DEVELOPMENT OF TRAVEL DEMAND MODEL USING MACHINE LEARNING TECHNIQUES

PROJECT REPORT

Submitted by

AKHILESH GOPINATH

REG. NO. : TKM20CETE01

to

A P J Abdul Kalam Technological University

in partial fulfilment of the requirements for the award of the Degree

of

Master of Technology

in

Transportation Engineering

DEPARTMENT OF CIVIL ENGINEERING

T.K.M. College of Engineering, Kollam

July 2022

DECLARATION

I undersigned hereby declare that the project report “Development of Travel Demand Model Using Machine Learning Techniques”, submitted for partial fulfilment of the requirements for the award of degree of Master of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Dr. Munavar Fairooz C, Assistant Professor. This submission represents my ideas in my own words and where ideas or words of others have been included; I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/ or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

KOLLAM

07.07.2022 **AKHILESH GOPINATH**

DEPARTMENT OF CIVIL ENGINEERING

T.K.M. COLLEGE OF ENGINEERING, KOLLAM

CERTIFICATE

Certified that this report entitled **'DEVELOPMENT OF TRAVEL DEMAND MODEL USING MACHINE LEARNING TECHNIQUES'** is the report of final project presented by **AKHILESH GOPINATH, Reg. No. : TKM20CETE01** during **2021-2022** in partial fulfilment of the requirements for the award of the Degree of Master of Technology in Transportation Engineering of the A P J Abdul Kalam Technological University.

Guide	Coordinator	Head of the Department
Dr. Munavar Fairooz C Assistant Professor Dept. of Civil Engineering TKMCE, Kollam	**Dr. Kavitha Madhu** Associate Professor Dept. of Civil Engineering TKMCE, Kollam	**Dr. Sajeeb R** Professor Dept. of Civil Engineering TKMCE, Kollam

ACKNOWLEDGEMENT

I take this opportunity to express my deep sense of gratitude and sincere thanks to all who helped me to complete the project successfully.

I am deeply indebted to my guide, **Dr. Munavar Fairooz C,** Assistant Professor, Department of Civil Engineering for his excellent guidance, positive criticism and valuable comments.

I am greatly thankful to my project coordinators, **Dr. Kavitha Madhu**, Associate Professor, **Prof. Meenu Tomson**, Assistant Professor and **Prof. Jijin A**, Assistant Professor, Department of Civil Engineering for their constant supervision as well as for providing necessary information regarding the seminar.

I am greatly thankful to **Dr. Sajeeb R.**, Professor Head of the Department of Civil Engineering for his kind support.

Finally, I thank my parents and friends who directly and indirectly contributed to the successful completion of my project.

AKHILESH GOPINATH

ABSTRACT

As urbanization is happening at a sky-rocketing pace, the population in the Indian cities is growing rapidly which in turn is showing huge growth in travel demand. Motor vehicle ownership has risen alarmingly in several metropolitan areas over the past few decades at a pace of roughly 9% annually. The majority of Indians travel by bus, which not only meets their transportation needs but also serves as a required tool to draw in more affluent passengers (private car users). Unfortunately, there is a lot that can be done to enhance the state of bus services, including consistency, safety, and security. To draw discerning passengers to bus service and to prevent the expanding use of private vehicles, the quality of bus service urgently needs to be improved. User perception study is a practical and proven option to understand the need of travelers and to identify the reasons that deter the choice riders from using the public mode of transport. Revealed preference data is collected to understand the present condition and stated preference data is collected from the same respondents to understand their priority areas for improvement in the service quality for the future. The data is analyzed using conventional and advanced machine learning modeling techniques. Results indicated that choice riders are willing to shift towards the public mode of transport like ordinary and premium buses if certain attributes like travel information, security, comfort level, etc. are improved. These findings provide guidance for enhancing Kolkata's bus service based on the needs of choice riders. Even though the methodology was illustrated concerning the city of Kolkata, it might be used in other locations to derive metropolitan area service design and enhance bus services.

Keywords: *choice riders, captive riders, revealed preference, stated preference*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

FSM	:	Four Stage Model
ML	:	Machine Learning
SVM	:	Support Vector Machine
MNL	:	Multinominal Logit Model
AET	:	Access and Egress Time
IVTT	:	In Vehicle Travel Time
TI	:	Travel Information
DCOM	:	Degree of Comfort
SP	:	Stated Preference
RP	:	Revealed Preference

CHAPTER 1

INTRODUCTION

1.1 Background

Any system which includes making large-scale decisions must involve modelling into the process. The system's performance is impacted by a wide range of variables. The human brain is incapable of keeping track of every participant in the system, as well as their connections and interdependencies. Therefore, we turn to models that are both simple and complicated enough just to replicate important interactions in actuality. Modeling may be mathematical, symbolic, or physical. Models' focus on isolating crucial interactions rather than replicating the complete structure is an essential element. The study of how people act when making decisions about the provision and use of transportation is known as transport modelling. Transport modelling techniques, contrary to other engineering models, have developed from a variety of fields, including economics, psychology, geography, sociology, and statistics. By using an appropriate zone system, travel demand modelling seeks to explicitly establish the spatial distribution of travel. Thus, forecasting people's desired travel choices based on the generalized travel costs of various options is implied by demand modelling. The fundamental choices are selecting a destination, a mode of transportation, and a route. Despite the use of many modelling strategies, we will just cover the traditional transport model, also known as the four-stage model (FSM).

The results of a travel demand model can be used to inform transportation planning decisions. The model's output varies depending on the ideas and data incorporated as well as the complexity of the specific model. For roads with functional classes of minor arterial and above, small models usually offer consumers predicted highway volumes. Users generally access everything from local model regions and transit predictions in large model regions. Furthermore, some more advanced models offer consumers data on travel behavior influenced by toll methods, college/university trips, and vehicle forecasts. Models of travel demand are not meant to predict bicycle or pedestrian trips owing to their aggregate nature and regional orientation.

1.2 Four-Stage Model

The four sub-models that make up the classic model are trip assignment, modal split, trip distribution, and trip generation. As shown in figure 1.1, the framework starts by defining the research region, sectioning it into multiple zones, and taking into account the system's total transportation network. The database also includes details of each zone's current (base year) population density and economic activities, like employment, retail space, educational opportunities, and leisure amenities. The model for trip generation is then constructed utilizing the available data to determine the total number of trips drawn to and produced by each zone. The next stage is to disperse these trips from each zone to several alternative destination sectors inside the research region using trip distribution models. The next stage uses modal split models to divide the trips into several stages according to modal attributes. In effect, this is reducing the trip matrix generated for various modes together into trip matrix unique to that mode.

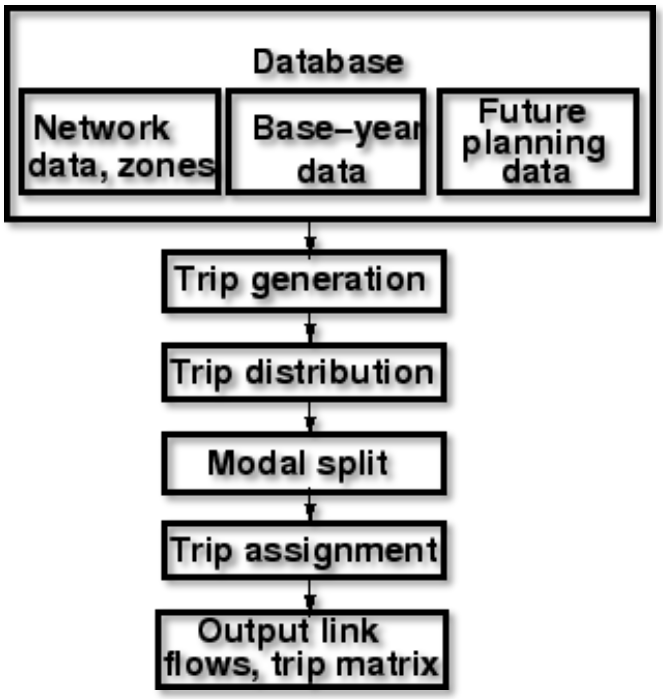

Figure 1.1: General form of the four stage modeling (Source: civil.iitb.co.in)

Each trip matrix is then assigned to the road system associated with that particular mode by using trip assignment models. The process will display the volume of traffic created on each network link. It analyses the trips zone by zone, splits the trips into modes based on the preferences of the passengers, and then assigns the trips to the

system. This method makes it easier to comprehend how alterations to transportation networks in the future might affect how people move and also how societal preferences might alter how networks flow.

1.3 Objective

The objectives of the proposed study are as given below:

- To develop a mode choice model using MNL model based on choice data.
- To develop a choice model using Machine Learning Techniques.
- To conduct a comparison study among different techniques and select the most suitable one based on prediction and accuracy for the modeling purpose.

1.4 Methodology

The methodology planned to achieve the mentioned objectives is described below:

- **Study of literature-** A background study was done on the different model developments and machine learning techniques.
- **Data collection-** Kolkata-based secondary data was collected from choice-based travellers using a questionnaire survey. The data consist of user perception responses of travellers using Ordinary buses, Premium buses, Taxis, and Private cars.
- **Analysis and Model Development -** To create a digital database for a model estimate, the replies gathered from choice riders were encoded. While the qualitative aspects like security setup, traffic information, and comfort were dummy coded, the quantitative attributes like access and egress time, waiting time, in-vehicle travel time, fare, and span of service were entered in cardinal linear form. Conventional modeling has been done using the Multinominal Logit Model in NLOGIT and an advanced model has been made with ML techniques such as Perceptron, SVM, Decision trees, and Random Forest using Python Code.
- **Interpretation of Results-** The coefficients obtained from the MNL model, predictions, and classification of mode choice were interpreted.

1.5 Scope

The demand model is limited to the secondary data obtained from Kolkata city. The modes are limited to Premium Buses, Ordinary Buses, Taxis, and Private Cars. A general mode choice model is limited to the Multinomial Logit Model for the conventional technique and Machine Learning like Perceptron, SVM, Decision Trees, and Random Forest for advanced techniques.

1.4 Study Area

At 24,000 people per square kilometre, Kolkata has one of India's highest population densities. Approximately 6% of the city's entire territory is covered by its meagre road network (Dandapat, 2017). With limited road space and growing travel demand, the city is experiencing severe congestion, vehicular emission, and delay. In Kolkata, buses are the main form of public transportation. In reality, the majority of travellers in Kolkata—about 60%—travel by bus (Dey, 2012). Bus service is widely available across the city and is primarily provided by local buses.

Ordinary service in Kolkata is operated by both private and Government operators. However, the Government controls the service in terms of fare, route permit, etc., for all ordinary routes. The key operator controlling the operation of bus service in Kolkata is West Bengal Transport Corporation (WBTC) (Figure 1.2). Apart from ordinary service, premium service is also operated by WBTC.

Figure 1.2: Types of Buses (CSTC Ordinary and WBSTC Premium Services)

(Source: https://indianexpress.com)

1.5 Organization Of Report

The thesis is structured into seven main chapters.

Chapter one describes the background of the study, problem statement, objectives, and scope of the study.

Chapter two reports the literature review on the various models, mode splits models such as simple binary logit model, multinominal logit model, and machine learning techniques such as Perceptron, SVM, Decision trees, etc.

Chapter three discusses the research methodology. Data collection and analysis methods adopted for the study are described in this chapter. It discusses the questionnaire survey adopted to study the user perception regarding the service quality of public transport in Kolkata.

Chapter four presents the identification of mode choice models both conventional and advanced for prediction and classification as per the collected data and modeled database.

Chapter five reports the analysis of the model i.e. the output coefficients obtained from NLOGIT software in MNL modeling and accuracy and classification from ML techniques coded using Python script.

Chapter six presents the overall general conclusion and conclusion concerning the models and outputs.

CHAPTER 2
LITERATURE REVIEW

2.1 Introduction

This chapter presents a critical review of the previous research across the world, on the various transportation/non-transportation models created using conventional and advanced machine learning (ML) techniques. The chapter also discusses the opportunities of ML techniques in general which can be incorporated to solve transportation-related challenges irrespective of the study area.

Current revelations in existing technologies and the growth of new information technologies have given choices about travel demand new dimensions. To distinguish and model the mobility, activity, and travel decisions of businesses and developers along with those of persons and households, and also to design a system of models that decision-makers and planners can use to analyze the influence of changes to the transportation system and the progression of information technologies (e.g. various telecommuting, Intelligent Transportation Systems, etc.). This approach anticipates the implementation of an operational model system as a stage method starting with the current "best practice" of heterogeneous travel demand model systems.

2.2 Modal Split Models

2.2.1 Theoretical framework

A behavioral model is an example of the choices travelers make when provided with different alternatives. These choices are based on the conditions under which the different modes of transportation are offered, i.e., the distances, prices, and other level-of-service characteristics of the contending alternate mode of transportation.

An individual is shown as selecting a mode that maximizes their usefulness (Ben-Akiva and Lerman, 1985). The attraction that a user relates with a particular trip is the concept of a transportation mode's utility. As a result, the user is seen selecting the mode with the highest attraction based on various parameters, like transfer time, in-vehicle travel time, access time to the transit point, wait period for the mode to reach the access point, travel expenses, parking charges, etc. All of the travel demand models presented in this section are supported by the theory known as utility maximization.

2.2.2 Logit models

Since they have the capability to mathematically describe complex travel patterns of any population, logit models are the most widely utilized modal split models in the field of transportation planning. The chance of a person *i* choosing a mode *n* out of the M total available modes is given as, to briefly describe the framework:

$$\mathbf{P_{in}} = \frac{\exp(V_{in})}{\sum_{m \varepsilon M} \exp(V_{im})} \qquad (2.1)$$

where, V_{in}: utility function of mode *n* for individual *i*

V_{im}: utility function of any mode *m* in the choice set for an individual *i*

P_{in}: probability of individual *i* selecting mode *n*

M: total number of available travelling modes in the choice set

Logit models can be classified into two categories in general; they are binary and multinomial logit models. Multinomial logit models imply a greater range of options, whereas binary choice models can only model with two discrete choices, i.e., the individual has only two feasible alternatives for selection (Khan et al., 2007).

Binary logit models

Binary logit models are used when there are just two possibilities available overall, or when M = 2. Figure 2.1 beneath illustrates a binary logit model with the decision set consisting of the options bus and car.

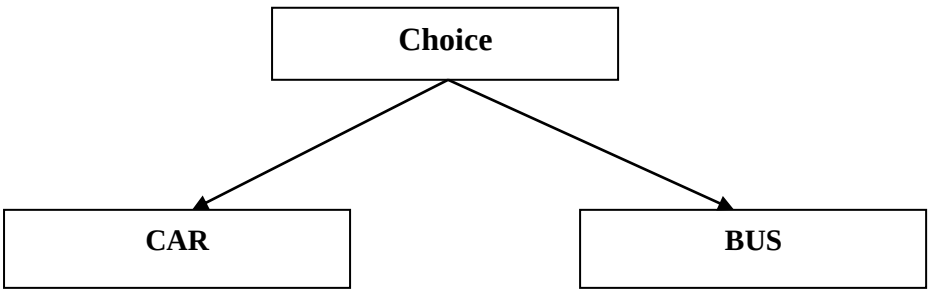

Figure 2.1: Simple Binary Logit Model

Nested Binary Logit Model

The major drawback of the above model is that it is supposed to be applicable if in the choice set the mode alternatives are autonomous of each other.

In these scenarios, a nested logit model that permits correlation between the values of the options in common groups can be utilized to relieve the limitations of the simple logit models. All the groups of associated choices are arranged in tiers or nests to provide the framework of a nested logit model. Each nest is depicted by a composite choice that interacts with the other options present for an individual. Figure 2.2 shows an illustration of a nested logit model, which is a sequel of Figure 2.1, by nesting the two fundamental and similar modes of public transportation, the Ordinary Bus, and Premium Bus.

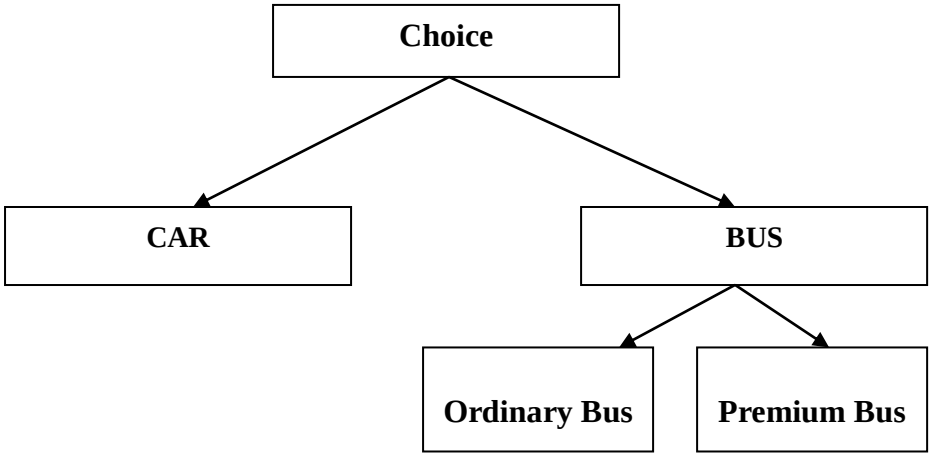

Figure 2.2: Nested Binary Logit Model

Multinominal Logit Models

Based on the features of the potential trip options in the option set, multinomial logit models are likewise divided into basic and nested multinomial logit models, much like binary logit models. The representation of simple multinomial logit model is shown below in Figure 2.3.

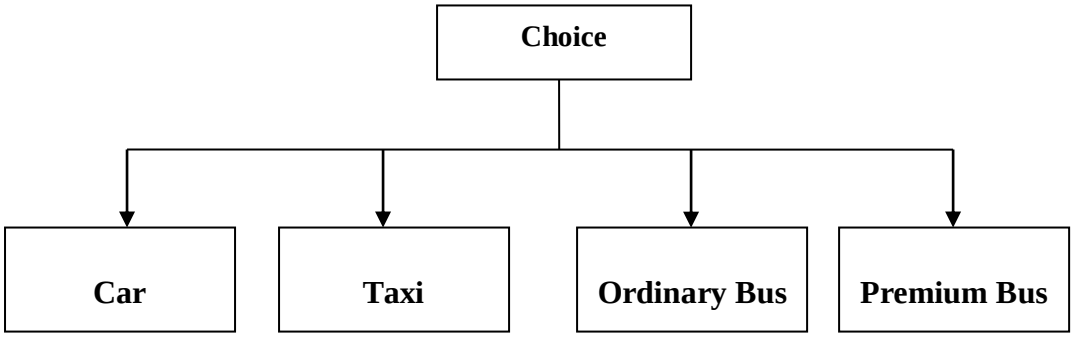

Figure 2.3: Multinominal Logit Model

In order to predict the future of travel patterns of the research area, the size of the choice set specified helps in the selection of a suitable mode choice model. A binary modal split model can be used if the decision set includes two forms of transportation or two groups of modes of transportation (Khan et al., 2007). For larger choice sets, on the other hand, multinomial modal split models could be chosen.

2.3 Machine Learning Techniques

Machine learning techniques are a vital component of implementing smart transportation in past few years. It involves identifying patterns in the data (Koushik et al., 2020). The intricate relationships between roads, transportation traffic, environmental factors, and traffic crashes are investigated during this scenario through the employment of an upgraded deep learning approach. The intricate relationships between roads, transportation traffic, environmental factors, and traffic crashes are investigated during this scenario through the employment of an upgraded deep learning approach. The proposed model consists of two main parts: an unsupervised feature learning module to connect the explanatory factors and feature representations, and a supervised module for predicting traffic crashes. (Tizghadam et al., 2019). How machine-learning classifiers can outperform logit models in classification and prediction is not surprising. Machine learning allows for much more adjustable model structures as opposed to logit models, which have predetermined model structures. This can help lessen the model's inaptness with empirical data (Xie et al., 2003; Bishop and Nasrabadi, 2006).

An ML algorithm's job can be summarized in basic as follows: Given a set of input vectors and the corresponding target output vectors, the goal is to train a functional relationship between the input and output vectors (Bhattacharya et al., 2007).

2.3.1 Classification of Machine Learning

In machine learning, four major categories that are recognized: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning (Russell & Norvig, 2009). Each data point in supervised learning is made up of some input parameters (referred to as features) and related output variables (called label or target variables). Regression is the name for supervised learning when the output variable is

continuous and has a range of possible values. Supervised learning is referred to as classification if the output variable is discrete and can only accept a finite number of values that indicate classes. Figure 2.4 provides an overview of classification and sub-classification of ML with examples.

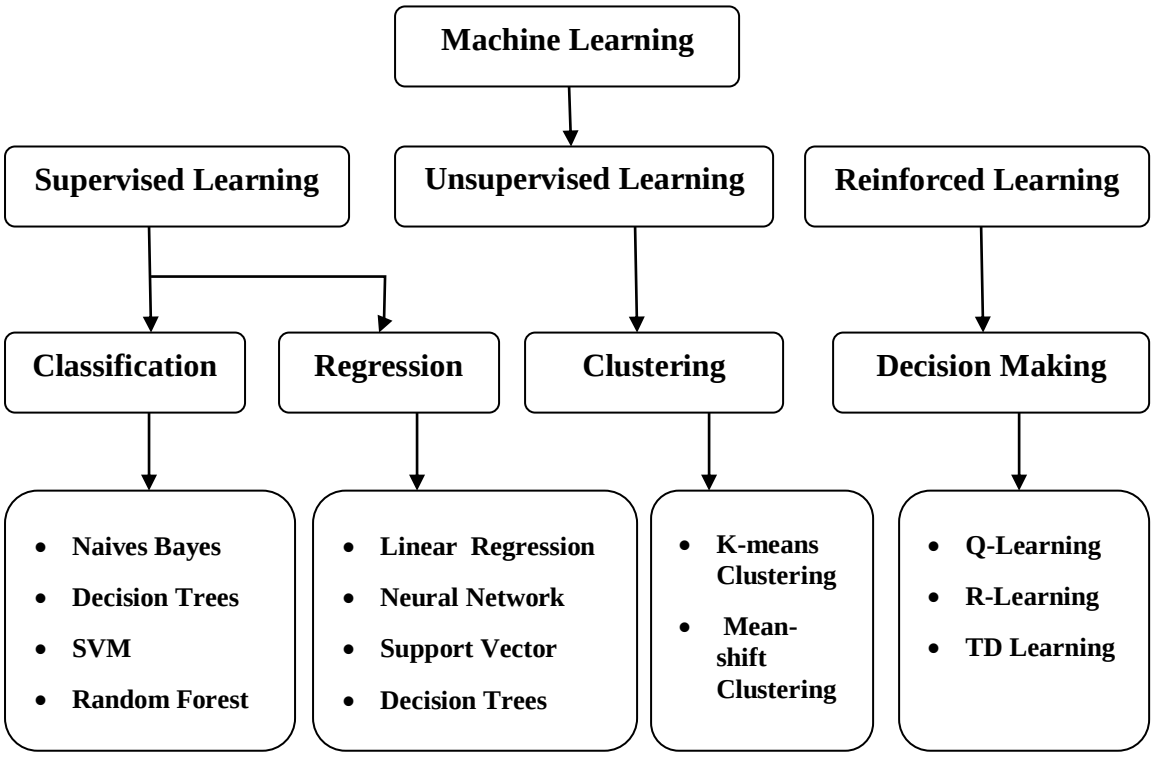

Figure 2.4: Classification Tree of Machine Learning

Supervised learning

Supervised Machine Learning (SML) is the pursuit of algorithms that draw broad hypotheses from situations provided externally, and then forecast circumstances in the future (Osisanwo et al., 2017). The major supervised learning that deals with classification as per (Taiwo, 2010) are: Linear Classifiers, Logistic Regression, Naïve Bayes Classifier, Perceptron, Support Vector Machine; Boosting, Decision Tree, Random Forest (RF); Neural networks, Bayesian Networks, and so on. In SML we have to input variables (X) and an output variable (Y). To learn the function that maps input to output and ML method is used. Figure 2.5 provides a clear idea of its working algorithm. It is called supervised because the process is learned from the training set and can be imagined as a teacher who is supervising the learning process.

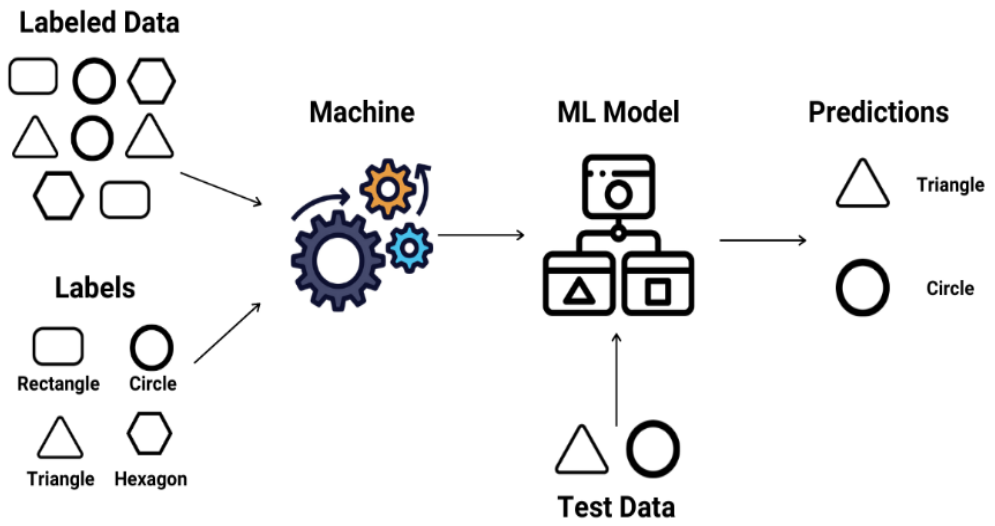

Figure 2.5: Supervised Learning (Source: Enjoy Algorithm)

Unsupervised Learning

In an unsupervised learning model, the computer learns from an unlabeled dataset and makes sense of it by identifying features, co-occurrences, and underlying patterns. X is the input; however, there are no equivalent output variables in this. In this case, the machine tries to create a phony output before learning mapping functions. In essence, they attempt to use the underlying traits found in the input data as their output data. This method is mostly employed to conduct a more thorough data analysis. Figure 2.6 gives a general idea of unsupervised learning's working.

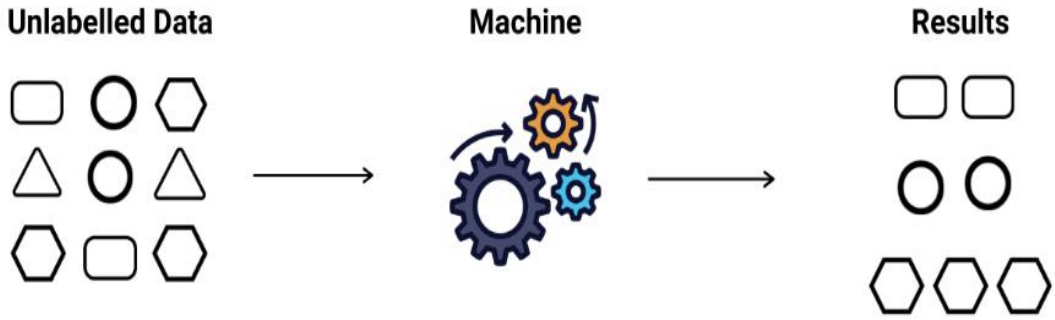

Figure 2.6: Unsupervised Learning (Source: Enjoy Algorithm)

Reinforcement Learning

Understanding a mapping from events to behaviors to optimize a scalar reward is known as reinforcement learning. In contrast to most machine learning methods, this one requires the learner to experiment with different actions to determine which ones result in the greatest reward. In the most intriguing and difficult situations,

choices can have an impact on not only the immediate reward but also the scenario that follows and all rewards that follow that (Sutton, 1992). The two most crucial traits that set reinforcement learning apart from other types of learning are trial-and-error searching and delayed rewards. Figure 2.7 shows the workflow of reinforcement learning.

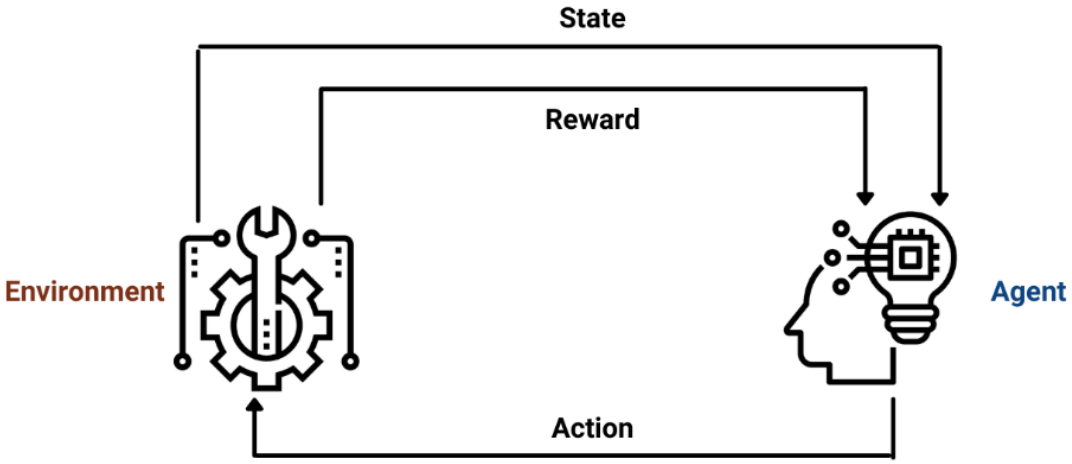

Figure 2.7: Reinforcement Learning (Source: Enjoy Algorithm)

CHAPTER 3
METHODOLOGY

3.1 General

The overall methodology adopted for fulfilling the objective and scope of the present work includes three major tasks as shown in Figure 3.1. The preliminary task is to do a thorough literature review to have a strong base idea about machine learning and the conventional method of travel behavior analysis using Stated Preference and Revealed Preference data. Collection of secondary data and development of a mode choice model in MNL using secondary data is the second task. The final task is to generate ML models using different techniques and conduct a comparative study to identify the most suitable ML technique. Figure 3.1 below shows the flow chart of how the study was carried forward.

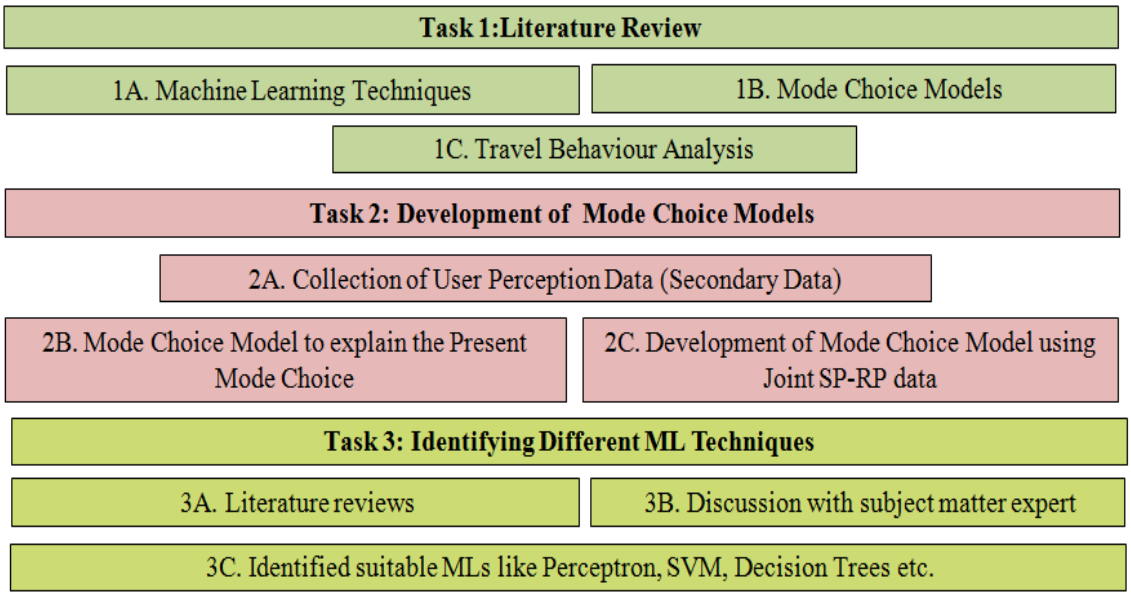

Figure 3.1: Research Methodology

- **Study of Literature-** A background study was done on the different model developments and machine learning techniques.
- **Data Collection and Development of Mode Choice Models-** Kolkata-based secondary data was collected from choice-based travellers using a questionnaire survey. The data consist of user perception responses of travellers using Ordinary buses, Premium buses, Taxis, and Private cars.

- **Analysis-** To create a digital database for the model estimate, the responses gathered from choice riders were coded. While the qualitative attributes like security setup, traffic data, and comfort were dummy coded, the quantitative attributes like access and egress time, waiting time, in-vehicle travel time, fare, and span of operation were entered in cardinal linear form. Following this conventional modeling is done in the MNL model using NLOGIT software. Next, different supervised learning techniques such as Perceptron, SVM, Decision Trees, and Random Forest models are created using Python coding to predict and classify the mode choice. Finally, a comparison is done among these techniques based on their accuracy and data handling capacity to find the most suitable ML technique.
- **Interpretation of Results**

  Accuracy results are interpreted and suitable ML is chosen among others and also the conventional technique.

3.2 Data Collection

Many transportation studies have used revealed preference (RP) and stated preference (SP) surveys because they offer useful data for analyzing how travellers make decisions about setting priorities and network modifications (Borjesson, 2008; Peer et al., 2013). Whilst SP surveys typically subject passengers to a variety of fictitious scenarios and log their decisions under varied conditions, RP surveys collect genuine travel data from the respondents. Depending on the characteristics of the subject, the alternative qualities, and its choices (preferences) are recorded by the survey, discrete choice models can then be generated.

3.2.1 Stated preference (SP) data

It is generally used to test travellers' responses to a scenario that has not been implemented at the present and the choices and options given are hypothetical. For example: If a new bus route has to be started. The scenarios and alternatives need to be designed carefully as the responses from people may be misleading at times. People may choose the most ideal scenario at times but in reality, when the project or work is in the implementation stage most of the respondents may not use the facility because of financial

constraints or any other reason. According to a study by (Brownstone et al., 2000), it was observed that most of the drivers preferred to choose vehicles causing less environmental effects, and their responses showed the same, however in the ground scenario their behaviour was different. Figure 3.2 shows a sample questionnaire used for collecting SP data of hypothetical scenarios shown to them.

E: Stated Preference Survey

Choice set to be presented to the trip maker depending on his/her predominant mode (*Car, Taxi, Ordinary Bus, and Premium Bus*) and trip length (*Short trip if trip length is equal to or less than 6 km and Long trip if trip length is greater than 6km*) of current/most recent trip.

Stated Preference Survey of Car users (Long trip)

Attributes	Current Choice: Car	Car	Ordinary bus
In Vehicle Travel time (minute)	T=		
Fare (INR)	F=		
Headway (minute)			
Span of Operation (hours)			
Security			
Information			
Given the Alternative Modal Characteristics presented above which option would you choose?		Car	Ordinary Bus

Rate the acceptability of the Given Fare of Bus Service in 1-5 Scale:

Unacceptable	Slightly Unacceptable	Neutral	Slightly Acceptable	Acceptable

What is an acceptable fare for you to pay for the Improved Bus Service? _____

In the future, what would be your propensity to make trip using the option selected above?

Willingness to comply	Very Weak	Moderately Weak	Neutral	Moderately Strong	Very Strong

Figure 3.2: Sample of the SP Questionnaire

3.2.2 Revealed preference (RP) data

RP study deals with the choices of respondents in real in the existing condition that is a travel mode for a particular trip. It gives an idea of the present scenario to go for alternatives revolving around the present condition for future development. RP is limited to existing alternatives and therefore. In figure 3.3 we can see a sample RP questionnaire used to collect data from respondents regarding the current scenario.

D: Revealed Preference Data

Attributes \ Alternatives	Premium Bus	Ordinary Bus	Private Car	Taxi
Access & Egress Time (minute)*				
Headway (minute)*				
In vehicle Travel Time (minute)*				
Fare (Rupees)*				
Comfort (Level 1/2/3/4)				

Figure 3.3: Sample of the RP Questionnaire

After looking at the advantages and disadvantages of SP and RP data alone, a lot of studies have been done in the past decade to find the potential of combining both RP and SP data for modeling. The advantages of joint SP-RP models are huge and clear, however, the technicality of the same is not easy. For this project around 1400 responses are used for different modeling purposes (both conventional and advanced).

CHAPTER 4
DEVELOPMENT OF MODE CHOICE MODEL

4.1 Conventional Model

As part of the work, Multinominal Logit Model has been generated using NLOGIT software as shown in Figure 4.1. Here the new project is opened using the project tab and the excel data is fed into the software. Then inside the model tab we create our required model such as nest binary or discrete etc.

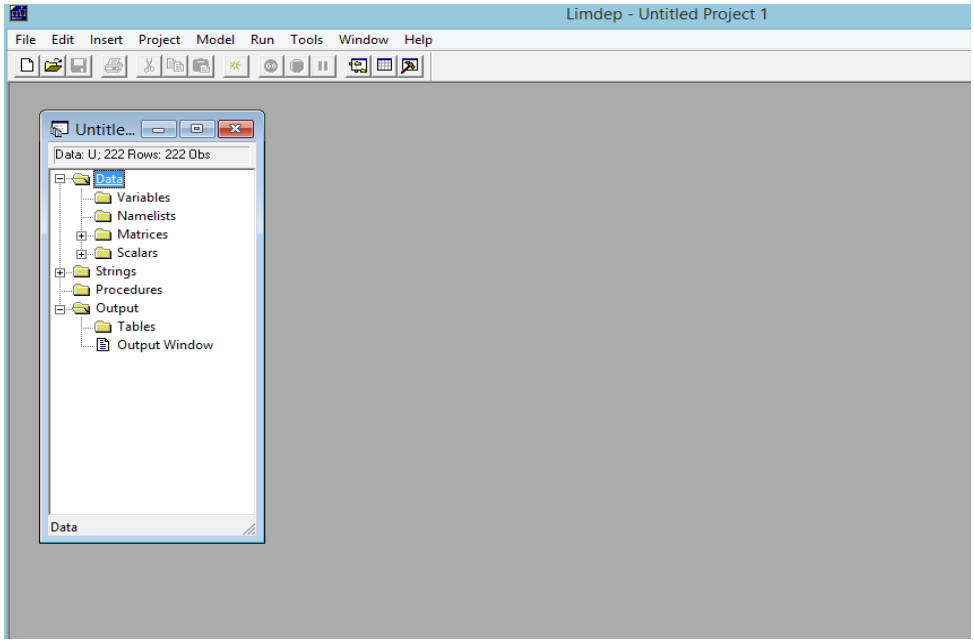

Figure 4.1: NLOGIT Software

Basic steps used in Modeling include preparation of data set using cardinal coding and dummy coding for quantitative and qualitative attributes respectively. The data set is read into the system and input attributes are selected. Values are run in the software to give desired outputs. The MNL output is then interpreted for the magnitude of the value and its sign.

4.2 Development of Machine Learning Algorithm

It is crucial to remember that the accuracy, effectiveness, and robustness of the machine learning algorithm are significantly influenced by the quality, kind, and data size in both supervised and unsupervised learning. Even though every machine learning application aims to collect actuality and model uncertainty, the learned model

typically only captures the reality that is shown by the data set rather than the actual world. A common step-by-step procedure for developing a machine learning algorithm is shown in the flowchart in Figure 4.2. As shown in the graphic, data pre-processing and learning for any machine learning application depend on how real-world problems are characterized and the data is gathered.

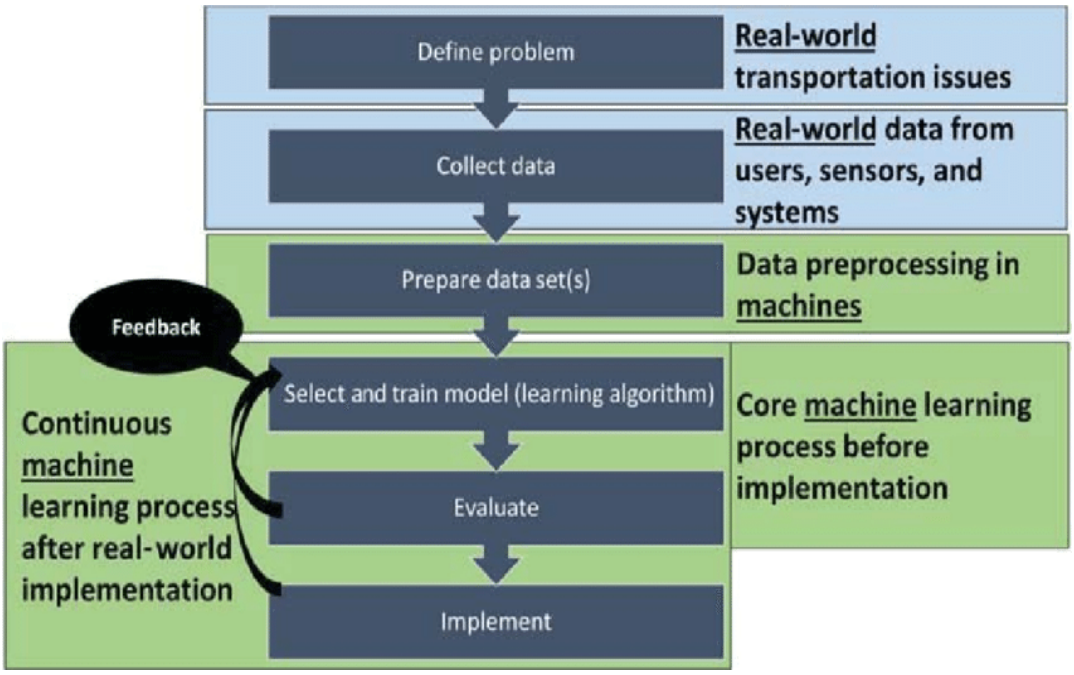

Figure 4.2: Machine Learning Algorithm Development Approach
(Source: Bhavasar et al., 2017)

4.2.1 Problem definition

Any machine learning application must start by defining the problem, which has a direct impact on all subsequent crucial steps up until the model is calculated. The fundamental inquiries that must be made to characterize a problem and use the proper machine learning technique are listed below.

- What input and output factors should a learning system take into account?
- In what categories of machine learning techniques (such as classification, clustering, and regression) are we interested?
- Do the issue and the data fall into the category of (un-, semi-)supervised learning techniques?
- In the learning system, what data size and format will be utilized?

4.2.2 Data collection

The desired outcome for the application is specified in the problem definition; nevertheless, determining the problem's inputs is based on a range of factors. For instance, the output value should be the average speed of the portion soon to offer correct travel time data for a certain highway stretch. But it's not clear from this information which input variables need to be taken into account. So, the first step in gathering data is to create a list of possible input and output variables. Although there are no specific guidelines for creating this list, a transportation engineer's basic understanding of the transportation system, user data, and user behaviour is crucial.

Understanding how much data is necessary is the second phase in the data gathering process. There are no set guidelines for responding to this issue. A typical procedure is to choose the volume of data set that reflects enough variety of the reality for the learning algorithm to be effective almost all of the time; however, academics have created particular criteria that are technique specific. Usually, time, available resources, and informed guesses are significant factors. Yet, it is possible to produce a feedback mechanism that assesses and trains to learn algorithms after their application in the actual world, as shown in Figure 4.2. However, the additional running time and system complexity of these solutions come at a cost.

4.2.3 Data fusion

Machine learning applications' effectiveness and efficiency rely on the caliber and range of information resources that the learning algorithm takes into account. Utilizing data fusion techniques, it is possible to obtain hidden core knowledge, fuse and merges many data sets into a singular, integrated, and organized data set strategically, and improves prediction accuracy beyond that which is attainable with any one of the separate data sets, and more.

4.2.4 Data pre-processing

The purpose of data pre-processing, which is a crucial stage, is to eliminate

the noise from the raw information and convert it into a format that will limit the possible number of numerical errors in difficult mathematical operations. Any set of data has noise, which is made up of data trends containing measurement inaccuracies and flaws brought on by uncalibrated data gathering equipment. These errors can have a major effect on learning. One of the most common problems in this kind of scientific calculation is a numerical mistake. Examples comprise rounding and truncation errors, sensitivity issues, conditioning issues, and issues with machine accuracy. The noise can be diminished or eliminated using a variety of filtering techniques.

4.3 Perceptron Machine Learning

For binary classification problems, the Perceptron is a linear machine learning technique. One of the fundamental forms of artificial neural networks might be this one. Although it obviously isn't deep learning, it is unquestionably an important first step. Figure 4.3 shows algorithmic diagram of Perceptron.

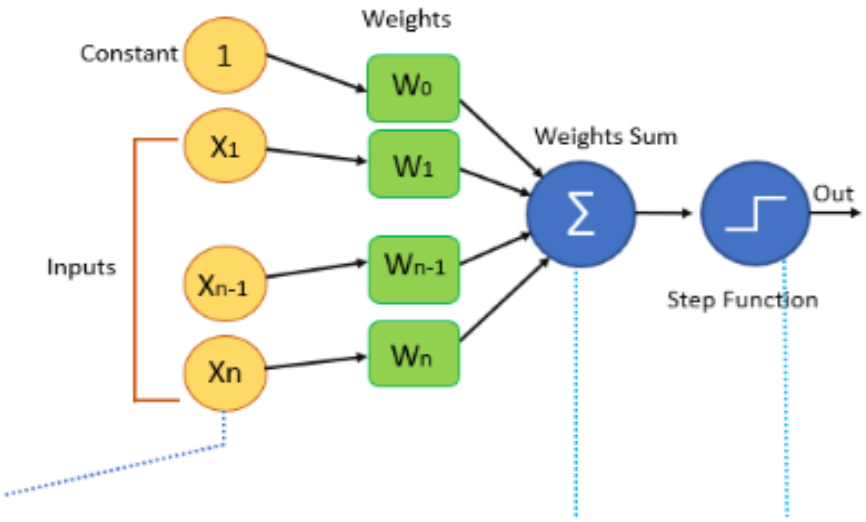

Figure 4.3: Perceptron algorithm Block
(Source: www.educba.com)

Inputs, X1, X2..., Xn are attributes of the model. Weights, W0, W1... Wn is random values that are generated during the training of the model. The weight parameter shows how strongly the units are connected. This is yet another crucial characteristic of Perceptron parts. The effectiveness of the related input neuron in determining the output is directly proportional to weight. To construct the weighted

total, the Perceptron model first multiplies all of the input values and their associated weights. The activation function "f" is then used with this weighted sum to get the desired result. The letter "f" stands for the step function, another name for this activation function.

4.4 Support Vector Machine (SVM)

One of the most well-liked supervised learning algorithms, Support Vector Machine, or SVM, is used to solve Classification and Regression problems but it is largely employed in Machine Learning Classification issues. The goal of the SVM algorithm is to find the best fit line or decision boundary that can categorize n-dimensional space, enabling us to quickly classify new data points in the future.

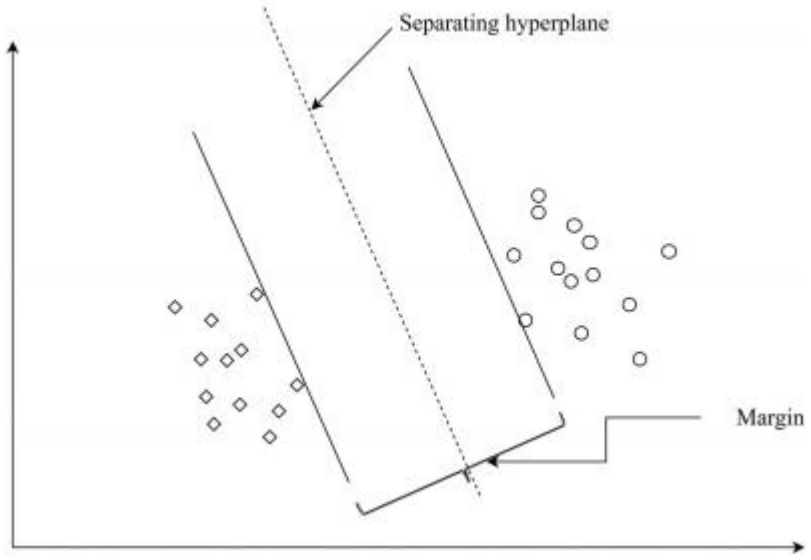

Figure 4.4: Support Vector Machine (Source: Koushik et al., 2020)

A hyperplane is a name given to this optimal decision boundary (Figure 4.4). In comparison to Neural Networks, SVM is significantly faster and computationally more cost efficient. SVM has been employed in numerous activity-travel behaviour researches as they are as accurate as the top classifiers. As with neural networks, mode choice models are common (Tang et al., 2018; Weng et al., 2018), and more recently, SVM implementation can be seen in the navigation data mining sector. SVM is used in other contexts by (Lin et al., 2009) to categorize people's lifestyles. They create clusters based on thorough activity travel data and make an effort to forecast lifestyle clusters from census figures. They also investigate how information of day's past actions affects forecasting of the next action. According to

their research, the performance of the model could be increased while simultaneously lowering the computational cost by feeding the SVM only those features that the MNL model determined to be meaningful. A drawback of SVM is that if it is supplied with several unnecessary characteristics, over fitting may cause the performance to decline (Allahviranloo & Recker, 2013). Since the development of Deep Learning and Random Forest algorithms, SVM utilization has decreased.

4.5 Decision Trees (DT)

A supervised learning method called a decision tree can be used to solve regression and classification problems, but it is typically favoured for doing so. It is a tree-structured classifier, where internal nodes stand in for a dataset's characteristics, branches for the decision-making process, and each leaf node for the classification results. The Decision Node and Leaf Node are the two nodes of a decision tree (Figure 4.5).

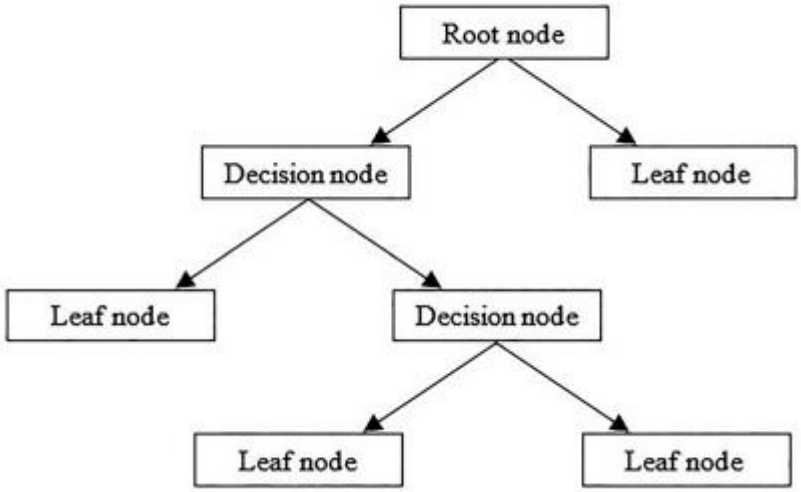

Figure 4.5: Decision Trees (Source: Koushik et al., 2020)

While a Leaf node is the result of judgments and doesn't have any more branches, Decision nodes are used to create decisions and have different branches. DT differs from other ML techniques in that they are more interpretable. The designer can comprehend the steps taken by the tree to reach a decision. Some research merely uses DT to comprehend the significance of and connections among variables. Decision trees can also be used to analyze continuous and categorical data (Kelleher et al., 2015).

4.6 Random Forest (RF)

Widely used machine learning algorithm Random Forest is a part of the supervised learning methodology. It can be applied to ML issues involving both classification and regression. It is built on the idea of ensemble learning, which is a method of integrating various classifiers to address difficult issues and enhance model performance. The Random Forest classifier, as its name suggests, averages decision trees based on various subsets of the supplied dataset to improve the dataset's predictive accuracy. The random forest uses predictions from each decision tree and predicts the outcome based on the votes of the majority of projections rather than relying solely on one decision tree. Figure 4.6 below shows algorithmic diagram of Random Forest.

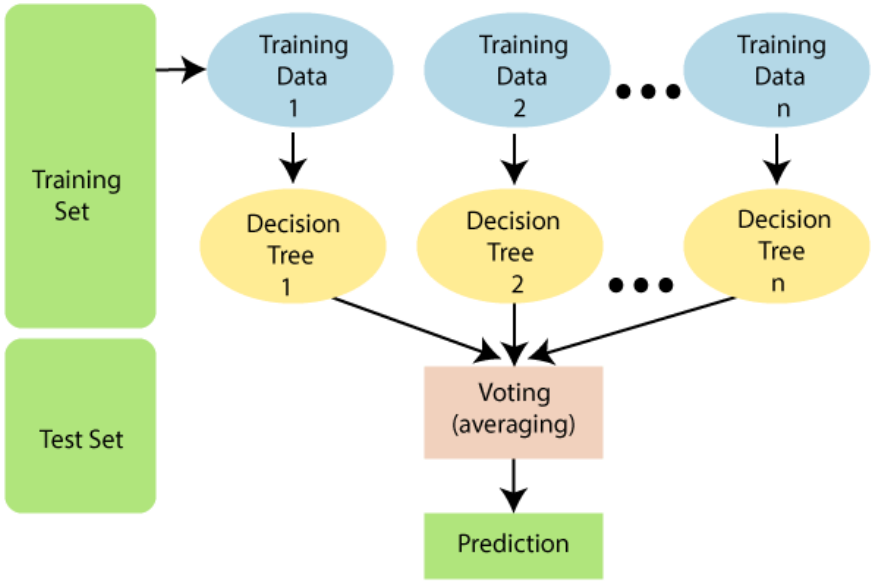

Figure 4.6: Random Forest (Source: www.javapoint.com)

The reasons why Random Forest is more preferred over other algorithms is that it takes less training time, predicts output with high accuracy irrespective of the size of the dataset, and can maintain its accuracy even if a huge volume of data is missing.

CHAPTER 5
ANALYSIS AND INFERENCES

5.1 Analysis of Multinominal Logit Model

For the analysis of the model, a total of fourteen attributes are considered of which five attributes are quantitative attributes which are access-egress time (AET), in-vehicle travel time (IVTT), headway (H), the span of operation, and fare. The qualitative attributes are defined with their levels in Table 5.1.

Table 5.1: Qualitative Attributes and Their Levels

Attributes	Level Description
Security arrangement (SE)	Level I : No security arrangement
	Level II : CCTV Surveillance at bus only
	Level III : CCTV Surveillance at bus stop only
	Level IV : CCTV Surveillance at bus and bus stops
Traffic information (TI)	Level I : Static information at bus stops
	Level II : Static information at bus stops and Web/mobile based static information
	Level III : Dynamic information at bus stop and Web/mobile based static information
	Level IV : Dynamic information at bus stops and Web/mobile based dynamic information
Degree of Comfort (DCOM)	Level I : Standing in overcrowded condition
	Level II : Standing in crowded condition
	Level III : Comfortable standing
	Level IV : Occupying seat

Headway is defined as the time intervals between two successive services to a particular route. The responses collected from choice riders were coded in MS Excel to develop a digital database for model estimation. SPOB in excel means stated preference Ordinary Bus and RPOB is revealed preference ordinary bus. Similarly, PB is Premium Bus. The increase from the RP level (present level) was made varied for different travelers based on their current way of travel to generate meaningful and realistic levels for SP qualities. The database shown in Figure 5.1 was taken as input data in NLOGIT software for the generation of the MNL model.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Sl no.	MODE	AET	IVTTKM	FAREKM	H	SPAN	SE1	SE2	SE3	TI1	TI2	TI3	DCOM1	DCOM2	DCOM3	CHOICE
2	1	SPCAR	0	2	1280	15	16	0	1	0	1	0	0	1	0	0	1
3	1	SPOB	0	5	800	20	12	0	0	0	0	0	0	0	0	0	0
4	1	RPOB	5	5.16	129.0323	15	0	0	0	0	0	0	0	0	1	0	1
5	1	RPCar	4	3.23	645.1613	0	0	0	0	0	0	0	0	0	0	0	0
6	2	SPCAR	0	2	2500	15	12	0	1	0	1	0	0	0	0	0	0
7	2	SPOB	0	3	800	15	16	1	0	0	0	0	1	1	0	0	1
8	2	RPOB	5	5.16	129.0323	15	0	0	0	0	0	0	0	0	1	0	1
9	2	RPCar	4	3.23	645.1613	0	0	0	0	0	0	0	0	0	0	0	0
10	3	SPCAR	0	2	1280	15	24	0	1	0	1	0	0	1	0	0	1
11	3	SPOB	0	5	500	30	24	0	0	1	0	1	0	0	0	0	0
12	3	RPOB	5	5.16	129.0323	15	0	0	0	0	0	0	0	0	1	0	1
13	3	RPCar	4	3.23	645.1613	0	0	0	0	0	0	0	0	0	0	0	0
14	4	SPCAR	0	2	1280	15	16	0	1	0	1	0	0	1	0	0	1
15	4	SPOB	0	5	800	30	16	0	1	0	0	0	0	0	0	0	0
16	4	RPOB	5	5.16	129.0323	15	0	0	0	0	0	0	0	0	1	0	1
17	4	RPCar	4	3.23	645.1613	0	0	0	0	0	0	0	0	0	0	0	0

Figure 5.1: Quantitative and Qualitative Attributes Coded in Cardinal Linear Form and Dummy Coded Form

Here, dummy code (0,0,0) for qualitative attributes is considered as the base and the best level. Level I, II, and III are coded in the order (1,0,0), (0,1,0) and (0,0,1) respectively. ρ^2 value between 0.2 and 0.4 suggests a satisfactory level of model fit. (Louviere et al., 2000).

Table 5.2: Coefficient estimates of RP only, SP-Pooled, Joint SP-RP Models for Choice Riders

Attributes	RP only	SP pooled	SP-RP combined
Access-Egress Time	-0.064 (-6.47)	---	-0.069 (-6.75)
Waiting Time	-0.044 (-10.36)	-0.181 (-11.32)	-0.058 (-17.81)
In vehicle Travel Time	-0.810 (-14.46)	-0.263 (-1.65)	-0.481 (-11.21)
Cost	-0.00459 (-22.14)	-0.011 (-18.31)	-0.0046 (-24.35)
Span of Operation	---	0.349 (11.16)	0.12 (10.59)
Security Arrangement I	---	-2.87 (-7.54)	-1.3 (-8.70)
Security Arrangement II	---	-2.76 (-6.96)	-1.1 (-7.79)
Security Arrangement III	---	-2.35 (-6.07)	-0.88 (-6.79)
Traffic Information I	---	-4.41 (-9.43)	-1.58 (-11.21)
Traffic Information II	---	-3.98 (-7.62)	-1.38 (-8.65)
Traffic Information III	---	-2.57 (-5.34)	-0.77 (-5.12)
Crowding I	-1.73 (-16.85)	---	-1.71 (-16.97)
Crowding II	-1.61 (-12.76)	---	-1.48 (-12.27)
Crowding III	-1.07 (-9.17)	---	-0.97 (-8.59)
Constant [Ordinary Bus]	-3.09 (-18.77)	-4.9 (-2.13)	-2.87 (-19.98)
Constant [Premium Bus]	1.26 (5.70)	-5.3 (-8.82)	1.53 (8.49)
Constant [Car]	-0.75 (-8.89)	-3.7 (-9.92)	-0.71 (-8.99)
ρ^2	0.22	0.25	0.44

5.2 MNL Results and Inferences

The RP, SP, and Joint SP-RP model estimation results for choice riders are summarized in the table above. The coefficients obtained are significant at a 90% confidence level and the signs for each attribute are meaningful as well. From Table 5.2 it may be seen that all the attributes other than the span of service are negative which shows the disutility connected with the attributes that increase with the increase in the magnitude of the attribute. However, the positive sign shows a declining tendency in disutility and a rise in the attribute's magnitude. The levels of qualitative qualities that have a negative sign are less useful than the base levels. Adding to that, the ρ^2 of the model is 0.44 which tells that the model is in good fit.

As can be seen, security setting IV, which served as their baseline, is thought to be more useful than security setups I, II, and III (CCTV Surveillance at bus and bus stops). The decision riders believe that security configurations II and III are of more use, followed by security configuration I (no security configuration). Traffic information I (static information at bus stops) was perceived as having lower utility, similar to how traffic information IV (dynamic information at bus stops and web/mobile-based dynamic information) was considered as having better utility. The need of providing up-to-date information for premium services is demonstrated by this. When compared to base level, crowding level I (standing in an overcrowded environment) is thought to have the highest disutility (occupying a seat).

5.3 Python Code Algorithm

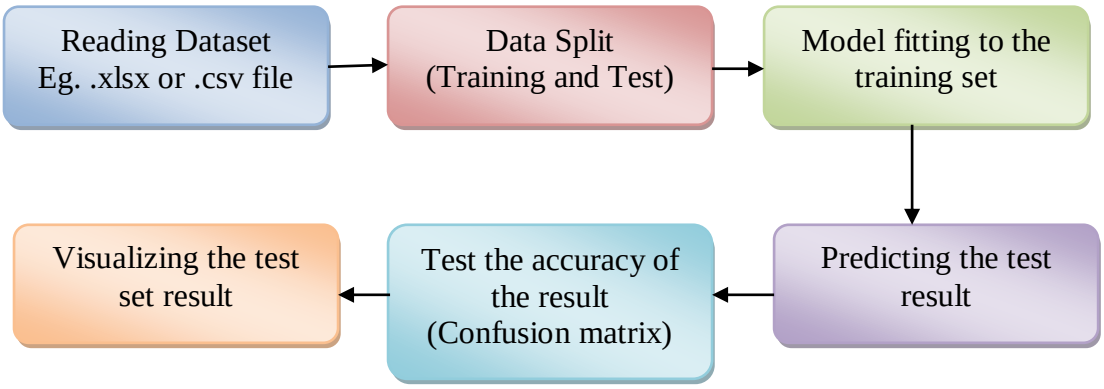

Figure 5.2: Basic Flowchart for Coding the Model

From Figure 5.2 is seen that the dataset file is imported into the program using the 'pandas' library. All the input attributes are defined and stored in a variable (x).

Similarly, the output result is defined as well and stored in another variable (y). Next, the data is split using 'train_test_split' in an 80 to 20 ratio i.e. 80% training and 20% test data. Then the training data is fitted into the model. 20% of the Test data is then predicted. The accuracy of the model is checked using 'accuracy_score' where the predicted result is compared with the actual result. At last, the rest is obtained in the form of a confusion matrix where detailed prediction figures are shown.

5.4 Machine Learning Results

5.4.1 Accuracy and count

The number of observations in the dataset was 1479 out of which 369 were a car, 348 were ordinary buses, 398 were premium bus users, and 364 were respondents chose taxis for conveyance. The python script for the below is shown in Appendix 1.

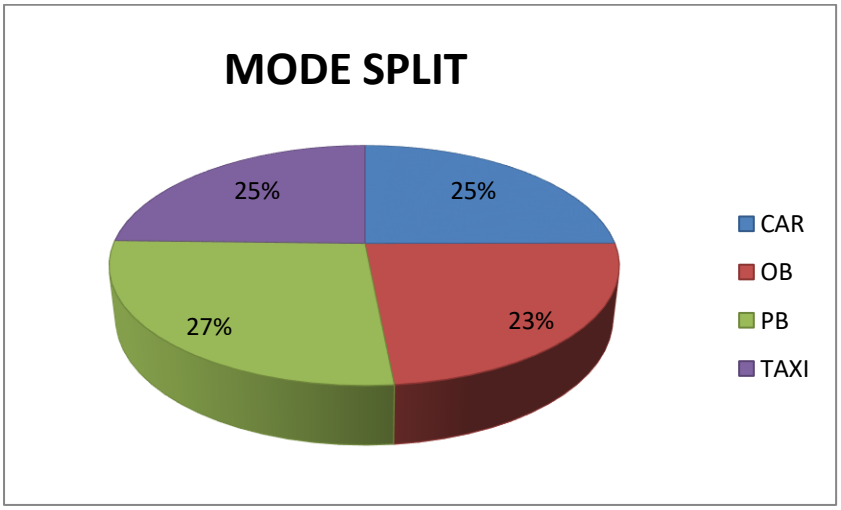

Figure 5.3: Mode Split Classification

- MODEL 1 (Perceptron)

```
Accuracy of Perceptron -  49.42528735632184
              recall  support

CAR             0.99      369
OB              0.00      348
PB              0.91      398
TAXI            0.00      364

accuracy        0.49      1479
```

The accuracy of the Perceptron model is 49.42%. 99% time car was predicted right and the premium bus was predicted correctly 91% of the time.

- MODEL 2 (Support Vector Machine)

```
Accuracy of SVM -  61.66328600405679
              recall  support

CAR            0.71      369
OB             0.43      348
PB             0.58      398
TAXI           0.74      364

accuracy       0.62      1479
```

The accuracy of the Support Vector Machine is 62%. Car is predicted correctly 71% time; OB is rightly predicted 43%, PB 58%, and Taxi 74% of the time.

- MODEL 3 (Decision Tree)

```
Accuracy of DT -  87.35632183908046
              recall  support

CAR            0.95      369
OB             0.77      348
PB             0.82      398
TAXI           0.96      364

accuracy       0.87      1479
```

The accuracy of the model is 87%. Car and taxi have been predicted correctly the highest number of times i.e. 95 and 96 percentage of times.

- MODEL 4 (Random Forest)

```
Accuracy of RF -  87.96484110885734
              recall  support

CAR            0.95      369
OB             0.80      348
PB             0.81      398
TAXI           0.96      364

accuracy       0.88      1479
```

The accuracy of the model is 88%. Car and taxi have been predicted correctly the highest number of times i.e. 95 and 96 percentage of times. The prediction percentage of Ordinary and Premium buses has also improved to 80 and 81% respectively compared to the values from the decision tree.

The comparison of accuracies between different ML techniques is shown in Figure 5.4. The chart shows a linear increase in accuracy while moving from Perceptron to Random Forest.

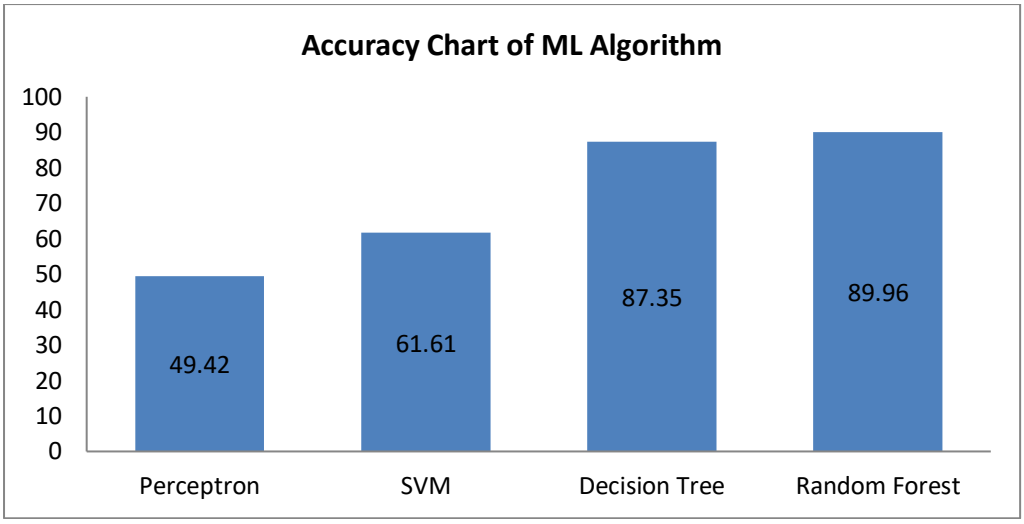

Figure 5.4: Accuracy Comparison Chart

5.4.2 Confusion matrix

A confusion matrix is a matrix table, used to represent the performance of a classification model. It makes visualization and interpretation of the result easy to understand. The values on the x-axis are the predicted values and the ones on the y-axis are actual values. When the predicted value becomes equal to the actual value then the accuracy is said to be more. In general, the diagonal cell elements should be non-zero numbers and all other adjoining cells should be zero. When the model predicted positive and the label was positive is truly positive and when the model predicted negative and the label was negative is a true negative.

- Model 1 (PERCEPTRON)

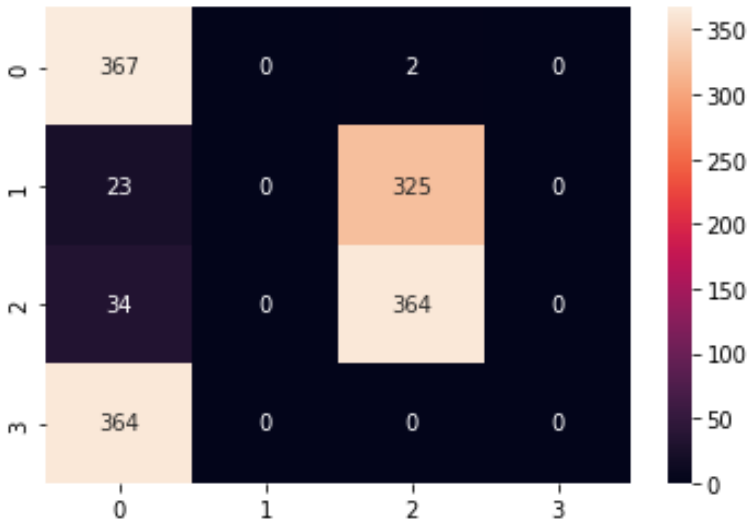

Figure 5.5: Confusion matrix for Perceptron

Figure 5.5 shows the values 0 to 3 that is 0, 1, 2, and 3 are the mode choices **Car, Ordinary Bus, Premium Bus, and Taxi**. In this model, ordinary buses and taxis were predicted zero number of times. Therefore, their accuracy is also zero.

- Model 2 (SVM)

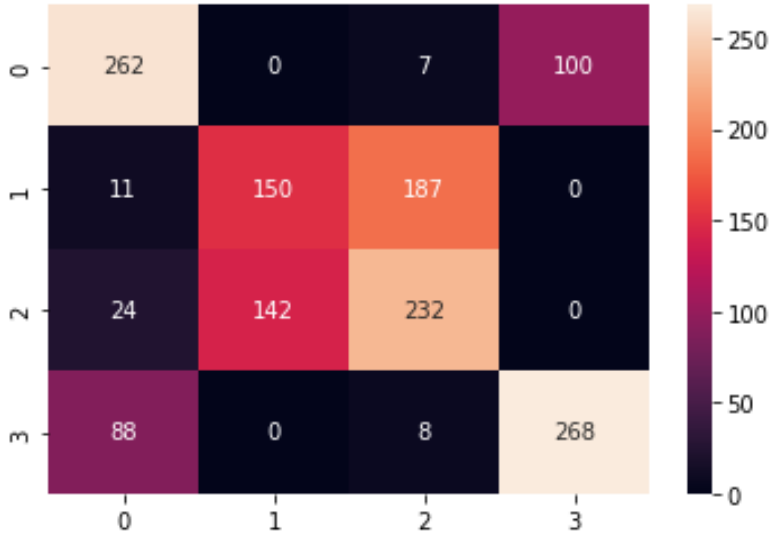

Figure 5.6: Confusion matrix for SVM

Figure 5.6 shows that car predictability is correct 262 times and has been predicted OB, PB, and Taxi 0, 7, and 100 times respectively. Therefore the accuracy for car predictability is reduced to 71%.

- Model 3 (DECISION TREE)

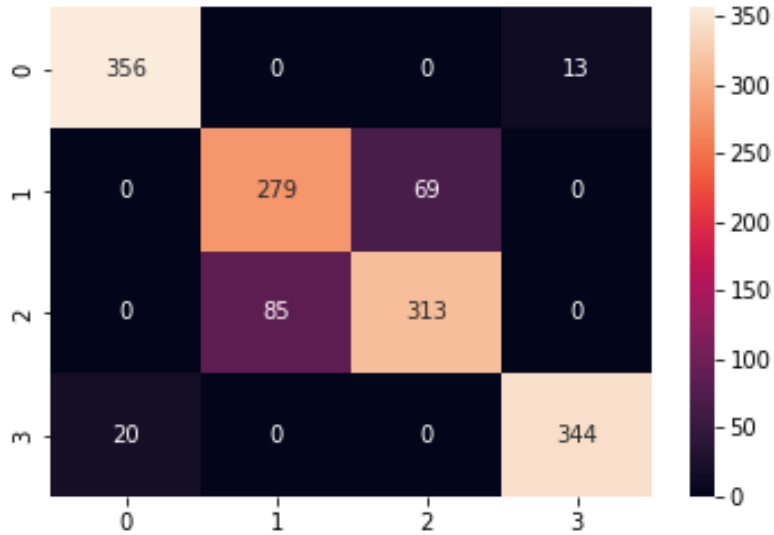

Figure 5.7: Confusion matrix for Decision Tree

In Decision Tree the predictability is more accurate than Perceptron and SVM. For

instance in Figure 5.7 it is shown that car has been predicted correct 356 times and just 13 number of times it has been predicted as Taxi. Therefore car predictability accuracy is 96%.

- Model 4 (RANDOM FOREST)

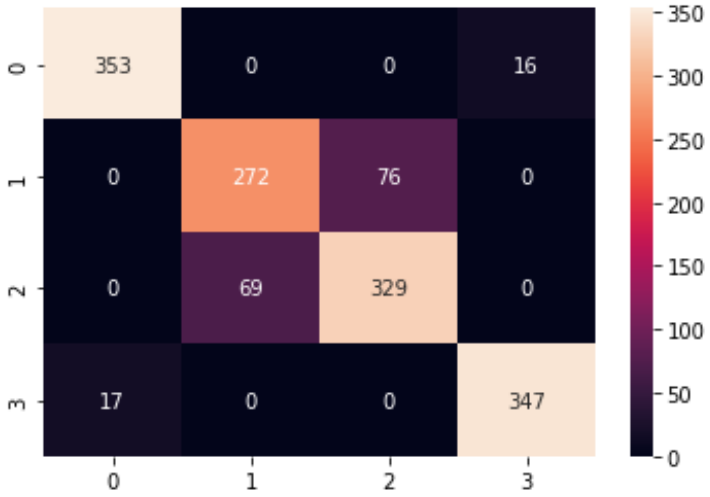

Figure 5.8: Confusion matrix for Random Forest

Figure 5.8 evidently shows that Random Forest predictions are the most accurate of all other models. 353, 272, 329, and 347 are the number of times Car, OB, PB and Taxi have been predicted correctly.

5.4.3 Percentage mode split

The Python script for percentage mode split is seen in Appendix 2 where the entire dataset is assigned for training and test data is provided in the form of a new MS excel sheet. The results below show the mode split when a dataset of 254 responses was used as input test data.

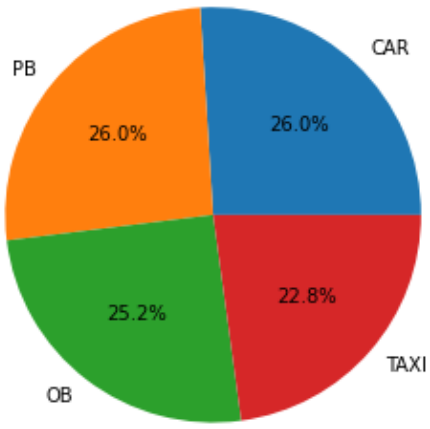

Figure 5.9: Mode Split Using SVM

Following the training dataset and the test data, it is observed in Figure 5.9 that Car and PB have the same percentage of mode share which is 26 percentages. The mode prediction percentages for OB and Taxi are 25.2 % and 22.8 % respectively.

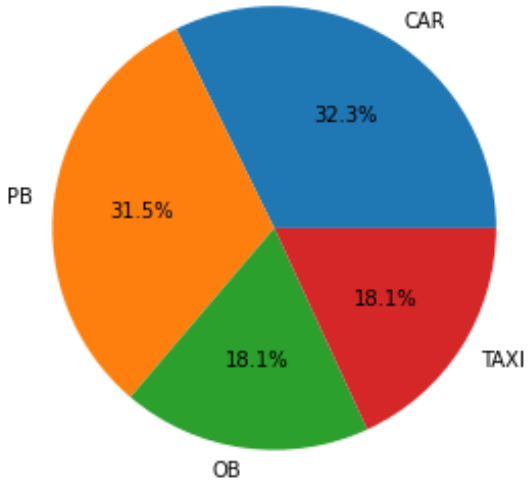

Figure 5.10: Mode Split Using Decision Tree

In case of a decision tree, the Car and Premium bus share is observed to have increased to 32.3 % and 31.5 % respectively. Since the overall accuracy of the Decision Tree model is higher than SVM the mode split result for the decision tree is found to be more accurate as well.

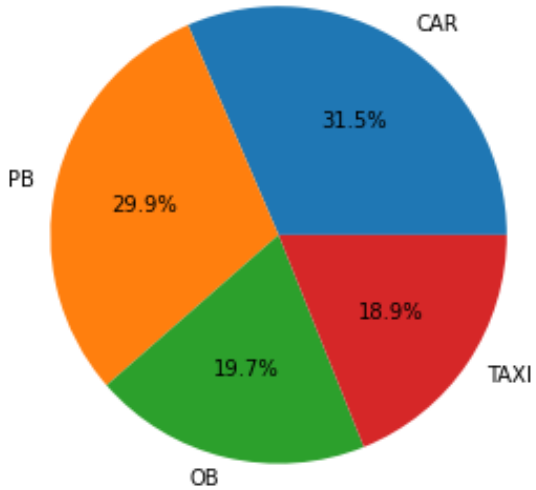

Figure 5.11: Mode Split Using Random Forest

The mode split percentage result using Random forest is somewhat similar to the one obtained using Decision Tree. As the model accuracies are almost similar, the mode split percentages observed are of lesser difference.

5.5 Machine Learning Inferences

The accuracy of the model has increased as the machine learning algorithm moved from basic to advance. For the same dataset, the accuracy percentage changed from 49% to 88% as the algorithm changed from Perceptron to Random Forest.

The accuracy may be improved by either expanding the response quality and count in the database or using a more advanced ML algorithm.

The prediction model gives the overall accuracy of the model while the percentage of people choosing a particular mode is found using the classification model.

Diagonal elements in the Confusion Matrix show the true positive value and other elements show the error count. Therefore to find the accuracy of the model only the values of diagonal elements need to be considered.

CHAPTER 6
CONCLUSION

6.1 General

With urbanization, the usage of private cars and taxi services for commuting has become a common phenomenon in cities. Even with the increase in the number of vehicles on the road the development in the road geometry is stagnant. This has massively affected the smooth flow of traffic and the environment as pollution in recent decades.

To shift the choice riders towards bus services, individual responses are recorded. The user perception survey, utilized to gather SP-RP data from the riders of choice, is precisely used to determine the level of service attribute improvement and fare increment. An examination of the current service reveals glaring flaws in terms of operating time, speed, information, and security, to name just some.

Model calibration and re-estimation of demand model coefficients are required in the case of MNL modelling to use the methodology. Although the conclusions are case-specific, it is hoped that the whole endeavour would assist scholars and practitioners in developing public transportation as a useful instrument for meeting travel demand.

From the study, it is observed that on comparing ML techniques with conventional methods, ML techniques perform much better say in easiness, accuracy, and speed. ML in transport modeling is improving at a rapid rate in terms of accuracy and the present study supports this fact. The future of machine learning in transport modelling is bright, and more study in this field is highly advised.

6.2 Future Scope of Work

The future scope of work is outlined as follows.

- In the present study, the demand model created is limited to the secondary data obtained from Kolkata city. However in future similar work can be conducted in different cities and demand comparison can be identified among cities.
- The modes are limited to Premium Buses, Ordinary Buses, Taxis and Private Cars in this study. Although, there are other bus services available in Kolkata,

such as minibuses and private buses, which are not taken into consideration for the current work. The concept might eventually be applied to the remainder bus services.

- In the present work, a general mode choice model is limited to Multinomial Logit Model for the conventional technique and Machine Learning like Perceptron, SVM, or Decision Trees for advanced techniques. Further, multilayered neural networks (such as ANN etc) can be used for improving the quality of the model.

REFERENCES

1. Allahviranloo, M., & Recker, W. (2013). Daily activity pattern recognition by using support vector machines with multiple classes. *Transportation Research Part B: Methodological*, 58, 16–43.

2. Assi, K. J., Nahiduzzaman, K. M., Ratrout, N. T., & Aldosary, A. S. (2018). Mode choice behavior of high school goers: Evaluating logistic regression and MLP neural networks. *Case Studies on Transport Policy*, 6(2), 225–230.

3. Ben-Akiva, M. E., Lerman, S. R., & Lerman, S. R. (1985). *Discrete choice analysis: theory and application to travel demand* (Vol. 9). MIT press.

4. Bhattacharya, B., Price, R. K., & Solomatine, D. P. (2007). Machine learning approach to modeling sediment transport. *Journal of Hydraulic Engineering*, *133*(4), 440-450.

5. Bhavsar, P., Safro, I., Bouaynaya, N., Polikar, R., & Dera, D. (2017). Machine learning in transportation data analytics. *Data analytics for intelligent transportation systems*, 283-307.

6. Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). New York: springer.

7. Börjesson, M. (2008). Joint RP–SP data in a mixed logit analysis of trip timing decisions. *Transportation Research Part E: Logistics and Transportation Review*, *44*(6), 1025-1038.

8. Brownstone, D., Bunch, D. S., & Train, K. (2000). Joint mixed logit models of stated and revealed preferences for alternative-fuel vehicles. *Transportation Research Part B: Methodological*, *34*(5), 315-338.

9. Cheranchery, M. F., & Maitra, B. (2017). Priority areas of intervention for improving urban bus services: Experience in Kolkata, India. *Transportation Research Record*, *2634*(1), 17-27.

10. Cheranchery, M. F., & Maitra, B. (2019). Improving Ridership and Reducing Subsidy for Premium Bus Service in Kolkata Metro City. *Journal of Transportation Engineering, Part A: Systems*, *145*(7), 04019030.

11. Das, S. S., Maitra, B., & Boltze, M. (2009). Valuing travel attributes of rural

feeder service to bus stop: Comparison of different logit model specifications. *Journal of Transportation Engineering*, *135*(6), 330-337.

12. Dey, T. (2012). Changing profile of state transport undertakings in mass transport services: A case of Kolkata city. *Researchers World*, 3(2), 45.

13. Heilig, M., Mallig, N., Hilgert, T., Kagerbauer, M., & Vortisch, P. (2017). Large-scale application of a combined destination and mode choice model estimated with mixed stated and revealed preference data. *Transportation Research Record*, *2669*(1), 31-40.

14. Hussain, H. D., Mohammed, A. M., Salman, A. D., Rahmat, R. A. B. O. K., & Borhan, M. N. (2017). Analysis of transportation mode choice using a comparison of artificial neural network and multinomial logit models. *ARPN Journal of Engineering and Applied Sciences*, 12(5), 1483–1493.

15. Irfan, M., Khurshid, A. N., Khurshid, M. B., Ali, Y., and Khattak, A. (2018). Policy Implications of Work-Trip Mode Choice Using Econometric Modeling. *Journal ofTransportation Engineering, Part A: Systems*, 144(8), 04018035.

16. Jaiswal, A., & Malhotra, R. (2018). Software reliability prediction using machine learning techniques. *International Journal of System Assurance Engineering and Management*, *9*(1), 230-244.

17. Karlaftis, M. G., & Vlahogianni, E. I. (2011). Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transportation Research Part C: Emerging Technologies*, *19*(3), 387-399.

18. Kelleher, J. D., Mac Namee, B., & D'Arcy, A. (2015). Fundamentals of Machine learning for predictive data analytics: Algorithms, worked examples, and case studies (First). Cambridge: MIT Press.

19. Khan, O. A. (2007). *Modelling passenger mode choice behaviour using computer aided stated preference data* (Doctoral dissertation, Queensland University of Technology).

20. Koushik, A. N., Manoj, M., & Nezamuddin, N. (2020). Machine learning applications in activity-travel behaviour research: a review. *Transport reviews*, *40*(3), 288-311.

21. Lin, H. Z., Lo, H. P., & Chen, X. J. (2009). Lifestyle classifications with and without activity-travel patterns. T*ransportation Research Part A: Policy and Practice*, 43(6), 626–638.

22. Louviere, J. J., Hensher, D. A., & Swait, J. D. (2000). *Stated choice methods: analysis and applications*. Cambridge university press.

23. Maitra, B., Dandapat, S., & Chintakayala, P. (2015). Differences between the perceptions of captive and choice riders toward bus service attributes and the need for segmentation of bus services in urban India. *Journal of Urban Planning and Development*, *141*(2), 04014018.

24. Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised machine learning algorithms: classification and comparison. *International Journal of Computer Trends and Technology (IJCTT)*, *48*(3), 128-138.

25. Peer, S., Knockaert, J., Koster, P., Tseng, Y. Y., & Verhoef, E. T. (2013). Door-to-door travel times in RP departure time choice models: An approximation method using GPS data. *Transportation Research Part B: Methodological*, *58*, 134-150.

26. Phanikumar, C. V., & Maitra, B. (2006). Modeling willingness-to-pay values for rural bus attributes under different trip purposes. *Journal of the Transportation Research Forum* (Vol. 45, No. 1424-2016-117980, pp. 31-44).

27. Reddy, R., & Shyam, G. K. (2018, August). Analysis Through Machine Learning Techniques: A Survey. In *2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT) (pp.*542-546). IEEE.22.

28. Russell, S. J., & Norvig, P. (2009). Artificial intelligence: A modern approach (3rd ed). Upper Saddle River, NJ: Pearson.

29. Sutton, R. S. (1992). Introduction: The challenge of reinforcement learning. In *Reinforcement Learning* (pp. 1-3). Springer, Boston, MA.

30. Tang, L., Xiong, C., & Zhang, L. (2018). Spatial transferability of neural network models in travel demand modeling. *Journal of Computing in Civil Engineering*, 32, 3.

31. Tizghadam, A., Khazaei, H., Moghaddam, M. H., & Hassan, Y. (2019). Machine learning in transportation. *Journal of advanced transportation*, *2019*.

32. Weng, J., Tu, Q., Yuan, R., Lin, P., & Chen, Z. (2018). Modeling mode choice Behaviors for Public transport commuters in Beijing. *Journal of Urban Planning and Development*, 144(3), 1 –9.

33. Xie, C., Lu, J., & Parkany, E. (2003). Work travel mode choice modeling with data mining: decision trees and neural networks. *Transportation Research Record*, *1854*(1), 50-61.

34. Yang, S., Deng, W., Deng, Q., & Fu, P. (2016). The research on prediction models for urban family member trip generation. *KSCE Journal of Civil Engineering*, 20(7), 2910–2919.

35. Zhang, Y., & Xie, Y. (2008). Travel mode choice modeling with support vector machines. *Transportation Research Record: Journal of the Transportation Research Board*, 2076(1), 141 –150.

APPENDIX 1

Base level code for prediction model is shown below. All the attributes such as mode characteristics values are entered by the user.

```
#BASE PERCEPTRON CODE
import pandas as pd
file=pd.read_excel("/content/meenu.xlsx")
x=file[["AET","IVTTKM","FAREKM","H","SPAN","SE1","SE2","SE3","TI1","TI2
","TI3","DCOM1","DCOM2","DCOM3"]]
y=file["MODE"]

#Data Split
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)

#Model Fit
from sklearn.linear_model import Perceptron
model=Perceptron()
model=model.fit(x_train,y_train)

#Attributes as Input
a=input("Access and Egress Time = ")
b=input("In Vehicle Travel Time  = ")
c=input("Fare = ")
d=input("Headway between each service = ")
e=input("Span of travel = ")
f=input("Security 1 = ")
g=input("SE2 = ")
h=input("SE3 = ")
i=input("Travel Information 1 = ")
j=input("TI2 = ")
k=input("TI3 = ")
l=input("Degree of Comfort 1 = ")
m=input("DCOM2 = ")
```

```
n=input("DCOM3 = ")

#Model Predict
result=model.predict([[a,b,c,d,e,f,g,h,i,j,k,l,m,n]])
result=model.predict(x_test)
print(result)

#Accuracy of the model
from sklearn.metrics import accuracy_score
score = accuracy_score(y_test,result)
print(score)
```

APPENDIX 2

Code for predicting the mode choice and calculating the accuracy of the model.

```
#1.Load the DATASET (PANDAS)
import pandas as pd

file=pd.read_excel("/content/trial1.xlsx")

x=file[["AET","IVTTKM","FAREKM","H","SPAN","SE1","SE2","TI1","TI2","TI3"
,"DCOM1","DCOM2","DCOM3"]]
y=file[["MODE"]]

print(x)
print(y)

#DATA SPLIT
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(x,y,train_size=0.8)

#MODEL 1 PERCEPTRON
from sklearn.linear_model import Perceptron
model1=Perceptron()
model1=model1.fit(xtrain,ytrain)
op1=model1.predict(xtest)
print(op1)
from sklearn.metrics import accuracy_score
score1=accuracy_score(ytest,op1)
print("Accuracy of Perceptron - ",score1*100)

from sklearn.metrics import classification_report
print(classification_report(ytest, op1))
confusion_matrix(ytest, op1)
sns.heatmap(confusion_matrix(ytest,op1), annot=True, fmt='0.0f')
results = confusion_matrix(ytest,op1)
```

```
print(results)

#MODEL 2 SVM
from sklearn.svm import SVC
import seaborn as sns

model2=SVC()
model2=model2.fit(xtrain,ytrain)
op2=model2.predict(xtest)

from sklearn.metrics import accuracy_score
score2=accuracy_score(ytest,op2)
print("Accuracy of SVM - ",score2*100)

from sklearn.metrics import classification_report
print(classification_report(ytest, op2))

from sklearn.metrics import confusion_matrix
confusion_matrix(ytest, op2)
sns.heatmap(confusion_matrix(ytest,op2), annot=True, fmt='0.0f')
results = confusion_matrix(ytest,op2)
print(results)


#INDIVIDUAL DATA PREDICTION
out=model2.predict([[15,37,25,20,0,1,0,1,1,1,0,1,1]])
print(out)

#MODEL 3 DECISION TREE
from sklearn.tree import DecisionTreeClassifier
model3=DecisionTreeClassifier()
model3=model3.fit(xtrain,ytrain)
op3=model3.predict(xtest)
from sklearn.metrics import accuracy_score
```

```
score3=accuracy_score(ytest,op3)
print("Accuracy of DT - ",score3*100)

from sklearn.metrics import classification_report
print(classification_report(ytest, op3))
confusion_matrix(ytest, op3)
sns.heatmap(confusion_matrix(ytest,op3), annot=True, fmt='0.0f')
results = confusion_matrix(ytest,op3)
print(results)

#MODEL 4 RANDOM FOREST CLASSIFIER
from sklearn.ensemble import RandomForestClassifier
model4=RandomForestClassifier()
model4=model4.fit(xtrain,ytrain)
op4=model4.predict(xtest)

from sklearn.metrics import accuracy_score
score4=accuracy_score(ytest,op4)
print("Accuracy of RF -",score4*100)

from sklearn.metrics import classification_report
print(classification_report(ytest, op4))
confusion_matrix(ytest, op4)
sns.heatmap(confusion_matrix(ytest,op4), annot=True, fmt='0.0f')
results = confusion_matrix(ytest,op4)
print(results)
```

APPENDIX 3

Code for mode classification from an input data set.

```
import pandas as pd
file=pd.read_excel("/content/trial1.xlsx")
file2=pd.read_excel("/content/trial2- test.xlsx")
#convert all strings to numerical data (label encoder)

df=pd.DataFrame(file)
df2=pd.DataFrame(file2)


x=df[["AET","IVTTKM","FAREKM","H","SPAN","SE1","SE2","TI1","TI2","TI3","
DCOM1","DCOM2","DCOM3"]]
y=df[["MODE"]]
x2=df2[["AET","IVTTKM","FAREKM","H","SPAN","SE1","SE2","TI1","TI2","TI3
","DCOM1","DCOM2","DCOM3"]]

#0-CAR
#1-PB
#2-OB
#3-TAXI

#MODEL 2 SVM
from sklearn.svm import SVC
import seaborn as sns

model2=SVC()
model2=model2.fit(xtrain,ytrain)
op2=model2.predict(x2)
print(op2)

op2=pd.DataFrame(op2)
from sklearn.preprocessing import LabelEncoder
```

```
le=LabelEncoder()
op2=op2.apply(le.fit_transform)

print(op2.value_counts())

from matplotlib import pyplot as plt
label=['CAR','PB','OB','TAXI']
plt.figure(figsize=(5,5))
plt.pie( (op2.value_counts()) , labels = label , autopct = '%1.1f%%', explode=(0,0,0,0)
)
plt.show()


#MODEL 3 DECISION TREE
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(x,y)

from sklearn.tree import DecisionTreeClassifier
model3=DecisionTreeClassifier()
model3=model3.fit(xtrain,ytrain)
op3=model3.predict(x2)
print(op3)

#import matplotlib.pyplot as plt

op3=pd.DataFrame(op3)
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
op3=op3.apply(le.fit_transform)

print(op3.value_counts())

from matplotlib import pyplot as plt
label=['CAR','PB','OB','TAXI']
```

```
plt.figure(figsize=(5,5))
plt.pie( (op3.value_counts()) , labels = label , autopct = '%1.1f%%', explode=(0,0,0,0)
)
plt.show()

#MODEL 4 RANDOM FOREST CLASSIFIER
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(x,y)

from sklearn.ensemble import RandomForestClassifier
model4=RandomForestClassifier()
model4=model4.fit(xtrain,ytrain)
op4=model4.predict(x2)

op4=pd.DataFrame(op4)
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
op4=op4.apply(le.fit_transform)

print(op4.value_counts())

from matplotlib import pyplot as plt
label=['CAR','PB','OB','TAXI']
plt.figure(figsize=(5,5))
plt.pie( (op4.value_counts()) , labels = label , autopct = '%1.1f%%', explode=(0,0,0,0)
)
plt.show()
```