression model.

1. Variable >glm.fit<- glm(income~., family=binomial(link='logit'),data = train.data)
2. This code show a Message error for warning check under which fitting the glm variable function holds probabilities 0 or 1 as numerically may occur

Part of the code above has been taken from [68].

```
glmfit1 <- glm(income ~.-fnlwgt, data=train,
family=binomial)
  summary(glmfit1)
```

```
   #incorporate train and set data set
   #Looping through various threshold to find the best
one
   threshold <- seq(0, 1, 0.04)
   acc <- rep(0, length(threshold))
   for (i in 1:length(threshold)) {
    glmprobs <- predict(glmfit1, newdata=test, type =
"response")
     glmpred <- rep("<=50k", length(test$income))
     glmpred[glmprobs > threshold[i]] = ">50k"
     glmtable <- table(glmpred, test$income)
     acc[i] <- sum(diag(glmtable))/sum(glmtable)
   }
   plot(acc, main="Threshold Selection")
   threshold[which.max(acc)]
   log.acc2 <- acc[which.max(acc)]
   log.acc2
```

The error for the first phase in the r studio will show a warning message which in this case is "Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred"[13][68][69] this message means that there exists chances where the information in the dataset can be divided into simple linear phases by parts because of easily distinguishable categories of 0 or 1. The outcomes shows the possible reasons that the most influencial predictors are age, workclass, Self-Emp-Inc, fnl-wgt, education-num and marital-status-married. As for analytical as well as statistical analysis the education-number in such cases does not contribute well. Similarly native-country also plays a very less significant role but when removing in some models it effects the accuracy results slightly by 0.20% to 0.70% [68][69].

The algorithm used for logistic regression model :

```
1. library(ROCR)

2. glmfit = glm(income ~.,data
=train,family=binomial('logit'))
summary(glmfit)threshold <- seq(0, 1, 0.04)

3. acc <- rep(0, length(threshold))
for (i in 1:length(threshold)) {

4. glmprobs <- predict(glmfit, newdata=test, type =
"response")

5. glmpred <- rep("<=50k", length(test$income))
glmpred[glmprobs > threshold[i]] = ">50k"

6. glmtable<-table(glmpred,test$income)acc[i]<
sum(diag(glmtable))/sum(glmtable)}

7. plot(acc, main="Threshold Selection")
threshold[which.max(acc)]log.acc <- acc[which.max(acc)]
log.acc

8. ROCRpred<- prediction(glmprobs, test$income)
perf<- performance(ROCRpred, "tpr", "fpr")plot(perf)
as.numeric(performance(ROCRpred, "auc")@y.values)

9.Attain Accuracy
#Baseline = Everyone makes below 50k
```

2 .Decision Tree

The algorithm used is as following :

```
1. Classification using decison trees
2. model generation for decision trees tree  = rpart(income
~ ., data = train, method ="class")

3. fancyRpartPlot(tree, main = "Decision tree for Adult
dataset [42]")
```

```
4. predictions on test and train data
5. prediction_dt_test=  predict(tree, test[,-15])
6. prediction_dt_train=  predict(tree, train[,-15])
7. predictions_test = ifelse(prediction_dt_test[,1] > = .5, "
<=50K", " >50K")
8. predictions_train = ifelse(prediction_dt_train[,1] > = .5, "
<=50K", " >50K")
9. printing the confusion matrix for both testing and training
data
10. conf_mat_dt_train = table(predictions_train,train[,15])
11. conf_mat_dt_test = table(predictions_test,test[,15])
12. print both confusion matrix
13. print("Confusion matrix for training data")
14. print(conf_mat_dt_train)
15. print("confusion matrix for testing data")
16. print(conf_mat_dt_test)
```

3. Naïve Bayes

Code Algorithm used for naïve bayes :

```
1. CLASSIFICATION USING NAIVE BAYES
2. making model for naive bayes which will be used for
prediction using model_nb = naiveBayes(incomelevel
~.,data = clean_train)
3. prediction on train and test data
   prediction_nb_train = predict(model_nb,train[,-15])
   prediction_nb_test = predict(model_nb,test[,-15])
4. confusion matrix for train and test data
   conf_mat_nb_train = table(prediction_nb_train,
train[,15])
   conf_mat_nb_test = table(prediction_nb_test,test[,15])
5. print both confusion matrix
   print("Confusion matrix for train data")
   print(conf_mat_nb_train)
```

```
   print("confusion matrix for test data")
6. print(conf_mat_nb_test)
```

4. SVM

A set of hyperplanes or a single hyper plane in context of separation of data records in scattered position to be arranged in such an order that distinguish between creating distance between largest of them from the hyperplane in ascending order such that the high or the ionfinite dimentional environment helps fgor the solving of classification problems. [13][11][40][70].

```
1. Load with library kernlab
2. library(kernlab)
3. svm4 <- ksvm(income ~ ., data = train)
4. svm4.pred.prob <- predict(svm4, newdata = test, type =
'decision')
5. svm4.pred <- predict(svm4, newdata = test, type =
'response')
6. confusionMatrix(test$income, svm4.pred)

# Output (accuracy)
######## Extras #########
7. svm.model<- svm(income~., data = train,kernel =
"radial", cost = 1, gamma = 0.1)
8. svm.predict <- predict(svm.model, test)
9. svm.pred.prob <- predict(svm.model, newdata = test,
type = 'decision')
10. confusionMatrix(test$income, svm.predict)

# Output (accuracy)
```

5. Scaling Up Accuracy By Random Forest

Random forests or random decision forests are an ensemble learning method for classification, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) of the individual trees[47].

```
1. rf.income <- randomForest(income~., train)
2. rf.pred.prob <- predict(rf, newdata = testing_set, type =
'prob')
3.          rf.pred.prob<-predict          (rf.income
newdata=test,type="prob") 4. rf.pred <- predict (rf.income,
newdata=test) rf.table <- table(rf.pred, test$income) rf.acc
<- mean(rf.pred == test$income) rf.acc
5. attain accuracy
```

6. Gradient boosting

This feature is the most vital feature of this research because for every new iteration this model helps to reduce the function capacity of losing the weights termed as “loss function” where the equation forms as (y=ax +b +e) , where e forms an error to be monitored while utilizing the model in working state .So for every running time and error clearance reduction it provides a better accuracy for each iteration to the responding feature on the basis of target classifier[13][11][48].

```
1. GRADIENT BOOSTING
2.boost.fit<gbm(as.numeric(income)1~.,data=train,distribut
ion ="bernoulli", n.trees = 2000 , interaction.depth = 2)
3. summary(boost.fit)
4. gbm.perf(boost.fit)
5. threshold <- seq(0, 1, 0.04)
6. acc <- rep(0, length(threshold))
7. for (i in 1:length(threshold)) {boost.probs
<predict(boost.fit, test, n.trees = 2000)boost.pred <-
ifelse(boost.probs > threshold[i], 1, 0)
8. acc[i] <- mean(boost.pred == as.numeric(test$income)-
1)}
plot(acc, main="Threshold Selection")
9. boost.acc <- acc[which.max(acc)]
10. boost.acc
11. Attain accuracy
```

```
   > gbm.perf(boost.fit)
   OOB generally underestimates the optimal number of
iterations although predictive performance is
reasonably competitive. Using cv_folds >1 when calling
gbm usually results in improved predictive
performance.
   388
   attr(,"smoother")
   Call:
   loess(formula = object$oobag.improve ~ x, enp.target
= min(max(4,
      length(x)/10), 50))

   Number of Observations: 2000
   Equivalent Number of Parameters: 39.99
   Residual Standard Error: 0.000619269
   > threshold <- seq(0, 1, 0.04)
   > acc <- rep(0, length(threshold))
   > for (i in 1:length(threshold)) {
   +     boost.probs <- predict(boost.fit, test, n.trees =
2000)
   +    boost.pred <- ifelse(boost.probs > threshold[i], 1,
0)
   +                acc[i] <- mean(boost.pred ==
as.numeric(test$income)-1)
   + }
   > plot(acc, main="Threshold Selection")
   > boost.acc <- acc[which.max(acc)]
   > boost.acc
    0.8900082
```

```
> #Max accuracy = 89.13 %
> tb2 <- table(boost.pred, test$income)
```

XIII. EVALUATION

The evaluation phase of this research paper will represent the model description with working algorithms under complexity analysis, operational analysis, efficiency analysis as well as comparison with accuracy scores of other algorithms resulted from different research papers which are verified and approved by any valid journal or source.

To start with the model description which will represent the complexity as well as the work flow structure of each algorithm at run time showing console output. Using the scikit learn package in Python and Radiant package in R offers a wide variety of classifier models like Logistic, SVM, Neural Network, Random Forest, and various ensemble classifier techniques. It is a time-consuming process to determine which would be the best method to classify the data at hand [11].

This research approach has been to try various methods on a trial basis based on understanding of the usage of techniques to see which performs best on the test dataset and to learn from various features that manipulate the accuracy results influencing the dataset following the pattern to answer all the research questions of this paper.

1. Logistic regression model :

```
> library(ROCR)
> glmfit = glm(income ~.,data =
train,family=binomial('logit'))
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
> summary(glmfit)

Call:
glm(formula = income ~ ., family = binomial("logit"), data
= train)

Deviance Residuals:
      Min       1Q   Median       3Q      Max
  -5.1357  -0.4716  -0.1783  -0.0372   3.5585

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Coefficients: (1 not defined because of singularities)
    .
    .
(Dispersion parameter for binomial family taken to be 1)

Null deviance: 40266  on 36631  degrees of freedom
Residual deviance: 22542  on 36587  degrees of freedom
AIC: 22632

Number of Fisher Scoring iterations: 11

> plot(acc, main="Threshold Selection")
> threshold[which.max(acc)]
0.48
> log.acc <- acc[which.max(acc)]
> log.acc
0.8619984
> ROCRpred<- prediction(glmprobs, test$income)
> perf<- performance(ROCRpred, "tpr", "fpr")
> plot(perf)
> as.numeric(performance(ROCRpred, "auc")@y.values)
0.9128794
```

The Confusion matrix formed from the logistic regression is as following :

```
> confusionmatrix_LR<- table(test$income, glmpred > 0.5)
> confusionmatrix_LR

         FALSE TRUE
 <=50K  8579  689
  >50K   1009 1933
```

Now From this confusion matrix we can study various other performance factors from which we can analyze how well the prediction model efficiency works upon certain set of problems in a dataset to reveal the actual prediction tendency of change in variables and features in a dataset. Technical details understanding confusion matrix involves functions requiring that the factors have exactly the same levels [49].

For two class problems, the sensitivity, specificity, positive predictive value and negative predictive value is calculated using the positive argument. For more than two classes, these results are calculated comparing each factor level to the remaining levels (i.e. a "one versus all" approach). In each case, the overall accuracy and Kappa statistic are calculated[49].

The overall accuracy rate is computed along with a 95 percent confidence interval for this rate (using binom.test) and a one-sided test to see if the accuracy is better than the "no information rate," which is taken to be the largest class percentage in the data[30].

To calculate performance we have the TP, FN, FP, and TN measures which will help to get the scores as following :

TP = 8579 , FN = 689 , FP = 1009 and TN = 1993

Measure	Value	Derivation

Sensitivity	0.8948	TPR = TP / (TP + FN)
Specificity	0.7372	SPC = TN / (FP + TN)
Precision	0.9257	PPV = TP / (TP + FP)
Negative Predictive Value	0.6570	NPV = TN / (TN + FN)
False Positive Rate	0.2628	FPR = FP / (FP + TN)
False Discovery Rate	0.0743	FDR = FP / (FP + TP)
False Negative Rate	0.1052	FNR = FN / (FN + TP)
Accuracy	0.8609	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.9099	F1 = 2TP / (2TP + FP + FN)
MCC	0.6068	Using Matthews Correlation Coefficient

2. Naive Bayes

Understandably, naive Bayes performed less efficient as compared to the other models because of the conditional independence assumption. The results from the console running the algorithm described in the proposed solution provides us the following output :

```
"Confusion matrix for train data"
> print(conf_mat_nb_train)

prediction_nb_train <=50K  >50K
          <=50K 26494  6229
          >50K   1393  2516
> print("confusion matrix for test data")
"confusion matrix for test data"
> print(conf_mat_nb_test)
> # error rate for training and testing data
> error_rate_test_nb
=(conf_mat_nb_test[1,2]+conf_mat_nb_test[2,1])/(conf_ma
t_nb_test[1,2]+conf_mat_nb_test[2,1]+conf_mat_nb_test[1,
1]+conf_mat_nb_test[2,2])
> print("Error rate for Naive Bayes on test data")
"Error rate for Naive Bayes on test data"
> print(error_rate_test_nb)
0.2109746
> error_rate_train_nb
=(conf_mat_nb_train[1,2]+conf_mat_nb_train[2,1])/(conf_
mat_nb_train[1,2]+conf_mat_nb_train[2,1]+conf_mat_nb_t
rain[1,1]+conf_mat_nb_train[2,2])
> print("Error rate for Naive Bayes on train data")
"Error rate for Naive Bayes on train data"
> print(error_rate_train_nb)
0.2080694
>
```

From these outputs the confusion matrix for naïve bayes on test data was generated as following :

```
> print(conf_mat_nb_test)

prediction_nb_test <=50K >50K
          <=50K  8784 2092
          >50K    484  850
```

To calculate performance we have the TP, FN, FP, and TN measures which will help to get the scores as following :

TP = 8784 , FN = 2092 , FP = 484 and TN = 850

Measure	Value	Derivation
Sensitivity	0.9478	TPR = TP / (TP + FN)
Specificity	0.2889	SPC = TN / (FP + TN)
Precision	0.8076	PPV = TP / (TP + FP)
Negative Predictive Value	0.6372	NPV = TN / (TN + FN)
False Positive Rate	0.7111	FPR = FP / (FP + TN)
False Discovery Rate	0.1924	FDR = FP / (FP + TP)
False Negative Rate	0.0522	FNR = FN / (FN + TP)
Accuracy	0.7980	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.8721	F1 = 2TP / (2TP + FP + FN)
MCC	0.3245	Using Matthews Correlation Coefficient

3. Neural Networks

Use caret package library to train a model using neural net. Neural networks are pretty complicated, involving non-linear transformations of our inputs into a ‘hidden layer’ of nodes that are then translated into our output prediction with a potentially very large number of parameters involved. The package NeuralNetTools has some nice functions available for visual understanding and connections between the features on a larger scale [50].

The output on the console after run time of algorithm is as following :

```
> nn <- nnet(income ~ ., data = train, size = 40, maxit =
500)
Check Error in nnet.default(x, y, w, entropy = TRUE, ...) :
```

```
too many (1921) weights
> nn.pred <- predict(nn, newdata = test, type = 'raw')
> pred <- rep('<=50K', length(nn.pred))
> pred[nn.pred>=.5] <- '>50K'
> # confusion matrix
> tb1 <- table(pred, test$income)
> emp <- data.frame( +   tb1, +   stringsAsFactors = FALSE
+ )
> tb1
```

From these outputs the confusion matrix for neural networks on test data was generated as following :

```
   pred    <=50K   >50K
<=50K    8556      910
 >50K     712     2032
```

To calculate performance of neural networks we have the TP, FN, FP, and TN measures which will help to get the scores as following :

TP = 8556 , FN = 910 , FP = 712 and TN = 2032

Measure	Value	Derivation
Sensitivity	0.9232	TPR = TP / (TP + FN)
Specificity	0.6907	SPC = TN / (FP + TN)
Precision	0.9039	PPV = TP / (TP + FP)
Negative Predictive Value	0.7405	NPV = TN / (TN + FN)
False Positive Rate	0.3093	FPR = FP / (FP + TN)
False Discovery Rate	0.0961	FDR = FP / (FP + TP)
False Negative Rate	0.0768	FNR = FN / (FN + TP)
Accuracy	0.8672	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.9134	F1 = 2TP / (2TP + FP + FN)
MCC	0.6289	Using Matthews Correlation Coefficient

4. Support Vector Machine (SVM)

Support Vector Machines SVMs, in general, work well with nonlinear data. We noticed nonlinearity in some of our variables like capital gain/loss etc and hence we also used SVMs for our classification task to model the parameters without any fine tuning[11][16].

The output on the console after run time of algorithm is as following :

```
acc <- rep(0, length(threshold))
> #incorporate train and set data set
> #Looping through various threshold to find the best one
> threshold <- seq(0, 1, 0.04)
> svm4 <- ksvm(income ~ ., data = train)
> svm.predict <- predict(svm.model, test)
> as.numeric(performance(ROCRpred, "auc")@y.values)
0.9119577
> ## manual roc
> r=pref$table[1,2]/(pref$table[1,2]+pref$table[2,2])
> threshold[which.max(acc)]
> svm.model1<- svm(income~age + workclass + education
+ capitalgain + occupation + nativecountry + sex +
hoursperweek + maritalstatus, data = train,kernel =
"radial", cost = 1, gamma = 0.1)
> log.acc2 <- acc[which.max(acc)]
> # with library kernlab
> library(kernlab)
> svm4 <- ksvm(income ~ ., data = train)
> svm4.pred.prob <- predict(svm4, newdata = test, type =
'decision')
> confusionMatrix(test$income, svm4.pred)
Confusion Matrix and Statistics
```

'Positive' Class : <=50K

From these outputs the confusion matrix for neural networks on test data was generated as following :

```
                Reference
Prediction  <=50K   >50K
   <=50K     8609     659
    >50K      919    2023
```

```
Accuracy : 0.8708
95% CI : (0.8647, 0.8767)
No Information Rate : 0.7803
 P-Value [Acc > NIR] : < 2.2e-16

 Kappa : 0.6357
 Mcnemar's Test P-Value : 7.032e-11

Prevalence : 0.7803
Detection Rate : 0.7051
Detection Prevalence : 0.7590
Balanced Accuracy : 0.8289
```

To calculate performance of neural networks we have the TP, FN, FP, and TN measures which will help to get the scores as following :

TP = 8609 , FN = 659 , FP = 919 and TN = 2023

Measure	Value	Derivation

Sensitivity	0.9035	TPR = TP / (TP + FN)
Specificity	0.7543	SPC = TN / (FP + TN)
Precision	0.9289	PPV = TP / (TP + FP)
Negative Predictive Value	0.6876	NPV = TN / (TN + FN)
False Positive Rate	0.2457	FPR = FP / (FP + TN)
False Discovery Rate	0.0711	FDR = FP / (FP + TP)
False Negative Rate	0.0965	FNR = FN / (FN + TP)
Accuracy	0.8722	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.9160	F1 = 2TP / (2TP + FP + FN)
MCC	0.6368	Using Matthews Correlation Coefficient

5. Decision Trees

In this Research for better comparison a trial to implement Decision Tree Model for our classification task was also conducted. Decision Trees (DTs) are a nonparametric supervised learning method used for classification. While they performed well on the training set, their performance on the testing set was less than expected. Although the distribution of train and the test features are similar, decision trees suffer from out of sample prediction[11][16][51]. The optimistic scope was also unsure about how to prune the tree for better results. The binary labeled data set was incredibly convenient to integrate with the Decision Tree structure[51].

```
> print("Confusion matrix for train data")
"Confusion matrix for train data"
> print(conf_mat_dt_train)

predictions_train <=50K  >50K
<=50K 26442  3508
>50K   1445  5237

> print("confusion matrix for test data")
"confusion matrix for test data"
> print(conf_mat_dt_test)

predictions_test <=50K >50K
<=50K  8790 1204
>50K    478 1738
```

```
> print("Error rate for Decision Trees on test data")
"Error rate for Decision Trees on test data"
> print(error_rate_test_dt)
0.1377559
>
> print("Error rate for Decision Trees on train data")
"Error rate for Decision Trees on train data"
> print(error_rate_train_dt)
0.1352097
```

To calculate performance of Decision Trees we have the TP, FN, FP, and TN measures which will help to get the scores as following :

TP = 8790 , FN = 1204 , FP = 478 and TN = 1738

Measure	Value	Derivation
Sensitivity	0.9484	TPR = TP / (TP + FN)
Specificity	0.5908	SPC = TN / (FP + TN)
Precision	0.8795	PPV = TP / (TP + FP)
Negative Predictive Value	0.7843	NPV = TN / (TN + FN)
False Positive Rate	0.4092	FPR = FP / (FP + TN)
False Discovery Rate	0.1205	FDR = FP / (FP + TP)
False Negative Rate	0.0516	FNR = FN / (FN + TP)
Accuracy	0.8622	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.9127	F1 = 2TP / (2TP + FP + FN)
MCC	0.5983	Using Matthews Correlation Coefficient

6. Random Forest

A creation of parent to root nodes of branching tree like architecture, the random forest helps to prove that it works quite high accurate on the basis of the problems in the dataset. With over a large number of probabilities to remove and try different variables with random model takes a lot of time but it helped to prove that native country also has less significance on accuracy by hit and trials on adding and removing it. [16]
The output on the console after run time of algorithm is as following :

```
> rf.income <- randomForest(income~.-nativecountry, train)
> rf.pred <- predict(rf.income, newdata=test)
> rf.table <- table(rf.pred, test$income)
> rf.acc <- mean(rf.pred == test$income)
```

```
> rf.acc
0.8869132
```

```
#confusion matrix for random forest
> tb4 <- table(rf.pred, test$income)
> tb4

rf.pred  <=50K   >50K
 <=50K   8752    899
  >50K    516    2043
```

Measure	Value	Derivation
Sensitivity	0.9472	TPR = TP / (TP + FN)
Specificity	0.6927	SPC = TN / (FP + TN)
Precision	0.9066	PPV = TP / (TP + FP)
Negative Predictive Value	0.8065	NPV = TN / (TN + FN)
False Positive Rate	0.3073	FPR = FP / (FP + TN)
False Discovery Rate	0.0934	FDR = FP / (FP + TP)
False Negative Rate	0.0528	FNR = FN / (FN + TP)
Accuracy	0.8869	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.9265	F1 = 2TP / (2TP + FP + FN)
MCC	0.6756	Using Matthews Correlation Coefficient

7. Gradient Boosting Machine (GBM)

The main focus to implement and fit Gradient Boosting Classifier model on the data for the Classification task was done selecting various qualitative and quantitative measures of GBM which are helpful in revealing the complexity analysis as well as the influence of different features upon the number of iterations as well as providing the freedom of changing the number of size max and iterations according to different types of datasets to help the analyzing process much more precise [11][51]. Some of the important features that were observed were Marital Status: Married, capital gain, hours per week and age, all of which are obviously important predictors of income level.

```
> boost.fit <- gbm(as.numeric(income)-1~.,data=train,
distribution =+"bernoulli", n.trees = 2000 , interaction.depth
= 2)
```

```
> summary(boost.fit)

                    var               rel.inf

relationship   relationship       27.8142576
education      education          18.8832471
capitalgain    capitalgain        18.1449611
educationnum educationnum         9.5897707
age            age                5.6414250
capitalloss    capitalloss        5.2212764
fnlwgt         fnlwgt             4.0345558
hoursperweek   hoursperweek       4.0041542
workclass      workclass          1.9745522
occupation     occupation         1.6407280
maritalstatus maritalstatus       1.5882777
nativecountry nativecountry       0.9236046
race           race               0.3238110
sex            sex                0.2153786
```

OOB generally underestimates the optimal number of iterations although predictive performance is reasonably competitive. Using cv_folds > 1 when calling gbm usually results in improved predictive performance.

```
> gbm.perf(boost.fit)
> 354
attr(,"smoother")
Call:
loess(formula = object$oobag.improve ~ x, enp.target =
min(max(4, length(x)/10), 50))
```

```
Number of Observations: 2000
Equivalent Number of Parameters: 39.99
Residual Standard Error: 0.0006059929
> threshold <- seq(0, 1, 0.04)
> acc <- rep(0, length(threshold))
> for (i in 1:length(threshold)) {
+   boost.probs <- predict(boost.fit, test, n.trees = 2000)
+   boost.pred <- ifelse(boost.probs > threshold[i], 1, 0)
+    acc[i] <- mean(boost.pred == as.numeric(test$income)-
1)
+ }
> plot(acc, main="Threshold Selection")
> boost.acc <- acc[which.max(acc)]
> boost.acc
> 0.8891236
```

```
> tb2 <- table(boost.pred, test$income)
> tb2

boost.pred  <=50K   >50K
        0    9128    1477
        1     140    1465
```

To calculate performance of Gradient Boosting Machine we

have the TP, FN, FP, and TN measures which will help to get the scores as following :
TP = 9128 , FN = 1477 , FP = 140 and TN = 1465

Measure	Value	Derivation
Sensitivity	0.9849	TPR = TP / (TP + FN)
Specificity	0.4980	SPC = TN / (FP + TN)
Precision	0.8607	PPV = TP / (TP + FP)
Negative Predictive Value	0.9128	NPV = TN / (TN + FN)
False Positive Rate	0.5020	FPR = FP / (FP + TN)
False Discovery Rate	0.1393	FDR = FP / (FP + TP)
False Negative Rate	0.0151	FNR = FN / (FN + TP)
Accuracy	0.8891	ACC = (TP + TN) / (Sum of Confusion matrix)
F1 Score	0.9186	F1 = 2TP / (2TP + FP + FN)
MCC	0.6111	Using Matthews Correlation Coefficient

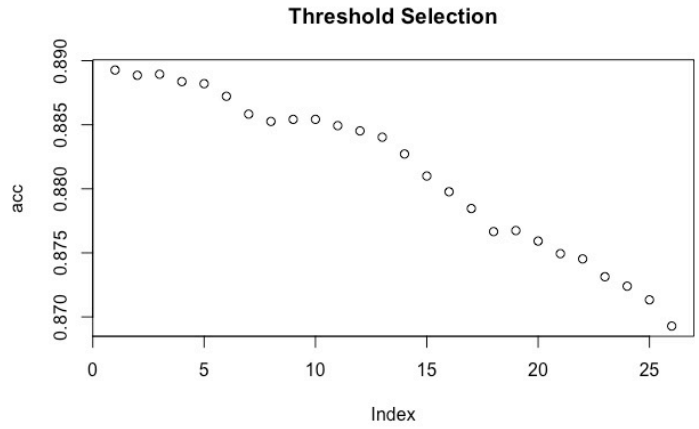


Fig 4.5 showing accuracy plot of GBM model

COMPLEXITY ANALYSIS

TIME COMPLEXITY
The function which helps to calculate time taken by the iterations as well as the whole algorithm to run under which the confusion matrix or graphical representation is involved is known as time complexity.it works on number of steps and for loops or any other functions, operators as well as set of library package tools in context of functions to be runned by a single or the whole algorithm.

BIG O NOTATION
Big O notation helps to understand how fast a function is going through all the process to frame an output result or accuracy.

This section will represent the algorithmic performance of running the whole program as well as running each algorithm to provide it's specific timing as well as big o notation on the complexity of algorithm for each of the models used on adult dataset [42] to provide better prediction. To begin with, the specifications of hardware used for rendering the whole code is as following :

- Model Identifier: MacBook Pro '15' 2012 (non retina)
- Processor Name: Intel Core i7
- Processor Speed: 2.3 GHz
- Number of Processors: 1
- Total Number of Cores: 4
- L2 Cache (per Core): 256 KB
- L3 Cache: 6 MB
- Memory: 4 GB

In addition to it, the software requirements used for running algorithmic outputs in R programming language was done in Rstudio Version (1.1.456).

The completion of the whole algorithms as well as the visualizations to be plot under the console as well as the total running time of the process takes upto 16-18 minutes maximum.

This is because the iterations as well as the number of epochs under GBM gets higher rates and they act as running from source once recall because if we start from any other point instead of the first line, the code will show errors because it will not be able to find the library packages required by different algorithms as well as corr plots and other visualizations.

RUNTIME AND BIG O NOTATION COMPLEXITY ANALYSIS FOR EACH ALGORITHM

To conclude the timing from starting to the end of the code , TicToc() package library was installed. To begin with, at the initial beginning of the algorithmic model tic() has been used and till the line where we want to include the execution time toc() is used.

The concept involves to provide the time taken by each algorithm to run as well as to provide the knowledge for how fast a function is enhancing. More the number of loops or functions in a model, lesser the time performance because in O (log n^x) where More the value of X lesser the quality of model which helps revealing that the model has lesser or more complexity counted by running the for loops

or functions in a program which includes nested functions or loops.

```
Algorithm Psedo Code ( Resulting 17 minutes Approx ) :
library(tictoc)
>tic() begin
.
.(Total Algorithms and functions)
.
   > toc() end
   1030.207 sec elapsed
> round(auc, 4)
```

The Following table represents the results for each algorithm to start from load and run the connectivity to dataset to the end of reading each line of algorithm to the accuracy forming a confusion matrix as well as the roc curve representing area under the curve value with the visualization of features in a graphical plot. In addition to it, the "total execution time" will represent the time from the first line to end of each model with the gg plot as well as feature visualizations. On the other hand, the "Individual Execution Time" will simply represents the time calculated by each model to run just the small part of core algorithmic logic that was used to run the code and output display showing accuracy results on the console. So, here are some of the results taken from various intervals using the TicTok Library Package.

Model	Total Execution	Individual Execution Time	Big O Notation
SVM	38.67 sec	12.32 sec	O (n_{SV} p)
ANN	23.182 sec	7.09 sec	O(p log n_T)
NAÏVE BAYES	56.381 sec	17.987 sec	O(p)
RANDOM FOREST	52.205 sec	14.09 sec	O(p log n)
DECISION TREES	27.947 sec	5.167 sec	O(p)
GBM	61.762 sec	18.2 sec	O(p log n)
LOGISTIC REGRESSION	19.862 sec	5.041sec	O(p)

Where :

- *p is the number of features.
- *n is the number of records.
- *n_T is the number of trees.
- *n_{SV} is the number of support vectors.

COMPARISON TABLE

This research represents the model with the largest training accuracy and a large testing accuracy. To test if the model is suiting best results we compared the training and testing accuracy to find testing accuracy comparable to if not lesser than the training accuracy as compared to other research work done on the "adult dataset [42]". To conclude that we are not overfitting, gradient boosting model is selected as the final model showing best result.

Model	Previous Accuracy	Verified New Results
SVM	84.92% [7]	87.22%
ANN	85.07% [11]	86.72%
NAÏVE BAYES	81.20%[19]	79.80%
RANDOM FOREST	82.27%[19]	88.69%
DECISION TREES	83.89%[19]	86.22%
GBM	86.29%[11]	89.01%
LOGISTIC REGRESSION	85.11%[17]	86.09%

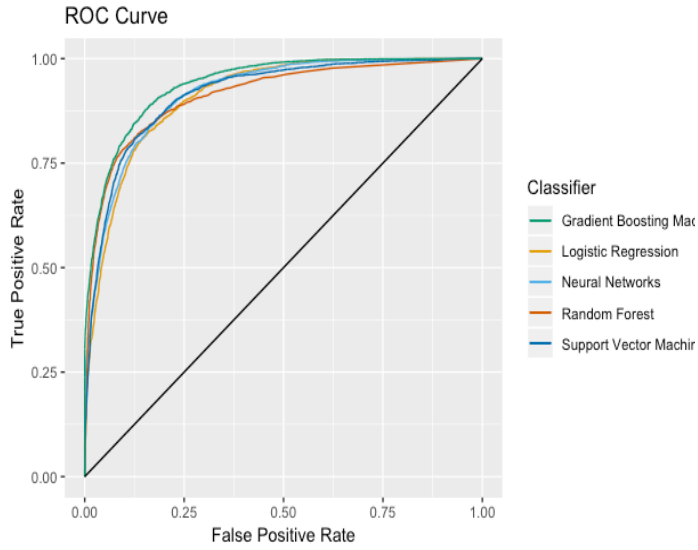
ROC CURVE PLOT

```
#Algorithm used for roc plot :

 auc <- rbind(performance(pr, measure =
'auc')@y.values[instance],
      performance(pr1, measure =
'auc')@y.values[[instance]],
            performance(pr2, measure =
'auc')@y.values[[instance]],
            performance(pr4, measure =
'auc')@y.values[[instance]],
            performance(pr5, measure =
'auc')@y.values[[instance]])

rownames(auc) <- (c('Logistic Regression', 'Neural
Networks',  'Random Forest', 'Support Vector
Machine','Gradient Boosting Machine'))

colnames(auc) <- 'Area Under ROC Curve'
round(auc, 4)
```



Area Under ROC Curve	
Logistic Regression	0.9125
Neural Networks	0.9211
Random Forest	0.9156
Support Vector Machine	0.9172
Gradient Boosting Machine	0.9404

Observations :

Removing non relevant attributes improves accuracy

- GBM with the highest area under curve value near to 1 proves as the best model for prediction of accurate results.
- For Random Forest, accuracy improves by 5%
- For Naïve bayes, accuracy falls slightly by 2%
- For SVM, accuracy improves by 2.1%
- Random Forest– Removing co-related attributes improves accuracy.
- Removing Native-Country improves accuracy by 0.91% and 5.83% in logistic as well as random forest respectively.

Results of importance of features determined using gradient boost regression can be seen from the performance metric table. Also under the visualization defined under the previous sections, it can be seen that the most useful features are relationship with married class, capital gain, followed by capital loss and age. This is in line with what we have obtained as important form logistic and neural networks. It is surprising that education, occupation and workclass did not contribute as much to the prediction as expected. Also the survey final weight was not a strong help in prediction. To add to it, Gender and race were not as useful for the classification either.

Answers to research questions :

1. Education plays a significant role.
2. If a person is married, his/her probability of getting over 50k annual income will increase significantly.
3. Sex doesn't play an important role.
4. The race of a person plays a significant role.
5. More the age of a person, the probability to get over 50k annual income will increase.

GBM with removal of native country and Random Forest with removal of final weight and native country works best to show highest accuracy rates.

POTENTIAL IMPROVEMENTS AND FUTURE WORK

Use of Prediction models by retailers to make analytical decisions on factory outlet setups in different regions of the country based on income of majority popluation. Companies Extending Credit will be helped make a decision. Exact salary data of individuals –Numerical data instead of categorical variables due to which we could better understand the data and categorize it into more efficient and accurate way instead of just classifying it as high or low.

REFERENCES

[1] Sharath R, Krishna Chaitanya S, Nirupam K N, Sowmya B J and K. G. Srinivasa, "Data analytics to predict the income and economic hierarchy on Census data," 2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, 2016, pp. 249-254.

[2] Fay, R.E. and Herriot, R.A. (1979), Estimates of Income for Small Places: An Application of James-Stein Procedures to Census Data, Joumal of the American Statistical Association, 74, 269-277

[3] Krieger, N., Williams, D. R., Moss, N. (1997). Measuring Social Class in U.S. Public Health Research: Concepts, Methodologies, and Guidelines. Annual Review of Public Health , 18,341-378.

[4] S. Latif and Z. Z. Lecturer, "Customer annual income prediction using resampling approach," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, 2017, pp. 3865-3870.

[5] A Zhong, Erheng, et al. "User demographics prediction based on mobile data." Pervasive and mobile computing 9.6 (2013): 823-837..

[6] A.Lazar and R. Zaremba, "Support Vector Machines Optimization - An Income Prediction Study," 2006 International Multi-Conference on Computing in the Global Information Technology - (ICCGI'06), Bucharest, 2006, pp. 44-44. doi: 10.1109/ICCGI.2006.67

[7] A Lazar. "Income Prediction via Support Vector Machine", IEEE conference on Machine Learning and applications,16-18 Dec. 2004 DOI: 10.1109/ICMLA.2004.1383506.

[8] Bhavin Patel et al, "Comparative Analysis of Classification Models on Income Prediction" IJETTCS,

Volume 5, Issue 4, April 2017, ISSN: 2321-8169, PP 451– 455.

[9] Y. Bengio et al., "Introduction to the special issue on neural networks for data mining and knowledge discovery," IEEE Trans. Neural Networks, vol. 11, pp. 545-549, 2000.

[10] S.Archana et al., "Survey of Classification Techniques in Data Mining", International Journal of Computer Science and Mobile Applications, Vol.2 Issue. 2, February- 2014.

[11] Vidya Chockalingam, Sejal Shah and Ronit Shaw: "Income Classification using Adult Census Data", https://cseweb.ucsd.edu/classes/wi17/cse258-a/reports/a120.pdf

[12] Sisay Menji Bekena:"Using decision tree classifier to predict income levels", Munich Personal RePEc Archive 30th July, 2017

[13] Mohammed Topiwalla: "Machine Learning on UCI Adult data Set Using Various Classifier Algorithms And Scaling Up The Accuracy Using Extreme Gradient Boosting", University of SP Jain School of Global Management.

[14] Chet Lemon, Chris Zelazo and Kesav Mulakaluri: "Predicting if income exceeds $50,000 per year based on 1994 US Census Data with Simple Classification Techniques", https://cseweb.ucsd.edu/jmcauley/cse190/reports/sp15/048.pdf.

[15] S.Deepajothi and Dr. S.Selvarajan: "A Comparative Study of Classification Techniques On Adult Data Set", International Journal of Engineering Research Technology (IJERT), ISSN: 2278-0181 Vol. 1 Issue 8, October2012.

[16] Haojun Zhu: "Predicting Earning Potential using the Adult dataset ", https://rstudio-pubs-static.s3.amazonaws.com/235617_51e06fa6c43b47d1b6daca2523b2f9e4.html

[17] Jason Nguyen ," Logistic Regression with UCI Adult Income", https://www.kaggle.com/flyingwombat/logistic regression-with-uci-adult-income

[18] Douglas J. Kennard. 2018. Computer-Assisted Crowd Transcription of the U.S. Census with Personalized Assignments for Better Accuracy and Participation. In Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries (JCDL '18). ACM, New York, NY, USA, 41-44. DOI: https://doi.org/10.1145/3197026.3197067

[19] Snehal Chemburkar et al., "Data Mining Project", https://github.com/snehalvartak/Income-Level-Prediction

[20] Institute of Technology Blanchardstown , BI and Data Mining Applications Prject", "Predicting earning potential on Adult dataset ", http://www.dataminingmasters.com/uploads/studentProjects/Earning_potential_report.pdf

[21] C Navoneel and B Sanket,"A Statistical Approach to Adult Census Income Level Prediction","CoRR","abs/1810.10076", 2018, http://arxiv.org/abs/1810.10076

[22] Machine Learning - What it is and why it matters, November 22, 2017,http://www.sas.com/en_us/insights/analytics/machine-learning.html

[23] P. Louridas and C. Ebert, "Machine Learning" in IEEE Software, vol. 33, no. 5, pp. 110-115,Sept.-Oct. 2016, http://ieeexplore.ieee.org.ezproxy.library.tufts.edu/stamp/stamp.jsp?tp=&arnumber=7548905&isnumber=7548893

[24] Jay-Louise Weldon. "Managing the census data base: data description, acquisition, and manipulation". In Proceedings of the June 7-10, 1976, national computer conference and exposition (AFIPS '76). ACM, New York,USA,863-867. DOI=http://dx.doi.org/10.1145/1499799.1499916

[25] Walter E. Simonson and William T. Alsbrooks. 1975. A DBMS for the U. S. Bureau of the Census. In Proceedings of the 1st International Conference on Very Large Data Bases (VLDB '75). ACM, New York, NY, USA, 496-498. DOI: https://doi.org/10.1145/1282480.1282521

[26] Christian Clausner, Justin Hayes, Apostolos Antonacopoulos, and Stefan Pletschacher. 2017. Unearthing the Recent Past: Digitising and Understanding Statistical Information from Census Tables. In Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage (DATeCH2017). ACM, New York, NY, USA, 149-154. DOI: https://doi.org/10.1145/3078081.3078106

[27] C. Sanjaya, M. liana and A. Widodo, "Revenue Prediction Using Artificial Neural Network," 2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies, Jakarta, 2010, pp. 97-99.doi: 10.1109/ACT.2010.53

[28] A. Kibekbaev and E. Duman, "Benchmarking Regression Algorithms for Income Prediction Modeling," 2015 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, 2015, pp. 180-185.doi: 10.1109/CSCI.2015.162

[29] Joey Chiao-Yin Hsiao, Carol Moser, Sarita Schoenebeck, and Tawanna R. Dillahunt. 2018. The Role of Demographics, Trust, Computer Self-efficacy, and Ease of Use in the Sharing Economy. In Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies (COMPASS '18). ACM, New York, NY, USA, Article 37, 11 pages. DOI: https://doi.org/10.1145/3209811.3209816

[30] Web page link "Rdocumentation" https://www.rdocumentation.org/packages/caret/versions/3.45/topics/confusionMatrix

[31] Tauseef Ahmad " Income Level Prediction using US census data" Webpage url https://medium.com/@tauseefahmad12/income-level-prediction-using-us-census-data-63cf8d44ed0

[32] J Fritz Github project adult_classification Url link : https://github.com/JFritz227/adult_classification

[33] "Artificial neural network" Wikipedia webpage Url link "https://en.wikipedia.org/wiki/Artificial_neural_network"
[34] Savan Patel article on "Support Vector Machine" Webpage url "https://medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72"
[35] Webpage article by Statistics Solutions, 2019, Url Link:"https://www.statisticssolutions.com/what-is-logistic-regression/"
[36] Wikipedia article "k-nearest neighbors algorithm" webpage URL "https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm"
[37] "Predictive Analysis" article by SAS solutions URL "https://www.sas.com/en_ca/insights/analytics/predictive-analytics.html"
[38] Sriram parthasarathy article on " the 4 common challenges of predictive analysis" july 10, 2018 webpage url "https://www.logianalytics.com/predictive-analytics/the-4-common-challenges-of-predictive-analytics/"
[39] Sudharsan Asaithambi article on "Why, How and When to Scale your Features" URL : https://medium.com/greyatom/why-how-and-when-to-scale-your-features-4b30ab09db5e
[40] Divya Rajprasad Webpage article "Predicting the Income Level based on Various Factors" url "https://rstudio-pubs-static.s3.amazonaws.com/230766_1628cb3f6e624ad3aec435f5821185d7.html"
[41] Sulman Khan "Analytics Edge: Unit 4 - Predicting Earnings from Census Data" URL LINK "https://rpubs.com/SulmanKhan/437149"
[42] Ronny Kohavi and Barry Becker, Data Mining and Visualization, Silicon Graphics. UCI machine learning repository source "Adult data set" https://archive.ics.uci.edu/ml/datasets/adult.
[43] Devin Soni webpage article "Dealing with Imbalanced Classes in Machine Learning" Url link "https://towardsdatascience.com/dealing-with-imbalanced-classes-in-machine-learning-d43d6fa19d2"
[44] Wikipedia article on "Mutlicollineraity" url link https://en.wikipedia.org/wiki/Multicollinearity
[45] Abhinav Singh "Predicting Income Level, An Analytics Casestudy in R" Webpage url link "https://cloudxlab.com/blog/predicting-income-level-case-study-r/"
[46] Liu, Huan et al. "Feature Selection: An Ever Evolving Frontier in Data Mining." FSDM (2010).
[47] Wikipedia article on "Random forest" https://en.wikipedia.org/wiki/Random_forest.
[48] Harshdeep Singh "Understanding Gradient Boosting Machines" webpage link url : https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab
[49] Webpage by Rdocumentation url link : "https://www.rdocumentation.org/packages/caret/versions/3.16/topics/confusionMatrix"
[50] Matthew Baumer "NeuralNetwork" webpage link url :" https://rpubs.com/mbaumer/NeuralNetworks"
[51] Badreesh Shetty "Supervised Machine Learning: Classification" url link : "https://towardsdatascience.com/supervised-machine-learning-classification-5e685fe18a6d"
[52] Ashish dutt "Learning a classifier from census data" webpage url "https://duttashi.github.io/blog/learning-a-classifier-from-census-data/"
[53] Jason Brownlee "Boosting and AdaBoost for Machine Learning" url link "https://machinelearningmastery.com/boosting-and-adaboost-for-machine-learning/"
[54] Sunil Ray "Quick Introduction to Boosting Algorithms in Machine Learning" webpage url https://www.analyticsvidhya.com/blog/2015/11/quick-introduction-boosting-algorithms-machine-learning/.
[55] "4 Overview of Data Science Methods." National Academies of Sciences, Engineering, and Medicine. 2017. Strengthening Data Science Methods for Department of Defense Personnel and Readiness Missions. Washington, DC: The National Academies Press. doi: 10.17226/23670.
[56] Tang, J., S. Alelyani, and H. Liu. 2014. Feature selection for classification: A review. In Data Classification: Algorithms and Applications (C.C. Aggarwal, ed.). Boca Raton, Fla.: CRC Press.
[57] Roger Wohlner "Economic Indicators: Gross Domestic Product (GDP)" url link "https://www.investopedia.com/university/releases/gdp.asp"
[58] Prince Grover "Gradient Boosting from scratch" webpage url "https://medium.com/mlreview/gradient-boosting-from-scratch-1e317ae4587d"
[59] Mario Polese "Regional Economics in Canada" url link "https://www.thecanadianencyclopedia.ca/en/article/regional-economics"
[60] Yogesh Singh "Importance of Predictive Analytics: What it is & why it's Important?" url link https://www.edupristine.com/blog/importance-of-predictive-analytics
[61] Zecevicp "chapter 8 code and data; couple of corrections for chapter 7" url "https://github.com/spark-in-action/first-edition/commit/09bad5a9a1522a5a5f8a618e88e068321a92dce8"
[62] Azamat Kibekbaev and Ekrem Duman "Benchmarking regression algorithms for income prediction modeling" url link "https://www.sciencedirect.com/science/article/pii/S0306437916300151"
[63] Sas solutions Article on "Predictive Analytics" url link "https://www.sas.com/en_us/insights/analytics/predictive-analytics.html"
[64] Lasse Koskinen "Modeling and Predicting Individual Salaries: A Study of Finland's Unique Dataset" url http://actuaries.org/PBSS/Colloquia/Helsinki/Papers/Koskinen.pdf
[65] Ritvik Khanna "Comparative Study of Classifiers in predicting the Income Range of a person from a census data" url :"https://towardsdatascience.com/comparative-

study-of-classifiers-in-predicting-the-income-range-of-a-person-from-a-census-data-96ce60ee5a10”

[66] Mark popovich “census data” url :” https://git.generalassemb.ly/markpopovich/census_data ”

[67] Ajanthan Rajalingam, Damandeep Matharu, Kobi Vinayagamoorthy, and Narinderpal Ghoman “Data Mining Course Project:Income Analysis” url “http://www.cas.mcmaster.ca/~cs4tf3/project_2002/report_kobi.pdf”

[68] Ashish Dutt “Learning a classifier from census data” url “https://www.r-bloggers.com/learning-a-classifier-from-census-data/”

[69] R tutorial webpage “Estimated Logistic Regression Equation” Url link : “http://www.r-tutor.com/elementary-statistics/logistic-regression/estimated-logistic-regression-equation”

[70] Mohammed Topiwalla “Machine Learning on UCI Adult data Set Using Various Classifier Algorithms And Scaling Up The Accuracy Using Extreme Gradient Boosting” webpage url link: “https://datascience52.files.wordpress.com/2017/02/machine-learning-on-uci-adult-data-set-using-various-classifier-algorithms-and-scaling-up-the-accuracy-using-extreme-gradient-boosting.pdf”

[71] Programming guide “Time complexity explained” url link: https://programming.guide/time-complexity-explained.html.

[72] ….