CUSTOMER LIFETIME VALUE PREDICTION

Project Report

Submitted for the Partial Fulfillment of the Requirements
for the Award of the Degree of

M.Sc. Artificial Intelligence

By

GOVIND KANNAN-200011024085

Department of Computer Science

SCHOOL OF TECHNOLOGY & APPLIED SCIENCES
CENTRE FOR PROFESSIONAL AND ADVANCED STUDIES
KOTTAYAM, KERALA
September-2022

SCHOOL OF TECHNOLOGY & APPLIED SCIENCES

CENTRE FOR PROFESSIONAL AND ADVANCED STUDIES

KOTTAYAM, KERALA

CERTIFICATE

Certified that this is a Bonafide Record of Project Report done by **Mr. GOVIND KANNAN** Reg.No **200011024085** for the partial fulfillment of the requirement for the award of the degree of M.Sc. Artificial Intelligence of Mahatma Gandhi University, Kottayam during the period 2020- 2022.

Place: Kottayam
Date:

Head of the Department   Project Guide

Submitted for the External Examination held on …...................................

Examiner1   Examiner2
(Name & Signature)   (Name & Signature)

SCHOOL OF TECHNOLOGY & APPLIED SCIENCES

CENTRE FOR PROFESSIONAL AND ADVANCED STUDIES

KOTTAYAM, KERALA

DECLARATION

I, **GOVIND KANNAN**, hereby declare that the project work entitled "CUSTOMER LIFETIME VALUE PREDICTION" under the guidance of **Dr. Saritha k** for the partial fulfillment of the course M.Sc. Cyber Forensic. This work has not been submitted for similar purpose anywhere else except to the School of Technology and Applied Sciences (STAS), Pullarikkunu, Kottayam.

I understand that detection of any such copying is liable to be punished in any way the school deems fit.

GOVIND KANNAN

Place:Kottayam

Date:

ACKNOWLEDGEMENT

I would express my gratitude to the Management of School of Technology& Applied sciences for providing me with all the required facilities without which the report would not have been possible. I express my heartfelt gratitude to my Principal, **Dr. BINDU M.S** for her warm support with regard to my work.

I would express my deep sense of gratitude to **Mrs. JISHA MARY GEORGE**, HOD-Computer science, School of Technology and Applied Sciences, Kottayam for her valuable help and guidance.

I would express my deep sense of gratitude and thankfulness to our project guide **Dr. SARITHA K** for his valuable help and guidance. I also extend my sincere thanks to all other members of the facility of the Department of Computer Sciences for their assistance and encouragement.

Last but not the least I would like to thank my friends for their cooperation and encouragement.

GOVIND KANNAN
200011024085

Abstract

Being able to predict the loyal customer in future is a big concern nowadays. A more customer-centric marketing strategy is enabled by accurate estimates of consumers' future lifetime value based on their profiles and historical purchasing behaviour. One difficulty with lifetime value modelling is that some customers never return, resulting in an intense distribution of lifetime value. We use supervised regression to predict the lifetime value of new consumers.

The purpose of this project is to look into regression models to identify which one is the most accurate at predicting customer lifetime value performance based on previous transaction data. This project compares the performance of seven supervised machine learning techniques: linear regression, lasso regression, ridge regression, random forest regression and decision tree. The study is performed on the transaction data from Kaggle public repository, the transaction data of Acquire valued shoppers challenge dataset. R-squared, RMSE, MAE, and CV(RMSE) were the metrics used to assess the models' performance. Linear regression and ridge regression were shown to be more accurate across all datasets in the investigation with almost 96% accuracy and 21% CVRMSE. Random forest and Lasso regression can be recommended as well with approximately 94% accuracy and 26% CVRMSE.

INDEX

1. Introduction

1.1 Introduction

We are living in a world of competition, so that for the success of a company or a business firm needs to satisfy the customer based on their purchasing behavior. In this regard, the management of the business firm or a company make a good relationship with the customer. For that, a company needs to collect a large database of a customer purchasing behavior and predict their lifetime. This is customer lifetime value prediction.

The Customer Lifetime Value (CLV) is the total amount of money spent by a customer or client at a store or company firm over the course of their lifetime. It is very important to figure out these customers and from that result we can create a customer profile regarding the amount of money he/she spend to the business and can invest money in the business to welcome new customers and retain the existing customers.

Using CLV or CLV predictors (such as previous purchase history) to allocate marketing resources presume that a customer's future value can be reliably forecasted. This hypothesis is rarely debated, and there is little actual data to back it up. The precision with which a customer's future value may be anticipated is on a scale. At one extreme, future behavior may be precisely anticipated based on the customer's previous behavior and the firm's marketing activity. The opposite extreme is where customers' future conduct is unaffected by their previous behavior or the firm's marketing efforts.

1.2 Regression

In Machine Learning, we use various kinds of algorithms to allow machines to learn the relationships within the data provided and make predictions based on patterns or rules identified from the dataset. So, regression is a machine learning technique where the model predicts the output as a continuous numerical value.Regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. It’s used as a method for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes.

Solving regression problems is one of the most common applications for machine learning models, especially in supervised machine learning. Algorithms are trained to understand the relationship between independent variables and an outcome or dependent variable. The model can then be leveraged to predict the outcome of new and unseen input data, or to fill a gap in missing data.

Regression analysis is an integral part of any forecasting or predictive model, so is a common method found in machine learning powered predictive analytics. Alongside classification, regression is a common use for supervised machine learning models. This approach to training models required labelled input and output training data. Machine learning regression models need to understand the relationship between features and outcome variables, so accurately labelled training data is vital.

Regression is a key element of predictive modelling, so can be found within many different applications of machine learning. Whether powering financial forecasting or predicting healthcare trends, regression analysis can bring organisations key insight for decision-making. It's already used in different sectors to forecast house prices, stock or share prices, or map salary changes.

This guide explores regression in machine learning, including what it is, how it's used, and the different types of regression in machine learning.

What is machine learning regression?

Regression is a method for understanding the relationship between independent variables or features and a dependent variable or outcome. Outcomes can then be predicted once the relationship between independent and dependent variables has been estimated. Regression is a field of study in statistics which forms a key part of forecast models in machine learning. It's used as an approach to predict continuous outcomes in predictive modelling, so has utility in forecasting and predicting outcomes from data. Machine learning regression generally involves plotting a line of best fit through the data points. The distance between each point and the line is minimised to achieve the best fit line.

Alongside classification, regression is one of the main applications of the supervised type of machine learning. Classification is the categorisation of objects based on learned features, whereas regression is the forecasting of continuous outcomes. Both are predictive modelling problems. Supervised machine learning is integral as an approach in both cases, because classification and regression models rely on labelled input and output training data. The features and output of the training data must be labelled so the model can understand the relationship.

Regression analysis is used to understand the relationship between different independent variables and a dependent variable or outcome. Models that are trained to forecast or predict trends and outcomes will be trained using regression techniques. These models will learn the relationship between input and output data from labelled training data. It can then forecast future trends or predict outcomes from unseen input data, or be used to understand gaps in historic data.

As with all supervised machine learning, special care should be taken to ensure the labelled training data is representative of the overall population. If the training data is not representative, the predictive model will be overfit to data that doesn't represent new and unseen data. This will result in inaccurate predictions once the model is deployed. Because regression analysis involves the relationships of features and outcomes, care should be taken to include the right selection of features too.

What are regression models used for?

Machine learning regression models are mainly used in predictive analytics to forecast trends and predict outcomes. Regression models will be trained to understand the relationship between different independent variables and an outcome. The model can therefore understand the many different factors which may lead to a desired outcome. The resulting models can be used in a range of ways and in a variety of settings. Outcomes can be predicted from new and unseen data, market fluctuations can be predicted and accounted for, and campaigns can be tested by tweaking different independent variables.

In practice, a model will be trained on labelled data to understand the relationship between data features and the dependent variable. By estimating this relationship, the model can predict the outcome of new and unseen data. This could be used to predict missing historic data, and estimate future outcomes too. In a sales environment, an organisation could use regression machine learning to predict the next month's sales from a number of factors. In a medical environment, an organisation could forecast health trends in the general population over a period of time.

Supervised machine learning models are generally used for either classification or regression problems. Classification is when a model is trained to categorise an object based on its features. This could include facial recognition software, or to identify a spam email in a firewall. A model will be trained on labelled input and output data to understand the specific features which classify a labelled object. On the other hand, a regression problem is when a model is used to predict continuous outcomes or values. This could be a model that forecasts salary changes, house prices, or retail sales. The model is trained on labelled input and output data to understand the strength of relationships between data features and output.

Regression is used to identify patterns and relationships within a dataset, which can then be applied to new and unseen data. This makes regression a key element of machine learning in finance, and is often leveraged to help forecast portfolio performance or stock costs and trends. Models can be trained to understand the relationship between a variety of diverse features and a desired outcome. In most cases, machine learning regression provides organisations with insight into particular outcomes. But because this approach can influence an organisation's decision-making process, the explainability of machine learning is an important consideration.

Common use for machine learning regression models include:

- Forecasting continuous outcomes like house prices, stock prices, or sales.
- Predicting the success of future retail sales or marketing campaigns to ensure resources are used effectively.
- Predicting customer or user trends, such as on streaming services or ecommerce websites.
- Analysing datasets to establish the relationships between variables and an output.
- Predicting interest rates or stock prices from a variety of factors.
- Creating time series visualisations.

1.2.1 Linear regression

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Different regression models differ based on – the kind of relationship between dependent and independent variables they are considering, and the number of independent variables getting used.Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression.

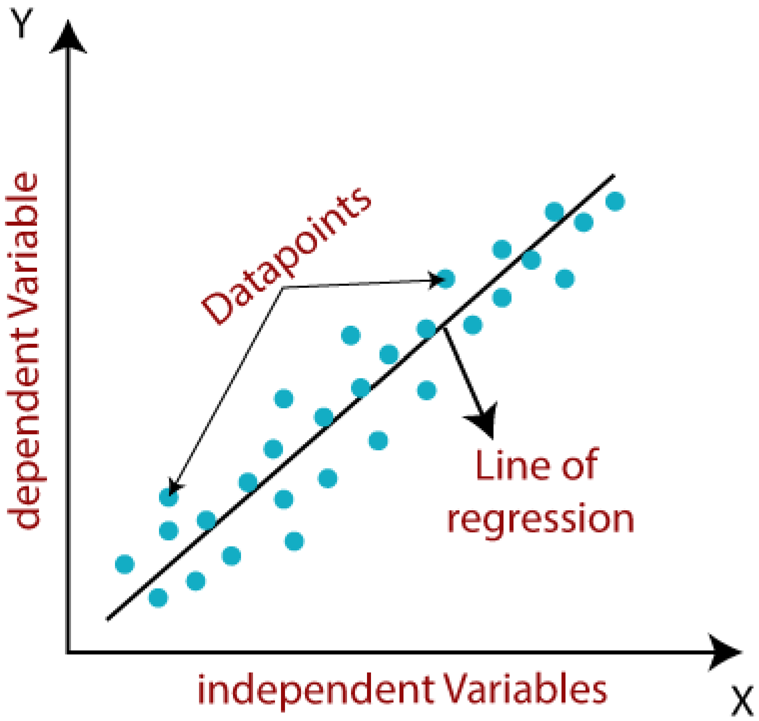


Fig. 1.1 graph for linea rregression

In the figure above, X (input) is the work experience and Y (output) is the salary of a person. The regression line is the best fit line for our model.

Hypothesis function for Linear Regression :

$$y = \theta_1 + \theta_2.x$$

While training the model we are given :

x: input training data (univariate – one input variable(parameter))

y: labels to data (supervised learning)

When training the model – it fits the best line to predict the value of y for a given value of x. The model gets the best regression fit line by finding the best θ1 and θ2 values.

θ1: intercept

θ2: coefficient of x

Once we find the best θ1 and θ2 values, we get the best fit line. So when we are finally using our model for prediction, it will predict the value of y for the input value of x.

Types of Linear Regression

Linear regression can be further divided into two types of the algorithm:

- Simple Linear Regression:
  If a single independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Simple Linear Regression.
- Multiple Linear regression:
  If more than one independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression.

Linear Regression Line

A linear line showing the relationship between the dependent and independent variables is called a regression line. A regression line can show two types of relationship:

- Positive Linear Relationship: If the dependent variable increases on the Y-axis and independent variable increases on X-axis, then such a relationship is termed as a Positive linear relationship.

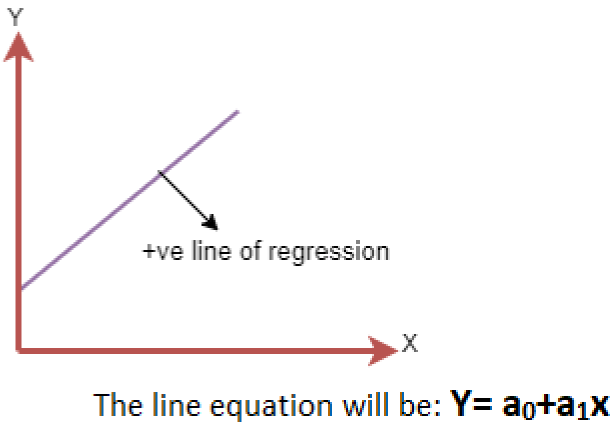


Fig. 1.2 Graph for positive linear relationship

- Negative Linear Relationship: If the dependent variable decreases on the Y-axis and independent variable increases on the X-axis, then such a relationship is called a negative linear relationship.

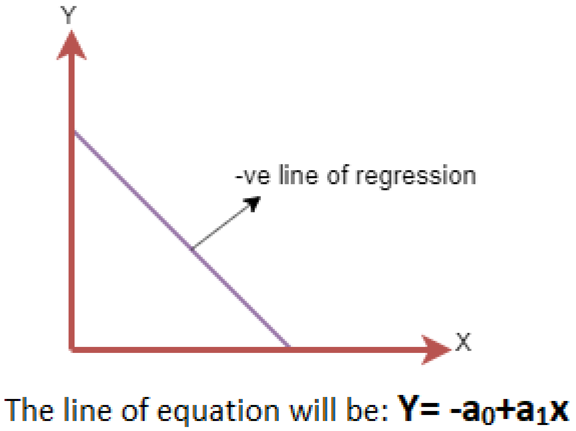


Fig. 1.3 Graph for Negative linear relationship

Finding the best fit line:

When working with linear regression, our main goal is to find the best fit line that means the error between predicted values and actual values should be minimized. The best fit line will have the least error.

The different values for weights or the coefficient of lines (a_0, a_1) gives a different line of regression, so we need to calculate the best values for a_0 and a_1 to find the best fit line, so to calculate this we use cost function.

Cost function-

- The different values for weights or coefficient of lines (a_0, a_1) gives the different line of regression, and the cost function is used to estimate the values of the coefficient for the best fit line.
- Cost function optimizes the regression coefficients or weights. It measures how a linear regression model is performing.
- We can use the cost function to find the accuracy of the mapping function, which maps the input variable to the output variable. This mapping function is also known as Hypothesis function.

For Linear Regression, we use the Mean Squared Error (MSE) cost function, which is the average of squared error occurred between the predicted values and actual values. It can be written as:

$$\text{MSE}= 1\frac{1}{N}\sum_{i=1}^{n}(y_i \;-(a_1x_i + a_0))^2$$

Where,

N=Total number of observation

Yi = Actual value

($a1x_i$+a_0)= Predicted value.

Residuals: The distance between the actual value and predicted values is called residual. If the observed points are far from the regression line, then the residual will be high, and so cost function will high. If the scatter points are close to the regression line, then the residual will be small and hence the cost function.

Gradient Descent:

- Gradient descent is used to minimize the MSE by calculating the gradient of the cost function.
- A regression model uses gradient descent to update the coefficients of the line by reducing the cost function.
- It is done by a random selection of values of coefficient and then iteratively update the values to reach the minimum cost function.

Model Performance:

The Goodness of fit determines how the line of regression fits the set of observations. The process of finding the best model out of various models is called optimization. It can be achieved by below method:

1. R-squared method:

- R-squared is a statistical method that determines the goodness of fit.
- It measures the strength of the relationship between the dependent and independent variables on a scale of 0-100%.
- The high value of R-square determines the less difference between the predicted values and actual values and hence represents a good model.
- It is also called a coefficient of determination, or coefficient of multiple determination for multiple regression.
- It can be calculated from the below formula:

$$\text{R-squared}=\frac{\text{Explained variation}}{\text{Total Variation}}$$

Assumptions of Linear Regression

Below are some important assumptions of Linear Regression. These are some formal checks while building a Linear Regression model, which ensures to get the best possible result from the given dataset.

- Linear relationship between the features and target:
  Linear regression assumes the linear relationship between the dependent and independent variables.
- Small or no multicollinearity between the features:
  Multicollinearity means high-correlation between the independent variables. Due to multicollinearity, it may difficult to find the true relationship between the predictors and target variables. Or we can say, it is difficult to determine which predictor variable is affecting the target variable and which is not. So, the model assumes either little or no multicollinearity between the features or independent variables.
- Homoscedasticity Assumption:
  Homoscedasticity is a situation when the error term is the same for all the values of independent variables. With homoscedasticity, there should be no clear pattern distribution of data in the scatter plot.
- Normal distribution of error terms:
  Linear regression assumes that the error term should follow the normal distribution pattern. If error terms are not normally distributed, then confidence intervals will become either too wide or too narrow, which may cause difficulties in finding coefficients.
  It can be checked using the q-q plot. If the plot shows a straight line without any deviation, which means the error is normally distributed.
- No autocorrelations:
  The linear regression model assumes no autocorrelation in error terms. If there will be any correlation in the error term, then it will drastically reduce the accuracy of the model. Autocorrelation usually occurs if there is a dependency between residual errors.

1.2.2 Lasso

Lasso regression is a classification algorithm that uses shrinkage in simple and sparse models(i.e model with fewer parameters). In Shrinkage, data values are shrunk towards a central point like the mean.

Lasso regression is a regularized regression algorithm that performs L1 regularization which adds penalty equal to the absolute value of the magnitude of coefficients.

“LASSO” stands for Least Absolute Shrinkage and Selection Operator. Lasso regression is good for models showing high levels of multicollinearity or when you want to automate certain parts of model selection i.e variable selection or parameter elimination. Lasso regression solutions are quadratic programming problems that can best solve with software like RStudio, Matlab, etc. It has the ability to select predictors.

$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

The algorithm minimizes the sum of squares with constraint. Some Beta are shrunk to zero that results in a regression model. A tuning parameter lambda controls the strength of the L1 regularization penalty. lambda is basically the amount of shrinkage:

- When lambda = 0, no parameters are eliminated.
- As lambda increases, more and more coefficients are set to zero and eliminated & bias increases.
- When lambda = infinity, all coefficients are eliminated.
- As lambda decreases, variance increases.

Also, If an intercept is included in the model, it is left unchanged.

There are two main regularization techniques, namely Ridge Regression and Lasso Regression.f a regression model uses the L1 Regularization technique, then it is called Lasso Regression. If it used the L2 regularization technique, it’s called Ridge Regression.
Limitation of Lasso Regression:

- Lasso sometimes struggles with some types of data. If the number of predictors (p) is greater than the number of observations (n), Lasso will pick at most n predictors as non-zero, even if all predictors are relevant (or may be used in the test set).

- If there are two or more highly collinear variables then LASSO regression select one of them randomly which is not good for the interpretation of data.

1.2.3 Ridge

Ridge regression is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values. The cost function for ridge regression:
Min(||Y – X(theta)||^2 + λ||theta||^2)

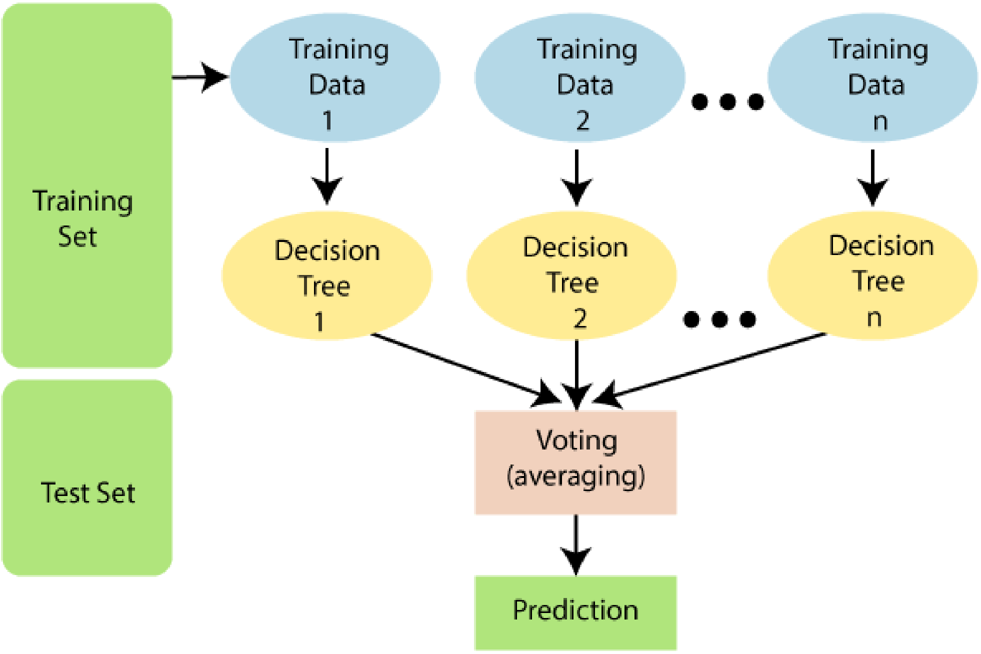
Lambda is the penalty term. λ given here is denoted by an alpha parameter in the ridge function. So, by changing the values of alpha, we are controlling the penalty term. The higher the values of alpha, the bigger is the penalty and therefore the magnitude of coefficients is reduced.

1.2.4 Random Forest

Random Forest Classifier: is one of the classification trees algorithms, the main goal of this algorithm is to enhance trees classifiers based on the concept of the forest.

To implement this algorithm the number of trees within the forest should be figured because each individual tree within a forest predicts the expected output. Then next the voting technique is used to select the expected output that has the largest votes number.

Random Forest model is imported from sklearn and the model is instantiated, and an then we use the fit method on the model to train the data. After training the model, we predict the data using predict method on the random forest model.



1.4 Random forest flow diagram

1.2.5 Decision Tree

Decision Tree is one of the most commonly used, practical approaches for supervised learning. It can be used to solve both Regression and Classification tasks with the latter being put more into practical application.

It is a tree-structured classifier with three types of nodes. The Root Node is the initial node which represents the entire sample and may get split further into further nodes. The Interior Nodes represent the features of a data set and the branches represent the decision rules. Finally, the Leaf Nodes represent the outcome. This algorithm is very useful for solving decision-related problems.

With a particular data point, it is run completely through the entirely tree by answering True/False questions till it reaches the leaf node. The final prediction is the average of the value of the dependent variable in that particular leaf node. Through multiple iterations, the Tree is able to predict a proper value for the data point.

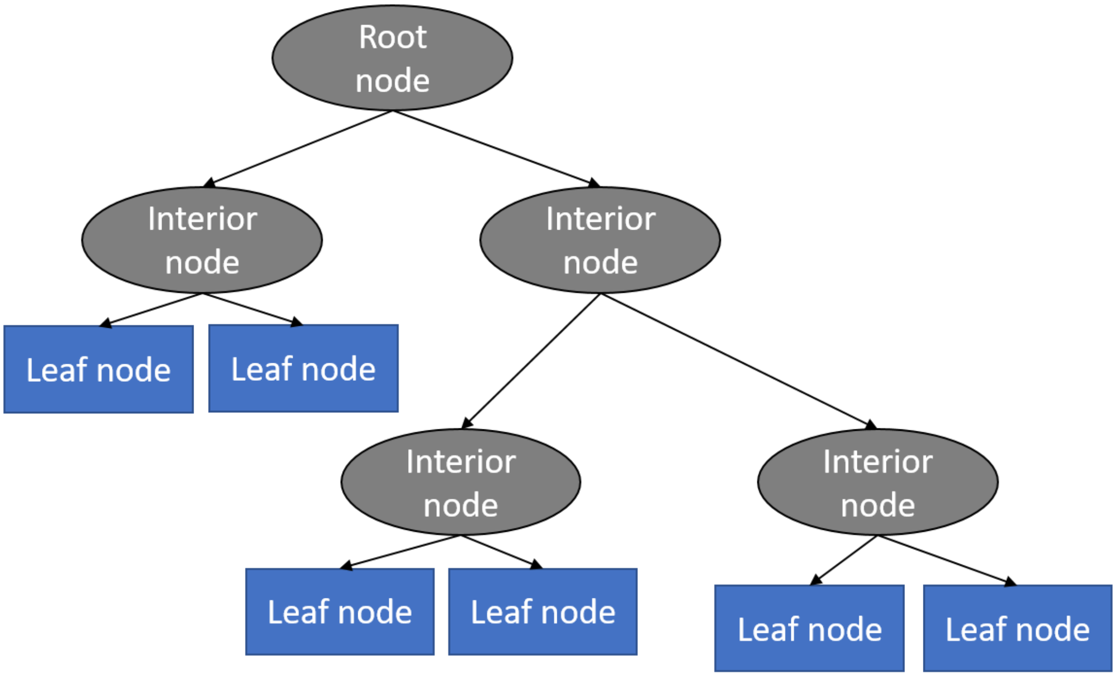


Fig. 1.5 Example for decision tree

1.3 Background

Customer lifetime value should be a key consideration when planning and budgeting for a variety of marketing initiatives, including customer acquisition campaigns. The business world's marketing focus changed from product-centric to customer-centric techniques. As a result, customer-company interactions have changed in such a way that numerous new marketing opportunities have emerged (Shirazi & Mohammadi, 2019).

It is necessary to establish a precise explanation of the Life Time Value (LTV) concept in order to properly understand its rationale and use. Lifetime Value is defined as the customer's present value in terms of generated income and incurred expenditures from a business standpoint. for the purpose of obtaining and maintaining a customer. The simplest basic equation is derived from the sum of the discounted free cash flows created by the person throughout:

$$LT\ V = \sum\ t=1\ ,T\ (Revt - Ct)/\ (1 + d)\ t\ ) - UAC$$

Where t is the period of cash flow; T is the total number of periods of projected life for the customer under consideration; Revt is the revenue from the customer in period t; Ct is the marginal cost of servicing the user to generate the revenue Revt in period t; d is the discount rate and UAC is the User Acquisition Cost.

In the case of credit card customers, estimating a credit card customer's CLV can assist a card issuer bank in making the clients with high CLV scores may be given priority during acquisition, and the acquisition channel can be determined appropriately, with a more expensive route for high-worth individuals and a less expensive channel for prospects with low CLV scores. Similarly, calculating CLV can aid in decision-making at the retention stage. The company may want to keep clients with high CLV ratings and might set the cost of retention efforts appropriately. CLV has been proposed as a statistic for identifying consumers, planning marketing campaigns, and making informed decisions in a systematic framework by researchers. Customers chosen based on the CLV statistic are more profitable than customers chosen based on other commonly used CRM indicators including previous-period customer income, prior customer value, and customer lifetime length (Aeron et al., 2008).

Customer Relationship Management (CRM) used in this study considers marketing strategy as well as product attractiveness. Marketing strategy, which includes advertising, promotion, membership programs, and other tactics, is now considered a necessary complement to the selling of freshly introduced or current items. Furthermore, academics and industry insiders have researched CRM extensively. Many people assume that the major goal of CRM is to find, satisfy, and keep the most profitable customers in try to minimise expenses and boost revenue. Customer profitability should be determined by a series of transactions or future transactions rather than a single transaction with the company. As a result, CLV has emerged as a critical metric in relationship marketing and a core CRM concept (Sun et al., 2021).

Companies are seeking for new ways to provide value and optimize their market offers in order to attract customers and make income. Many businesses use CLV on a daily basis to monitor and regulate marketing efforts as well as assess business success. The total CLV is defined as the discounted value of future profits generated by customers for the firm, or, in other words, the worth of a customer as the predicted sum of discounted future earnings, where each period provides a profit margin. Furthermore, CLV represents the current worth of future advantages as well as the equity of customer advertising strategies. CLV also plays an important part in the relationship's marketing. In relationship marketing, client relationships may be thought of as capital assets that must be properly managed (Chen, 2018).

1.3.1 Motivation

A company may be depending on chance purchases when allocating marketing spending based on past data. A certain amount of unpredictability will always exist in a customer's purchases. Is it true that clients who get special care are better consumers? Or are they consumers who were "better" over a recent period and will "drop back" to their real, non-best-customer conduct in the future? This paper gives a clear analysis of how well future consumer behavior may be predicted.

Calculating CLV has lots of applications and different authors develops different models such as performance measurement, targeting customers, marketing resources allocations, product offering, pricing, and customer segmentations. In this paper, I would like to predict the customer lifetime value of Acquire Valued Shoppers Challenge dataset and have to predict which shoppers will become repeat customer using machine learning and neural network. In recent years, the concepts of CLV have garnered attention in marketing literature. CLV has shown to be a useful measure for assessing business efficiency and business outcomes (de Caigny et al., 2018). The main aim of this paper is to conduct an analytical comparison of chosen CLV models and utilize statistical measures to assess their prediction capacity and quality in an e-commerce market (Jasek et al., 2019).

Customer brands often give discounts to their customers to attract their products. The loyal customers are those who return after their last purchase. It is possible to predict the valuable customers from their purchase history. The Acquire Valued Shoppers Challenge aims to identify which customers are more likely to make a repeat purchase. Nearly 350 million rows of entirely anonymized transactional data from over 300,000 shoppers are available in this collection. Figure shows the basic prediction procedure.

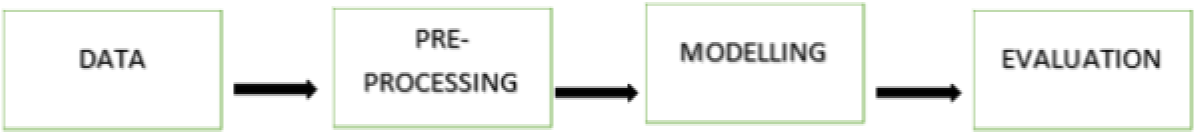


Fig. 1.6 Basic prediction procedure

The figure 1.1 above is a flow chart that the represents the whole process that carried out in this CLV prediction. As the transaction data contain almost 350 million rows, which is not possible to predict such large data. So that data sampling and cleaning of the data is done in pre-processing stage. After that, different regression models used to predict CLV. Finally, the model performance can be evaluated in the evaluation. The best model can be choose using the different evaluation metrics.

In recent years, understanding how to properly manage customer interactions has been a major topic among academics and practitioners. Although relationship marketing is not a new concept, companies have only recently started to concentrate on identifying and retaining successful long-term consumers. Maintaining and satisfying existing customers is far more profitable than regularly renewing a customer base with a high attrition rate. In the service industry, there is significant worry about customer retention. Identification of clients who are likely to switch is a prime objective.

A focus on performance and relationships involves analysis of customer lifetime value and customer equity, both of which demand long-term outcomes. In performance measurement, models for long-term consequences, such as customer happiness, should play a greater role (Rust, 2020). The connection between customers and e-commerce is mainly a virtual one, customer trust in the company is fundamental. Customer trust can also be built through enhancing the company's image. It is predicted that through strengthening the image, consumer trust would rise, which will have a significant impact on increasing customer frequency and monetary value (Dachyar et al., 2019).

It is critical to develop a successful and precise user Customer Churn Prediction (CCP) model in order to control customer churn. A churn prediction technique is built using statistical and Data mining (DM) models. Several DM approaches have been proposed for detecting loyal subscribers who are known to churn. Clustering, association, rule induction, Genetic Algorithm (GA), Decision Tree (DT), Neural Network (NN), and other applications utilise DM models. Users are viewed as a vital asset for commercial sectors in large, dynamic, and competitive enterprises within a market (M, 2020a).

The sum of all customers' CLVs is known as customer equity, and it is often used as a proxy for a company's value. This is especially true in-service industries, where customer equity is frequently significantly higher than the company's other assets. Because the CLV is determined by future revenues earned from a client, it is contingent on the customer's future behaviour, which is unknown in advance. The more accurately this behaviour is characterized, the more accurate the determined CLV value is as an indicator for a customer's worth. Estimating CLVs of individual consumers or client segments for the purpose of monitoring a company's marketing activity can provide a considerable competitive advantage, especially in competitive marketplaces. Exploiting existing and available customer data using a dynamical customer model for predicting an organization's future profit, on the other hand, is a highly complicated and knowledge-intensive task. This activity is normally carried out by analytics experts, either as an internal service provided by a business analytics department to the organization's decision makers or as a service provided by a specialized marketing analytics agency (Heitz et al., 2011).

2. Literature Review

2.1 Introduction

The literature review chapter will provide a brief overview of past study of CLV as well as the methods and techniques that will be used to complete this project. The study will look at a potential gap in the present research on customer lifetime value prediction using Machine learning and neural network.

In today’s competitive business world, the Customer Lifetime Value Model has considered as a major business approach, therefore calculating and improving the CLV has become increasingly important, where the life cycle of products and services and the duration of comparative advantages stability are becoming shorter. Focusing on valuable customers and creating a long lasting and valuable communication with them has become increasingly important. CLV, defined as the present value of all future earnings acquired from a customer over the course of their association with a corporation is crucial in establishing a marketing strategy, and methods based on it can lead to higher profitability for a company (Rezaei et al., 2021).

CLV is frequently used by businesses like as retail, telecommunications, and others to monitor and measure their success. Companies are under increasing pressure to make marketing responsible. Traditional marketing criteria such as brand awareness, attitudes, or even revenue and market share are insufficient to demonstrate a return on investment in marketing. The following are some of the numerous possible benefits provided by CLV that companies are becoming increasingly aware of: Increased customer retention and loyalty, Increased customer profitability, Customer value creation, Product and service customization, and Lower process, higher quality products and services. (Tarcza et al., 2018).

To produce more accurate predictions, established literature on CLV uses numerous data mining technologies, such as neural networks, clustering, decision trees, regression, support vector machines, and ensembles of hybrid methods. Data mining technologies can offer new opportunities by providing business analytics based on detailed customer data. One of the finest supporting tools for making distinct CLV decisions is appropriate data mining technologies, which are good at collecting and extracting important information and knowledge from vast customer databases.

2.2 CLV Modelling

In E-Commerce or retail, the relationship between the corporation and the buyer is an on-contractual relationship. (Riyanto, 2021) Customers that leave in the non-contract world do it silently; they don't have to inform us they're leaving. As a result, calculating the CLV becomes considerably more difficult. To assess whether a customer is alive but inactive or "dead," we would look at the time since their last transaction ("on "means the customer interacted with us, "dead "means they have become in active as a customer). The consumer lifetime for non-contracted businesses, (Glady et al., 2009b) proposed a new framework for the customer churn identification procedure in their research. This paper adds to the current writing by utilizing the client lifetime value as a premise idea for modelling and predicting churn in a non-contractual situation.

That is, when the future duration and modalities of the customers' relationship with the focal company are unknown. They anticipate churn based on the time-slope of the customer lifetime value, shifting from a product-centric to a customer-centric perspective. The approaches yield excellent outcomes in terms of the given profit measure, highlighting the need of correctly classifying potentially profitable churners in addition to attaining a good overall classification. This project discusses how an accurate forecast of a customer's future spending evolution can help improve relationship marketing decision-making for long-term customer.

For that divided the group of long-term consumers into two subgroups, primarily consisting of customers who increased or decreased their spending during the course of their connection with the organization. As a result, they use Bayesian network classifiers and information from first purchase occasions to predict the sign of the slope—which is the output of estimating a linear customer lifecycle—at the individual customer level. CLV is typically defined and estimated at an individual customer or segment level. This allows us to differentiate between customers who are more profitable than others rather than merely examining average profitability (Chen, 2018). The CLV for a customer is calculated using the formula (Gupta et al., 2004):

CLV=∑Tt=0(Pt−Ct)rt(1+i)t−ACCLV=∑t=0T(Pt−Ct)rt(1+i)t−AC

where:

Pt= price paid by a consumer at time t.

ct= direct cost of servicing the customer at time t.

rt= discount rate or cost of capital for the firm.

i= probability of customer repeat buying or being ‘alive’ at time t.

AC = acquisition cost.

T = time horizon for estimating CLV.

2.2.1 Traditional CLV RFM approaches

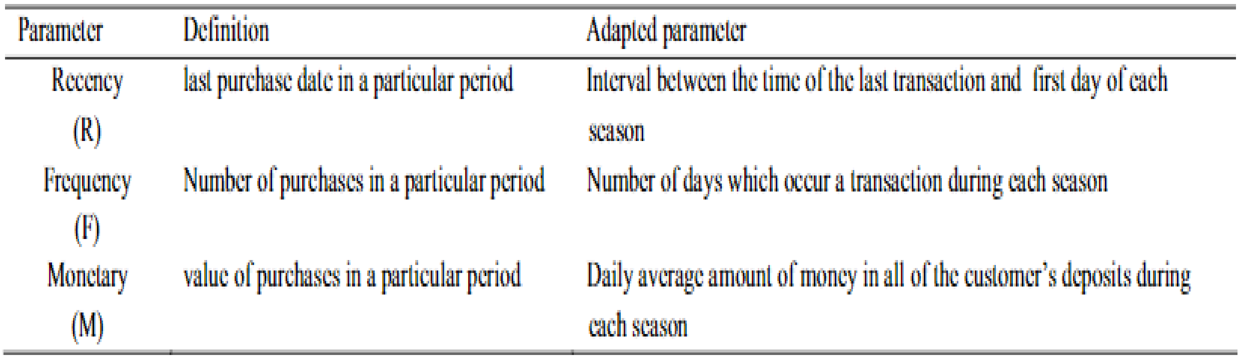
Data mining (DM) and customer behaviour analysis are the most prominent approaches for segmenting customers. The Recency Frequency Monetary (RFM) model is the most extensively used technique for analysing customer behaviour. Businesses collect a tremendous amount of data as a result of using IT, thus extracting useful information from that data is critical. Inside the data, DM techniques can uncover substantial underlying trends and connections.

Traditional classification is outperformed by Customer Segmentation (CS) based on DM in terms of accuracy. RFM analysis is a standard behaviour segmentation that uses transaction data to understand client buying behaviour. The original RFM model had three variables: recency (the last time since transaction of customer relations with the company), frequency (the number of purchases made by consumers), and monetary (the amount of money paid by customers in each period). Customers who are valued have the highest frequency and monetary value, as well as the lowest recency (Ernawati et al., 2021).

The proposed framework for applying DM approaches in connection with RFM-based CS is shown in Figure 2.1. The descriptive DM (association or sequence discovery) function of clustering is effective for finding the pattern or relationship between data. Simultaneously, predictive DM (classification, forecasting, or regression) predicts customer behaviour and uses visualization to better understand the characteristics of the client segment. Then use statistical profile as descriptor variables to visualize consumers' locations. This framework enables analysts to use DM approaches to better analyse and identify client behaviour, allowing businesses to design effective marketing strategies to gain a competitive advantage.

Mean-shift, DBSCAN, Agglomerative Clustering, and K-Means are the clustering techniques employed. Mean-shift Clustering is a centroid-based technique for updating candidates to the mean of the points inside a region.

To generate the final collection of centroids, the candidates are filtered to remove any close duplicates. Mean-shift can detect blobs in a smooth density of samples in this way. Clusters are shown by the DBSCAN algorithm as areas of high density separated by areas of low density. DBSCAN can be any form, not only convex, according to this general viewpoint. Agglomerative Clustering uses hierarchical clustering, which is a sort of clustering method that creates nested clusters by gradually merging or separating them. Agglomerative Clustering, on the other hand, works from the bottom up. This method will gather observations from each cluster and blend them together in a sequential manner. The K-Means algorithm clusters data by attempting to divide samples into 'n' equal-variance groups. K-Means, unlike some other clustering methods, requires the number of clusters to be specified (Parikh & Abdelfattah, 2020).



Parameter	Definition	Adapted parameter
Recency (R)	last purchase date in a particular period	Interval between the time of the last transaction and first day of each season
Frequency (F)	Number of purchases in a particular period	Number of days which occur a transaction during each season
Monetary (M)	value of purchases in a particular period	Daily average amount of money in all of the customer's deposits during each season

Fig. 2.2 Definition of RFM and Adapted RFM model parameters

In Figure 2.2, Customers were divided into groups using customer segmentation techniques. One of the most powerful and simple models that satisfy the last requirement is one with homogeneous segments. CRM implementation and customer value estimation using the RFM model (Recency, Frequency, and Monetary Value) was used. The research establishes a framework for segmenting clients into homogeneous groups and calculating CLV. Finally, evaluating the future value of each segment's customers by evaluating the customer's trend, by using the time series method, one might calculate the value in different seasons (Khajvand & Tarokh, 2011).

A process that combines RFM characteristics and the K-means algorithm into rough sets theory (the LEM2algorithm) has been presented not only to improve classification accuracy but also to extract classification rules for obtaining an outstanding CRM for businesses. Furthermore, it has the potential to successfully mitigate some of the disadvantages of data mining technologies. RFM features and the K-means algorithm can improve a corporation in classifying customer segmentation objectively. Based on the good findings of the experiment, this project believes that it will assist companies in focusing on their target customers and maximizing earnings in a win–win situation for both the firm and the buyer. Based on the findings, the organization can develop new customer-centric strategies to boost customer loyalty and maximize customer lifetime value. The company focuses more on high CLV valued customers because they are more profitable and have been loyal to the company for a long time. Companies treated customers differently based on CLV values since each company has limited resources, and management aim to maximize benefits within those resources. The firm's decision-makers make sales decisions by paying more attention to the most valuable clients, as determined by the CLV values of the clusters (Cheng & Chen, 2009).

Market segmentation enables retailers to precisely reach consumers with specific needs and wants by dividing the market into similar and identifiable segments, allowing retailers to focus on individuals with similar preferences, choices, needs, and interests. In the fast-changing retail industry, there is a clear need for advanced methods to discover market segments from sales and other data, with market segmentation empowering retailers to precisely reach consumers with specific needs and wants by dividing the market into similar and identifiable segments, to focus on individuals with similar preferences, choices, needs, and interests. Customers are evaluated as a section, rather than individually or directly, in segmentation. It helps companies to make the most of their limited resources by successfully serving customers in sub-groups. The RFM approach divides the market into parts based on demographic and behavioural factors. In two data mining jobs, a unique step-by-step technique with CLV and RFM analysis is suggested. For each RFM variant, regression and clustering algorithms are used individually. The research focuses on how to treat customers based on their buying habits. A novel modified regression technique and RFM variable are proposed to transform this into a computable parameter (C) (Yoseph & Heikkila, 2019).

The data mining process explained below:

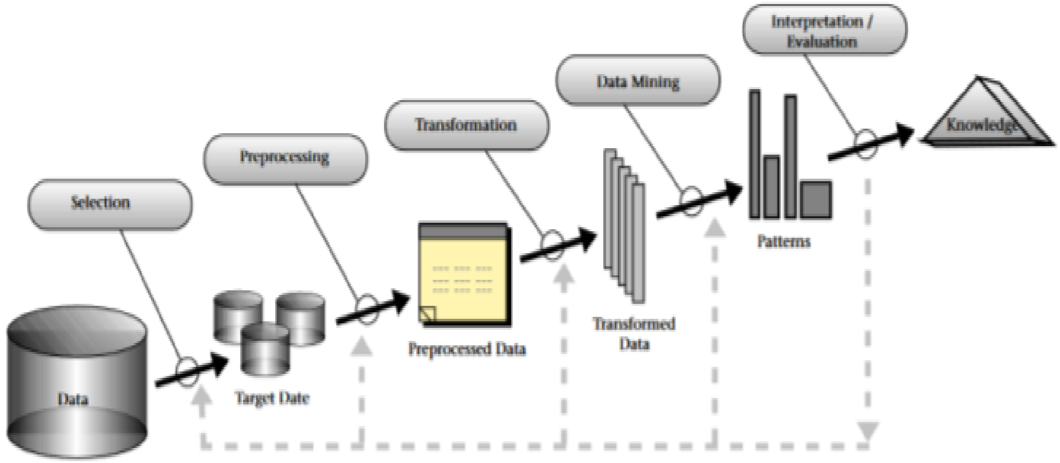


Figure 2.3 Data mining process

In figure 2.3 data mining techniques were applied. There are four stages to the analysis. The data pre-processing, which includes data cleaning, feature selection, and data transformation, is the objective of the first phase. In the second and third phases, regression and clustering techniques are used, respectively. The data pre-processing phase continues to deal with noisy data, missing values, and attribute reduction and transformation. To make pattern discovery easier, the data must be converted into a right format in this stage.

2.2.2 Machine Learning based approaches

For customer identification, customer acquisition, customer retention, and customer development, the CRM uses machine learning algorithms. The most often used supervised learning algorithms in CRM are decision tree-based algorithms and support vector machine techniques. (Singh et al., 2020), said that most crucial industries associated with the exponential expansion of users are e-commerce and telecommunications, which require appropriate machine learning approaches for client satisfaction and business profitability. The most widely used ML technique in CRM is supervised learning, which includes decision trees and the Support Vector Machine (SVM) algorithm. As a result, 48.48 percent of supervised learning approaches, 15.15 percent of unsupervised learning techniques, and 9.09 percent of other techniques have been used in CRM.

Deep learning is also mentioned in 27.27 percent of articles, indicating that future trends are changing to deep learning due to its ability to learn and grow on its own. A dynamic and data-driven framework for forecasting whether a consumer will make a purchase at the company in the near future within a particular time range (Martínez et al., 2020), propose a new collection of customer-relevant attributes derived from the timings and values of past purchases for this purpose. Monthly, these customer features are updated, and state-of-the-art machine learning algorithms are used to anticipate purchase behaviour. For this, we created a collection of feature variables that characterize the consumer at different times of the year. We used several art machine learning algorithms. The gradient tree boosting outperforms the Lasso and the extreme learning machine, according to the results. These findings are consistent with reports that boosted decision trees outperform other machine learning algorithms.

Multiple linear regression is employed as a typical data analysis strategy for modelling CLV in this study. A method of modelling the connection between a numerical dependent variable and two or more explanatory variables is known as multiple linear regression. Traditional data analytics approaches such as multiple regression are combined with social network analysis methodologies for the airline sector. Customers' travel data and social network data, in particular, are used as independent variables in a multivariate regression technique to calculate customer value. Two models were created to demonstrate the contribution of social network information, one comprising only flight related elements and the other containing both flight related and social aspects. Both the model's accuracy and prediction ability might be enhanced by incorporating social elements into the underlying model.

Although this project uses airline customer value determination to demonstrate the integration of social network information with data analytical tools, the suggested method may be applied to models in other domains (Çavdar & Ferhatosmanoğlu, 2018).
(Win & Bo, 2020) present a customer class prediction model that use the random forest algorithm to accurately categorize individual online retail customer classes in terms of lifetime value. Using selected characteristics and optimum parameters from Random Search, we demonstrated that our model worked consistently well. The approach can assist online merchants in determining which kind of customers

require the most work in order to sustain a retention plan. Simple models do relatively well, according to one of the most noticeable findings of (Donkers et al., 2007) analysis. These basic models, on the other hand, must account for customer retention as well as cross-selling. Models that simply account for customer retention miss an essential aspect of a customer's worth and, as a result, underpredict. Due to the fledgling nature of CLV research, these findings may have broader implications beyond only the insurance industry. The models' anticipated CLV can be utilized to segment customers. Firms can build various service concepts for each segment using this segmentation, and high-value clients can choose their preferred channel. Customers of little value receive poor service and are directed to low-cost channels such as the Internet. Only apply these segmentations if CLV can be anticipated with reasonable accuracy, according to the recommendations. The study offers consumer intelligence managers different CLV prediction methodologies that can evaluate in their market.

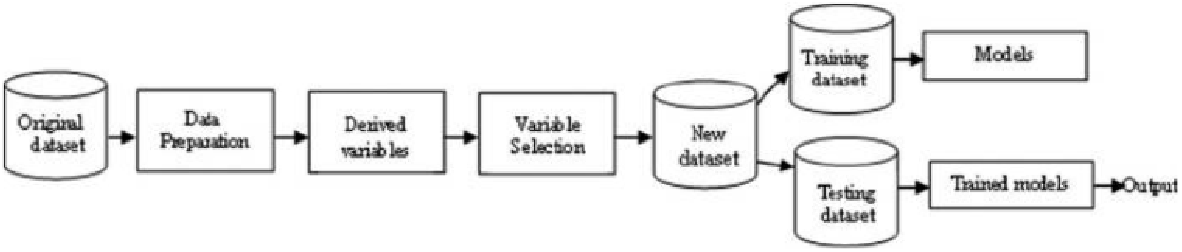


Fig 2.4 Basic Machine Learning pipeline

The basic machine learning pipeline is in figure 2.4. The pipeline shows that the original dataset is used for data preparation. The variable selection is the most significant part before modelling. Appropriate variables or features can be selected from the data and these features is used for modelling. A new dataset will be created after data preparation and feature selection. The new dataset is used for training and testing. The output from training and testing is the final output. The same approach is used in this paper, instead of feature selection feature engineering will be used to find the CLV. The engineered features is saved in a new dataset and this dataset is used for training and testing. The strongest basis model, random forests, was established in 2001 and popularized in the following years. Random forests are particularly good at handling noisy data, are very resistant to overfitting, and can be called an all-purpose model that requires even less parameter tuning than boosting. Random forests are made up of a lot of deep but interconnected trees that are constructed from diverse data samples. Have a great deal of potential because they're easier to use than boosting. The algorithm for creating a random forest is rather straightforward. Draw a random subset from the original training data for each tree in the random forest. Then, for this sample, create a modified decision tree in which each split selects attribute at random. The final product is an ensemble of random forest trees, which may be classified using a majority vote (Krauss et al., 2017).

A classic statistical tool for categorization is logistic regression. This method is a specific type of linear model. It uses binary dependent variables instead of continuous variables and assumes a linear relationship between the dependent variable and the independent variables when written in the logit scale, i.e., when using a log-odds transformation. The dependent variable in logistic regression represents the probability of an event occurring, and it is constrained between 0 and 1. Ordinary linear regression approaches are ineffective for estimating the dependent variable because they allow it to move outside the 0 1 range. Assume x is a vector of independent variables, and P (Y = 1x) is the probability of the event Y occurring (Miguéis et al., 2013).

The decision tree is a well-known technique that has been used to a variety of real-world challenges with great success. A decision tree is a symbolic learning technique that uses nodes and ramifications to organize information gathered from a training dataset in a hierarchical structure. It is simple to understand the outcomes of decision trees because the output can be structured in the form of a tree or rules. A decision tree may create models from a variety of datasets, including numerical and categorical data. Classification trees and regression trees are the two basic forms of decision trees. The target variable has discrete values, and the decision tree associates a probability for each class in a classification tree with each leaf node. Decision trees have been widely employed in various sectors, such as supplier selection, email churn prediction, and credit card churn prediction, since they are simple to understand (Nie et al., 2011).

2.2.3 Deep Learning based approaches

When it comes to anticipating churn in the grocery retail business, it relies on two factors. The first is based on the characteristics that will be used. The model was handed on to Rather of relying on client purchases, these values will be developed as a result of trends to cluster individuals. They are provided to the model as features. As a result, for each numerous features are built for the consumer to allow the model to Individual patterns should be learned and identified. As a result, there are two Datasets are established to analyse and test how data should be handled. portrayed to forecast churn The second consideration is the algorithms' implementation This research is unique in that it is the first of its kind. Deep learning is being used to forecast churn at the grocery store industry (Dingli et al., 2017).

Vector embedding is one of the approaches that may be used to construct an explainable model. For each data point it represents, the model must create an informative vector using this method. By allowing the model to execute a certain job, desired information may be incorporated into the vector. Hidden knowledge in the data can be discovered by inferring the infused information in the vector. A deep learning model can operate as a decision support system for a telecommunication firm thanks to it explain ability via vector embedding. This might aid the organization in making a decision based on consumer behavior, as decision support system recommendations are significantly connected with customer behavior (Cenggoro et al., 2021).

All of the network's connected weights are gradually modified based on this inaccurate value, working backwards from the output layer, via the hidden layer, and to the input layer, until the right output is created. By fine-tuning the weights in this way, the network is shown how to produce the correct output for a given input.

The two tables below are the summary of existing approaches from the literature review. Both tables provide a reference guide for how your models should perform in terms of accuracy and highlight the training time needed for models to produce accurate predictions. The Table 2.1 provides all the details about the CLV prediction using regression which highlight metrics used for the accurate prediction. Table 2.2 gives information about the CLV prediction using classifications.

CLV using regression

Researcher	Method	Training set size	Predict Horizon	Model Accuracy	Error/ performance
(Singh et al., 2018)	Gradient boosting algorithm	48 weeks	72 weeks	80%	mean square error
(Chen et al., 2019b)	DNN	48 weeks	48 weeks		RMSLE, NRMSE, SMAPE
(Jasek et al., 2018)	Extended Pareto/NBD model (EP/NBD), Markov chain model, Status Quo model	52 weeks	Long 52 weeks & short 13 weeks	82%	MAPE, MAE
(Chamberlain et al., 2017)	Random Forests, logistic regression, DNN	48 weeks	48 weeks	95%	Root mean squared error (RMSE)
(Glady et al., 2009a)	Pareto/NBD, Gamma/Gamma Submodel	48 weeks	48 weeks	946.2, 411.9	Root Mean Square Error (RMSE), Mean Absolute Error (MAE)
(Donkers et al., 2007)	Markov model	96 weeks	48 weeks	78%	Root Mean Square Error (RMSE), Mean Absolute Error (MAE)
(M, 2020b)	P-AGBPNN, NN			91.71%	sum squared error

Table 21 CLV using regression

It can be clearly seen in above table 2.1 that the developed models were within 48 to 52 weeks of data to train. So that, my research will use 26 weeks' worth of data to produce predictions. This benefits companies as a faster estimation of customer value are available, enabling companies to target higher value customer earlier.

CLV using classification: high value or low value customer

Researcher	Method	Training set size	Predict Horizon	Error/ performance
(Glady et al., 2009b)	Logistic regression Neural network Decision tree AdaCost Cost-sensitive tree	24 weeks	12 weeks & 24 weeks	AUROC PCC
(He et al., 2020)	Logistics Regression, Random Forest, Extremely randomized trees classifier (Extra Trees Classifier), Support vector machine (SVM), Neural network (NN), Gradient boosting method (GBM)	12 week (90 days)	12 weeks	Area Under the Curve (AUC)
(Almuqren et al., 2021)	Sentiment analysis	24 weeks	24 weeks	AUC ROC
(de Caigny et al., 2020)	CNN	24 weeks	24 weeks	AUC
(Martínez et al., 2020)	Logistic Lasso regression Extreme learning machine Gradient tree boosting	24 weeks	24 weeks	AUC
(Win & Bo, 2020)	Random forest	48 weeks	48 weeks	precision, recall, accuracy

Table 2.2: CLV using classification of high value or low value customer

From Table 2.2 it can be clearly seen that the developed models were within 12 to 48 weeks. Most of the model's performance were measured using Area Under Curve (AUC).

3. Methodology

3.1 Introduction

In this chapter, we discuss about the classic CRoss-Industry Standard Process for Data Mining (CRISP-DM) to learn how the CRISP-DM Model works and how organizations can utilize it as a data mining technique in six phases for their data mining initiatives. CRISP-DM was created to help beginners understand data mining and to give a generic process model that can be customized to meet the demands of any industry or firm. The methodology's early application in the business demonstrates that it is a useful tool for both beginners and expert data miners.

3.2 Data life cycle/ CRISP-DM

The data mining project follows several methodologies in developing the project from obtaining the dataset to visualizing the result. This research project adopts the CRISP DM methodology for building the data science project. CRISP-DM stands for Cross Industry Standard Process for Data Mining.

The different stages of CRISP-DM methodology are listed below:

- Business understanding
- Data understanding
- Data preparation
- Modelling
- Evaluation
- Deployment

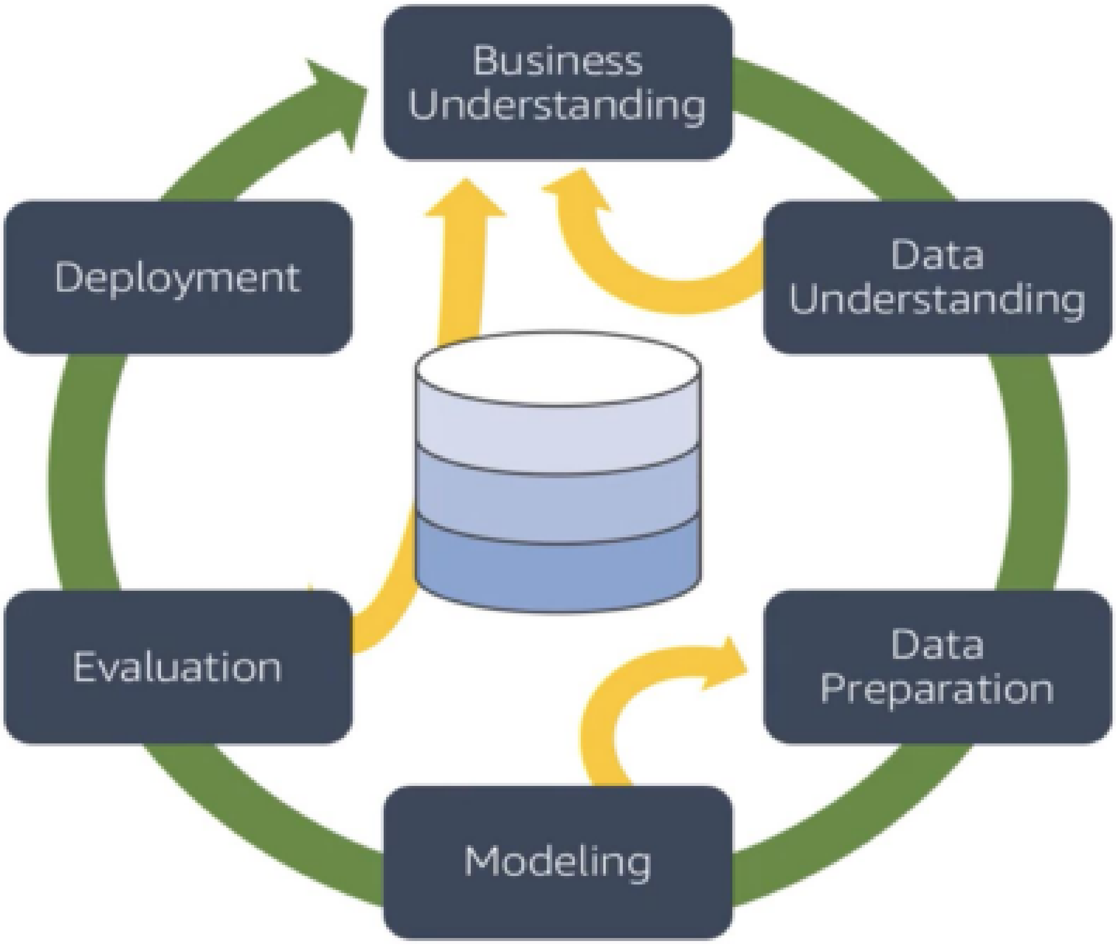


Fig. 3.1: different stages of CRISP-DM

The CRISP-DM is the most commonly used technique in 42% of organizations surveyed, followed by companies utilizing their own methodology (28%), it's no surprise that CRISP-DM has been employed in several studies on DM application in various fields. As a result, those studies assessing the effectiveness of DM programs
from a methodological standpoint have mostly focused on CRISP-DM stages.

3.3 Business Understanding

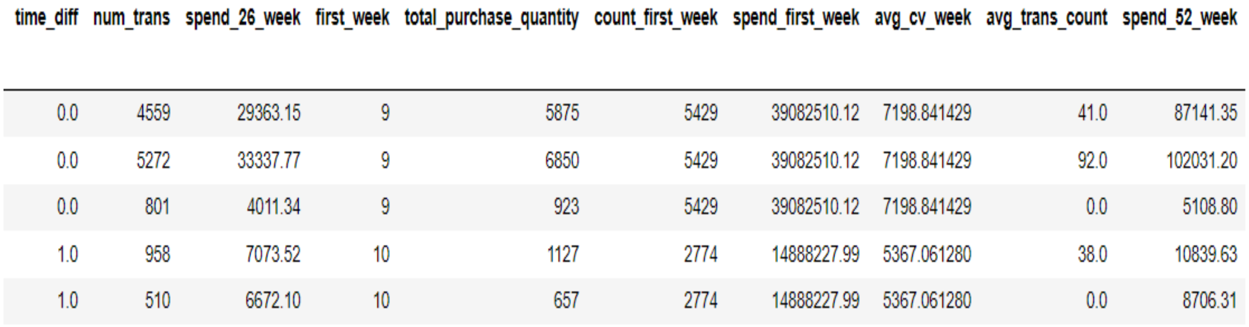
This phase focuses on gaining a business understanding of the project objectives and needs, then transforming that information into a data mining issue definition and a preliminary strategy to fulfil the goals. In this research the problem is identifying the behaviour of the customer and how they are loyal to the company. When we have distinguished the issue that we need to resolve by means of Machine learning and find the best model to predict CLV.

3.4 Data understanding

The data understanding phase begins with data collection and progresses via procedures to familiarize yourself with the data, find data quality issues, get early insights into the data, or locate intriguing subsets in order to create hypotheses about hidden information. Acquire Valued Shoppers Challenge dataset is publicly available dataset. The dataset is collected from Kaggle public repository. The data is in large blocks of CSV files. CSV file has 350 million rows and have transaction data of 300000 shoppers are available. The data having a large details like ID, chain, dept, category, company, brand, date, product size, product measure, purchase quantity and purchase amount.

3.5 Data preparation

The data preparation phase contains all processes that result in the final dataset (data that will be given into the modelling tool(s)) being developed from the raw data. Tasks for data preparation are likely to be repeated several times and in no particular order. Table, record, and attribute selection, as well as data processing and cleansing for modelling tools, are all tasks. From the dataset it is evident that feature engineering is needed before modelling. More features were created using the existing features. The total purchase amount is calculated using purchase quantity and purchase amount. And create more features from date and so on. The engineered features are shown below.



time_diff	num_trans	spend_26_week	first_week	total_purchase_quantity	count_first_week	spend_first_week	avg_cv_week	avg_trans_count	spend_52_week
0.0	4559	29363.15	9	5875	5429	39082510.12	7198.841429	41.0	87141.35
0.0	5272	33337.77	9	6850	5429	39082510.12	7198.841429	92.0	102031.20
0.0	801	4011.34	9	923	5429	39082510.12	7198.841429	0.0	5108.80
1.0	958	7073.52	10	1127	2774	14888227.99	5367.061280	38.0	10839.63
1.0	510	6672.10	10	657	2774	14888227.99	5367.061280	0.0	8706.31

Table 31: Engineered features

3.6 Modelling

Various regression modelling approaches such as linear regression, lasso regression, ridge regression, decision tree regressor and random forest are chosen and employed in this phase, and their parameters are calibrated to ideal levels. For the same data mining issue type, there are usually many approaches. Some approaches have special data format requirements. As a result, it's common to have to go back to the data preparation stage. Selecting models which is going to be used in the research project is the first step in the data modelling process. Selecting the suitable model includes a review of literature review and finding the commonly used predictive techniques which are already been successful. Based on the related work, the supervised machine learning algorithm is used in this research project and regression technique predict the quantitative variable. In this project the target variable is quantitative one. So, customer behaviour is coming under regression algorithm. To receive the data from a user as input an interface is created by using python programming language. This interface will generate the output.

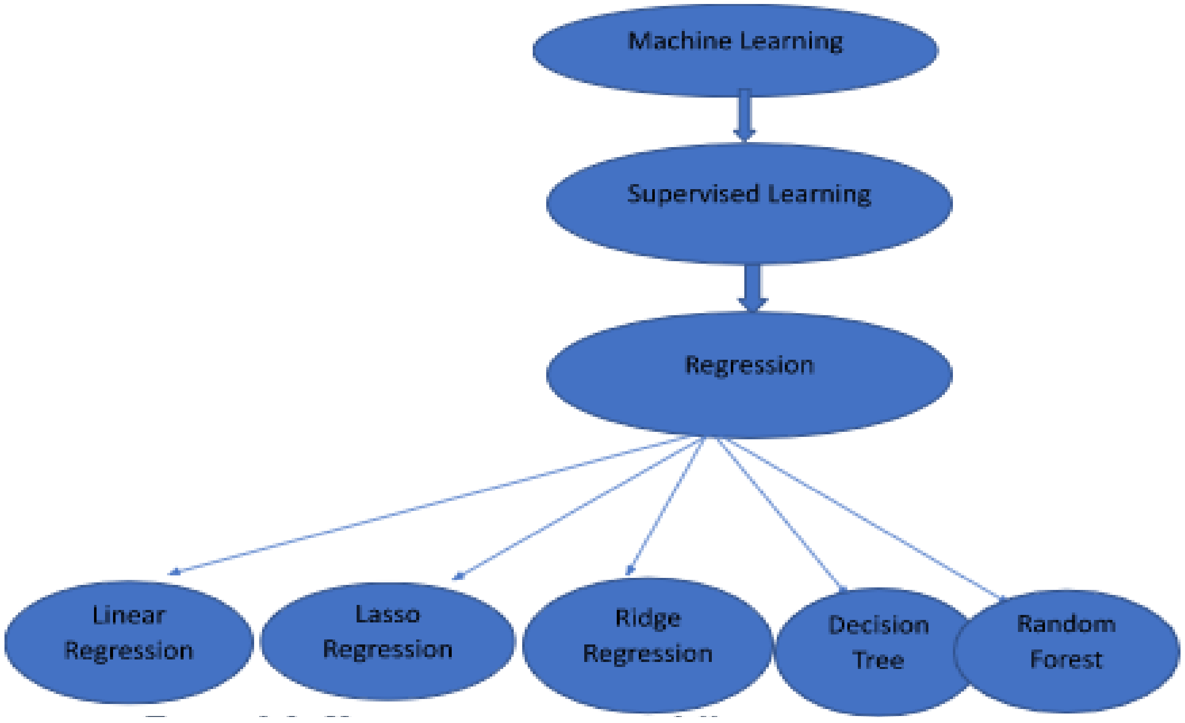


Fig. 3.2 Various regression modelling

3.7 Evaluation

The constructed a model (or models) that look to be of excellent quality from a data analysis standpoint at this point in the project. Before going to final deployment of the model, it is necessary to do a more complete evaluation of the model and assess the actions taken to build it in order to ensure that it meets the business objectives. One of the main goals is to see whether there is any critical business issue that has not been adequately addressed. A choice on how to use the data mining results should be made at the end of this step. The final dataset that used for modelling is 9587 rows and 13 attributes. For evaluating the model, the model accuracy can be measured using R-squared and the errors measured by Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Coefficient of Variation (CVRMSE) of each model.

3.8 Deployment

The model's creation isn't always the conclusion of the process. Even if the goal of the model is to improve data understanding, the information gathered must be arranged and presented in a way that the consumer can understand. The deployment step might be as easy as creating a report or as sophisticated as establishing a repeatable data mining process, depending on the needs. In many circumstances, the deployment procedures will be carried out by the user rather than the data analyst. In any case, it's critical to know ahead of time what activities will be required to really use the models that have been built.

4. Findings and Analysis

4.1 Introduction

This chapter discuss about the analysis and the result of the project done. This section explained as same as chapter Research methodology by following CRISP-DM methodology. This chapter presents the analysis of the research question, beginning with a review of statistics to determine the control variables and the CLV's basic properties. Following that, an analysis of the other factors will be conducted, followed by a critical evaluation of the findings.

4.2 Business Understanding

Business understanding is the key factor to understand the data. This section will focus mainly on the CRISP-DM life cycle to meet the research objectives.

The conceptual map for CLV prediction is:

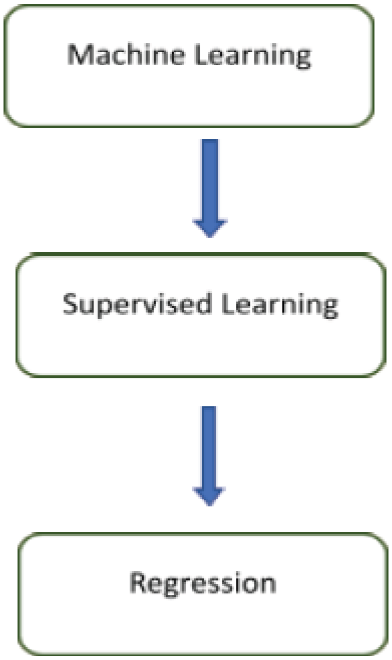


Fig. 4.1 conceptual map for CLV prediction

Steps to be followed in these sections are:

- Analyse the dataset
- Explore the dataset
- Read dataset using the programming language:

Here dataset is in CSV format, read the CSV data using python programming

- Finding missing values and null values
- Data cleansing
- Treat the missing values and null values
- Feature engineering
- Train each of the regression models.
- Score each model according to defined metrics and analyse the results.
- Compare the models in accordance with the scoring metrics to answer the research question.
- Assess the statistical meaning of the outcomes

The different types of regression modelling that this research is going to be utilising are:

- Linear Regression
- Lasso Regression
- Ridge Regression
- Random forest
- Decision tree

Each of these five models will be capable of producing a prediction on customer lifetime value prediction. The research will provide an indication of which of the regression models is the most effective at predicting the result using the following Metrics:

o Root Mean Square Error (RSME)
o R-Squared
o Mean Absolute Error (MAE)
o Coefficient of the Variation of the Root Squared Mean Error or CV(RMSE).

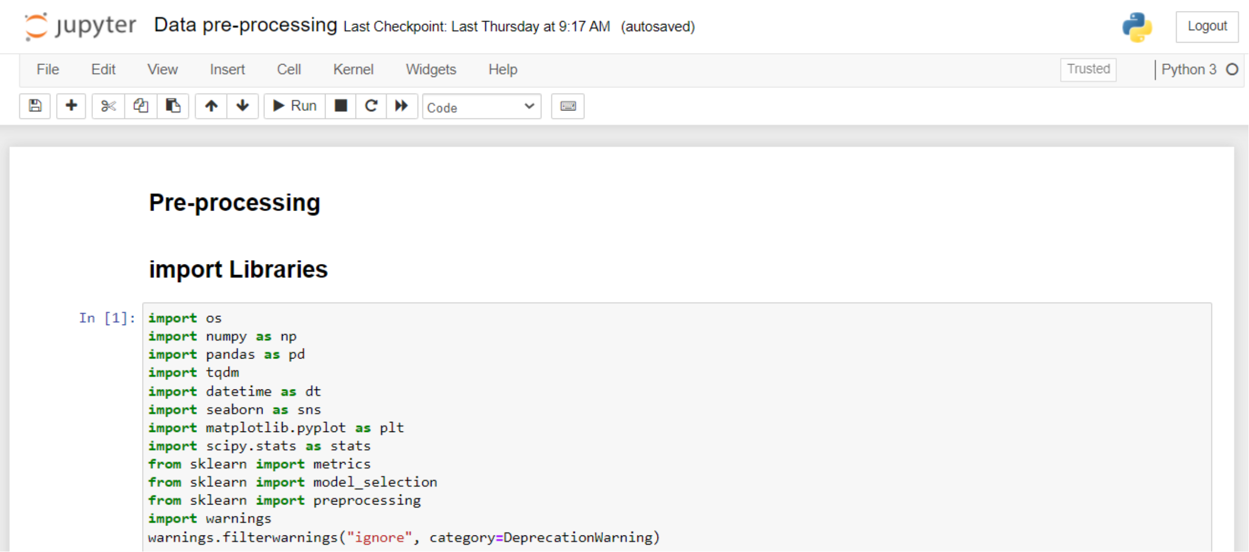
The business success factor of this research is the ability, using a combination of the metrics, to provide a series of rankings, which will be produced for each model, to enable the research to identify the model most capable of predicting CLV performance.

4.3 Data Understanding

In this process the understanding of data is more important to build efficient machine learning algorithms. In this phase we must perceive what sort of information that we have. We must see from where the information is coming from. The" Acquire Valued Shoppers Challenge" dataset is obtained from the Kaggle repository. Nearly 350 million rows of entirely anonymized transactional data from over 300,000 shoppers are available in this collection that is sufficient to construct a significant model. The data having a large number of details like customer id, company, brand, purchase amount, purchase quantity and so on. The data is in large blocks of csv files. There are numerous measurable properties that we can watch that may add the synopsis measurements too, as though there are values that are clear or nulls and how would we have to deal with these, and furthermore if any anomalies and how would we have to deal with these. We can likewise perceive how all the parameters are associated with one another to observe its importance to remaining parameters and to check whether there a solid connection between certain parameters.

4.4 Data Preparation

The data preparation is one of the stages in the methodology process and is also the important stage where the acquired data is transformed into pure quality for the sake of building better machine learning models. The extracted data needs to be pre-processed to understand by machine learning algorithms for classifying the data into specific category with better accuracy. The data is stored in the csv format. The csv file import to data frame using python programming.



Pre-processing

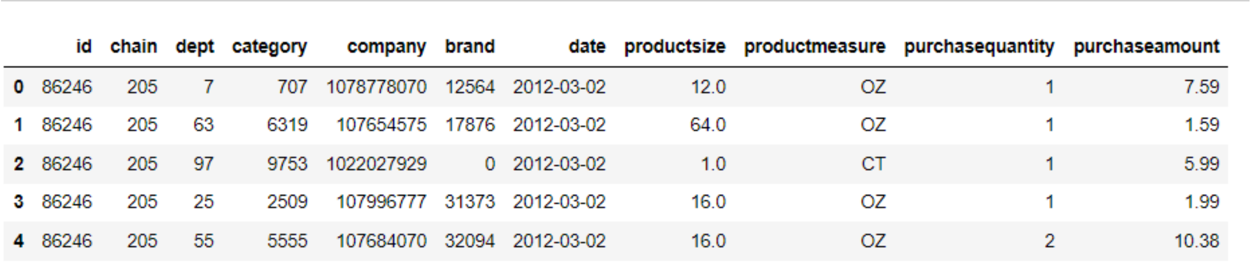
import Libraries

```
In [1]: import os
        import numpy as np
        import pandas as pd
        import tqdm
        import datetime as dt
        import seaborn as sns
        import matplotlib.pyplot as plt
        import scipy.stats as stats
        from sklearn import metrics
        from sklearn import model_selection
        from sklearn import preprocessing
        import warnings
        warnings.filterwarnings("ignore", category=DeprecationWarning)
```

Fig. 4.2 Import libraries

We applied libraries for data preparation and modelling are NumPy, Scikit-learn, pandas, Matplotlib and seaborn. The original dataset is a large csv file and has 3 GB data. It is hard to read the large dataset, in order to read the data, the large csv file is subset to 300,000

data for better analysis using chunk function. The sample data is saved to another csv file. Now the csv has 10000000 rows and 11 attributes. The data frame is shown below:



	id	chain	dept	category	company	brand	date	productsize	productmeasure	purchasequantity	purchaseamount
0	86246	205	7	707	1078778070	12564	2012-03-02	12.0	OZ	1	7.59
1	86246	205	63	6319	107654575	17876	2012-03-02	64.0	OZ	1	1.59
2	86246	205	97	9753	1022027929	0	2012-03-02	1.0	CT	1	5.99
3	86246	205	25	2509	107996777	31373	2012-03-02	16.0	OZ	1	1.99
4	86246	205	55	5555	107684070	32094	2012-03-02	16.0	OZ	2	10.38

Table 4-1: Dataset to Data Frame

Furthermore, the converted data is then processed with several steps such as rename columns with appropriate names, checking the null values, missing values. The original dataset has some missing values in the column product measures, then removes all the missing rows.Feature engineering is the next step, it is the most significant step before modelling. Create useful column to find customer value. A new column totalpurchase is created using the two existing columns, total purchase is the multiplication of purchase quantity and the purchase amount. Thereafter create more column such as year, month, week from date column.

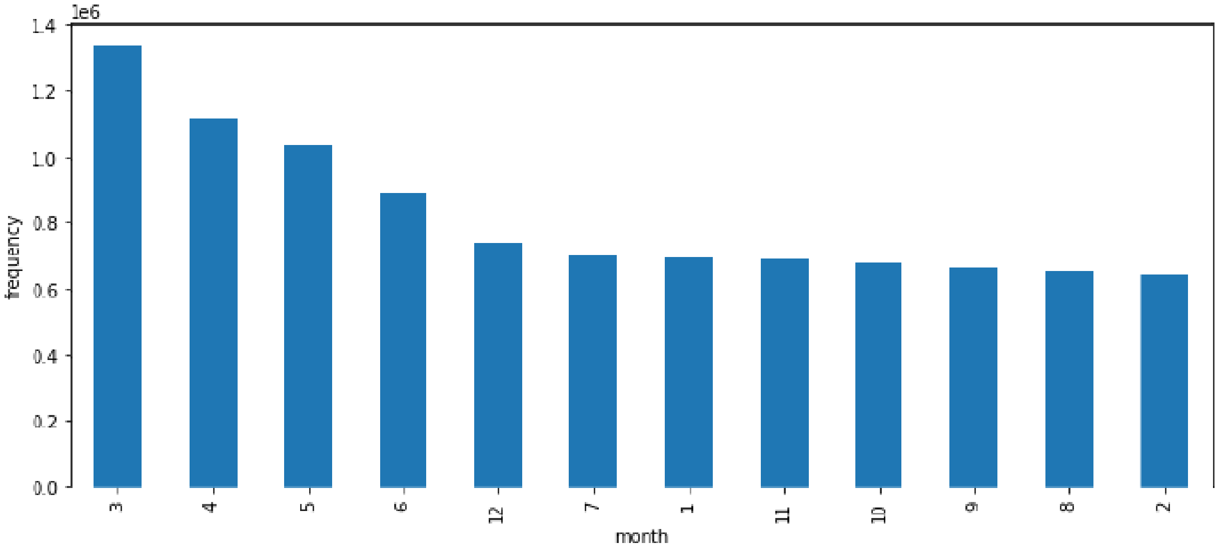


Figure 4.3:Transaction by month

From the above figure 4-3 transaction frequency by month, it can be clearly seen that most transactions occur in month three to six, people transact more in March followed by April, May and June and least in February. Rest of the months are quite similar which means the transaction frequency are approximately 2500.

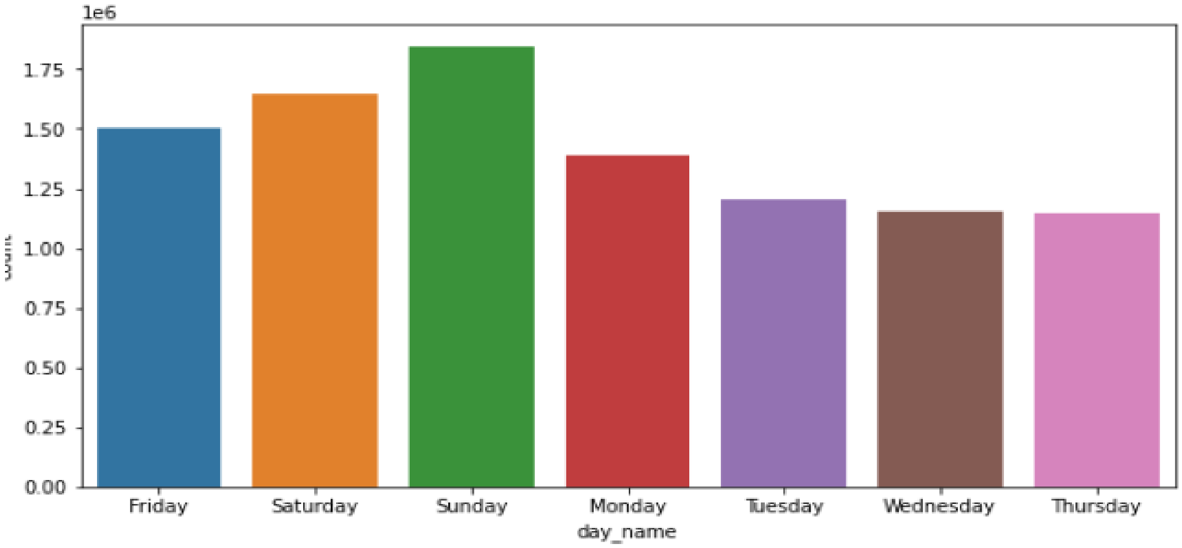
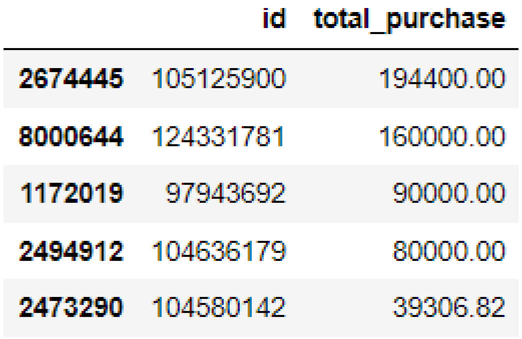


Fig. 4.4:Transaction by days

The transaction count by day in Figure 4.4 gives information about which day in a week has the highest transaction. The diagram indicates that weekends clearlyhave more transaction than traditional weekdays, Friday has the most transact among weekdays followed by Monday. Other days count is almost equal.



	id	total_purchase
2674445	105125900	194400.00
8000644	124331781	160000.00
1172019	97943692	90000.00
2494912	104636179	80000.00
2473290	104580142	39306.82

Table 4-2: Top five customer per purchase

From the table 4-2 above, it is evident that 'customer id' '105125900' is the customer who purchase the most and it was $ 194400. The top five customers are listed in the table.

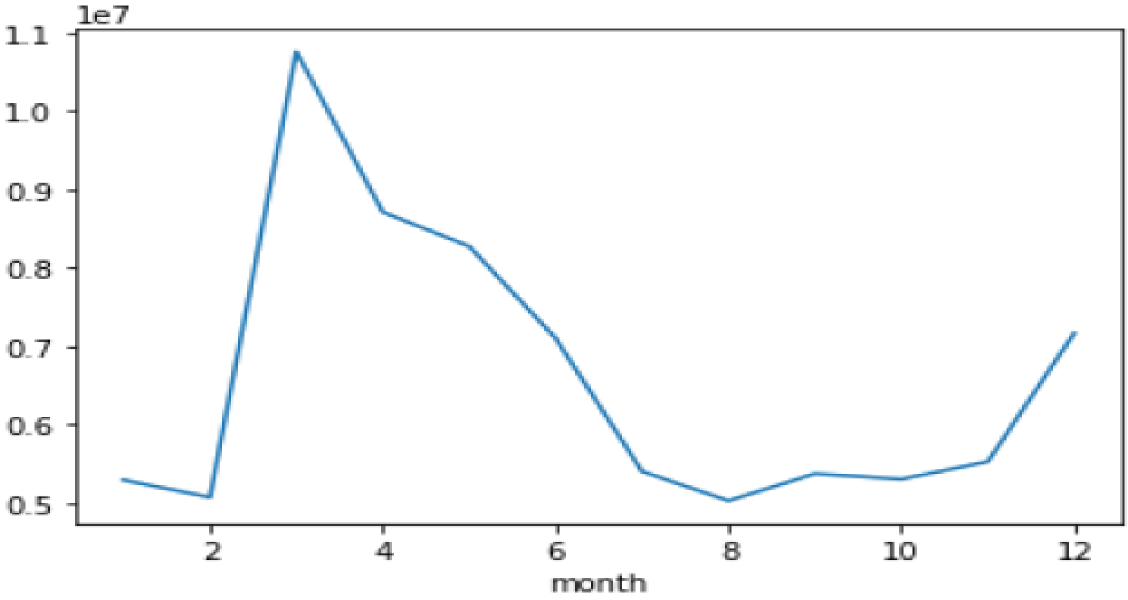


Fig. 4.5:Total purchase versus Month

The total purchases versus month graph figure 4.5 above shows that total purchase is peak in March and drop from May. Whereas February and August have the least purchases and rest of months has lowest purchase amount compared to other months.

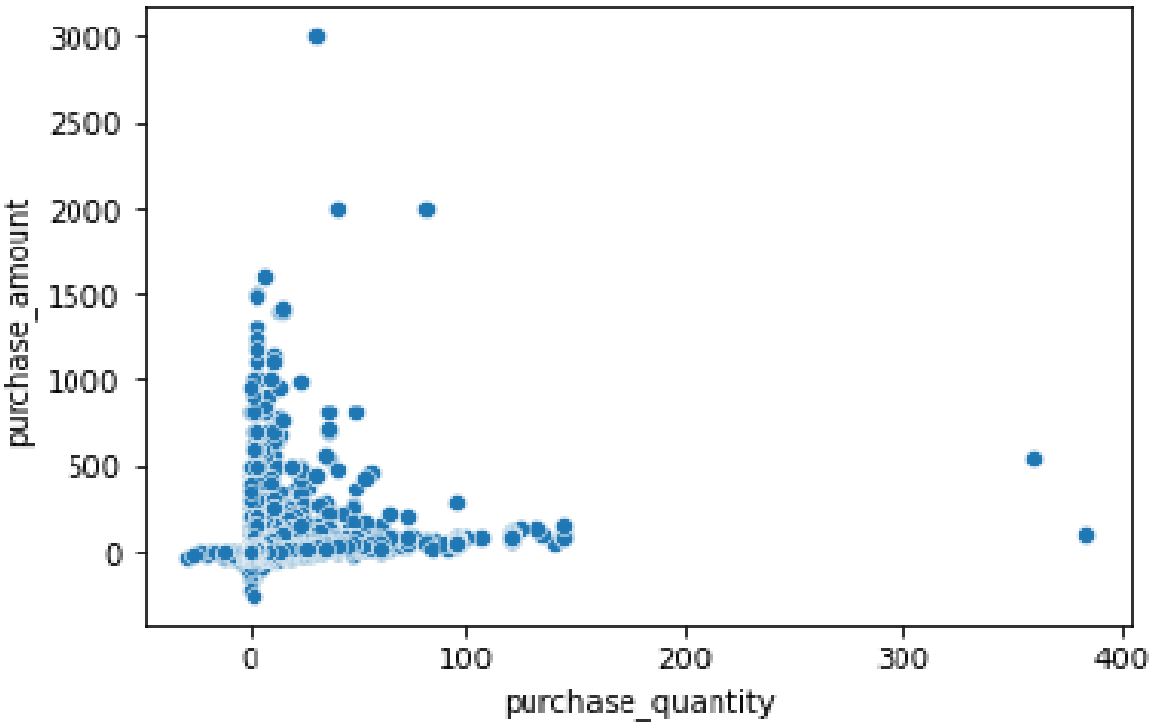


Fig. 4.6: purchase quantity vs purchase amount

The scatter plot between purchase quantity versus purchase amount figure 4.6 clearly seen that purchase quantity and amount purchase have linear relationship with each other. As purchase quantity increases the purchase amount also increases. Purchase quantity is higher from 0 to around 100 and purchase amount higher from 0 to around 1000.

Fig. 4.7 first week count

Fig. 4.7 first week count

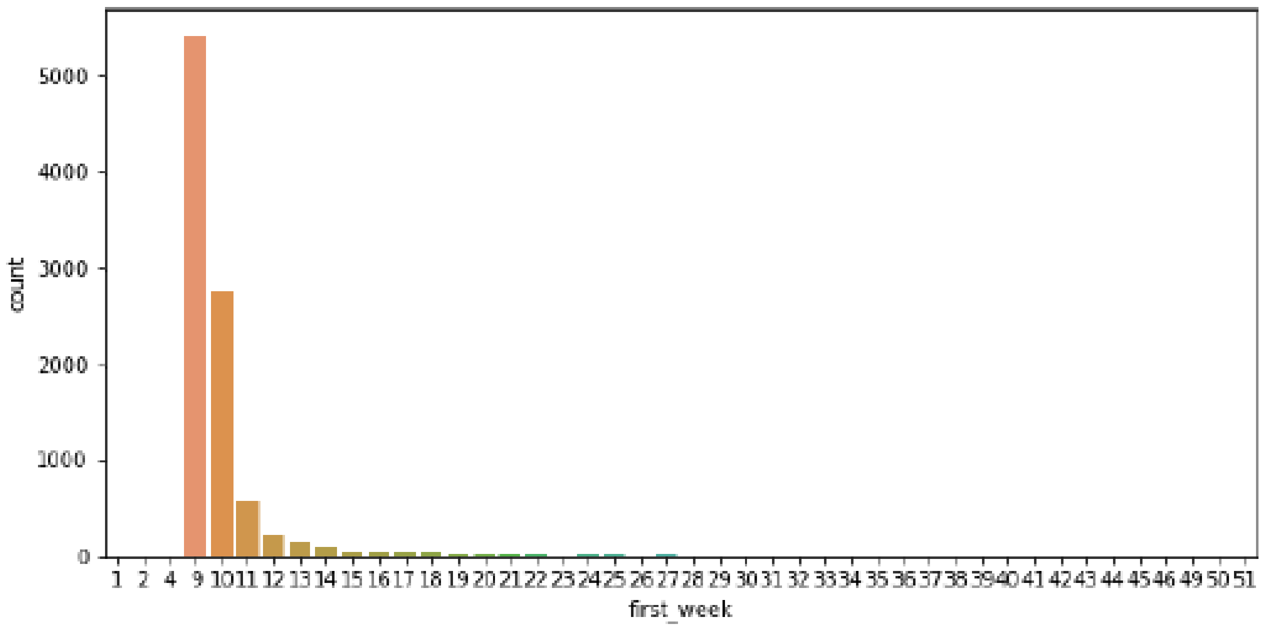
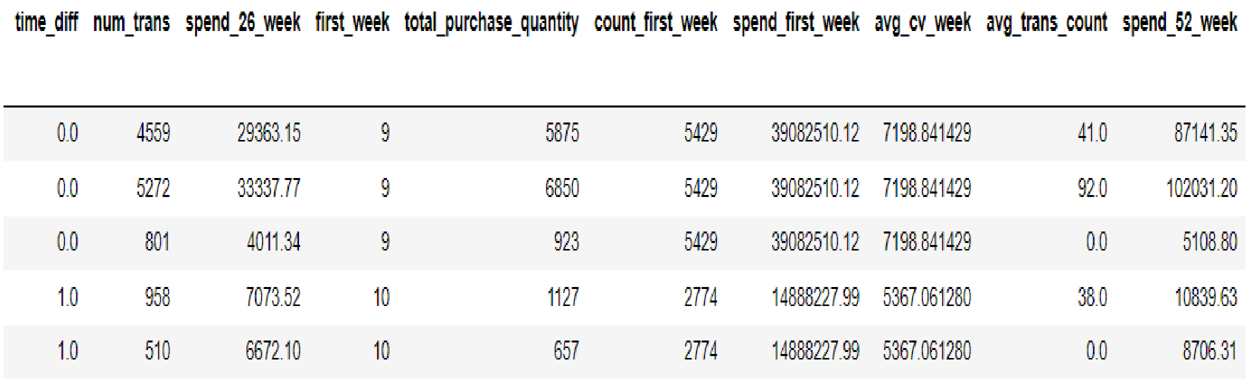


Fig. 4.7 first week count

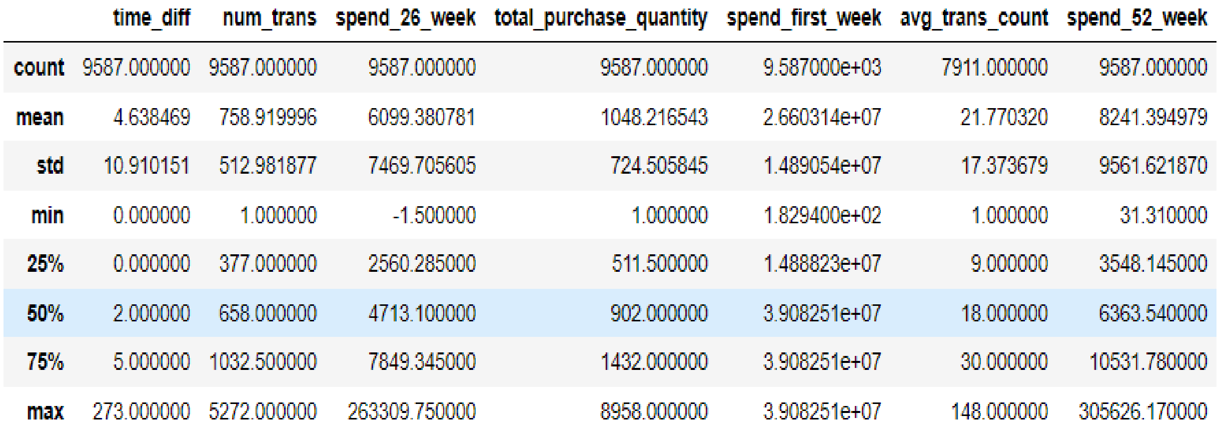
The first week count graph 4.7, it is evident that majority of the purchases start in week nine it is nearly 6000 and around 3000 purchases start in week ten. However only negligible number of transactions starts after week 11. From the past study on related paper, it is evident that the developed models were within 48 to 52 weeks of data to train. So that this research will use 26 weeks' worth of data to produce predictions. Therefore, the data is aggregated to first 26 weeks, which is exactly 182 days. Then make column according to each customer, one row containing useful features to predict 52-week value.



time_diff	num_trans	spend_26_week	first_week	total_purchase_quantity	count_first_week	spend_first_week	avg_cv_week	avg_trans_count	spend_52_week
0.0	4559	29363.15	9	5875	5429	39082510.12	7198.841429	41.0	87141.35
0.0	5272	33337.77	9	6850	5429	39082510.12	7198.841429	92.0	102031.20
0.0	801	4011.34	9	923	5429	39082510.12	7198.841429	0.0	5108.80
1.0	958	7073.52	10	1127	2774	14888227.99	5367.061280	38.0	10839.63
1.0	510	6672.10	10	657	2774	14888227.99	5367.061280	0.0	8706.31

Table 4-3: Final Data Frame used to build CLV modelling

The new features were engineered such as: Number of transactions, 26 weeks customer value, first week, Year, Total purchase quantity, 52 weeks customer value, time difference which means time difference between first and last transaction date, and so on. The new data frame is shown in the table 4.3.



	time_diff	num_trans	spend_26_week	total_purchase_quantity	spend_first_week	avg_trans_count	spend_52_week
count	9587.000000	9587.000000	9587.000000	9587.000000	9.587000e+03	7911.000000	9587.000000
mean	4.638469	758.919996	6099.380781	1048.216543	2.660314e+07	21.770320	8241.394979
std	10.910151	512.981877	7469.705605	724.505845	1.489054e+07	17.373679	9561.621870
min	0.000000	1.000000	-1.500000	1.000000	1.829400e+02	1.000000	31.310000
25%	0.000000	377.000000	2560.285000	511.500000	1.488823e+07	9.000000	3548.145000
50%	2.000000	658.000000	4713.100000	902.000000	3.908251e+07	18.000000	6363.540000
75%	5.000000	1032.500000	7849.345000	1432.000000	3.908251e+07	30.000000	10531.780000
max	273.000000	5272.000000	263309.750000	8958.000000	3.908251e+07	148.000000	305626.170000

Table 4-4: Summary statistics

The summary statistics of the final data frame is calculated using describe () function in pandas. It is summarised in table 4.4. Summary statistics shows all the statistics such as count, mean, std, minimum, maximum, first quarter, second quarter, and third quarter values, which is listed in the below table:

The summary statistics shows that the data frame has 9587 rows and mean of 26 weeks amount is 6099.38, whereas mean of 52 weeks amount is 8241.39. Other statistics can be seen from the above table.

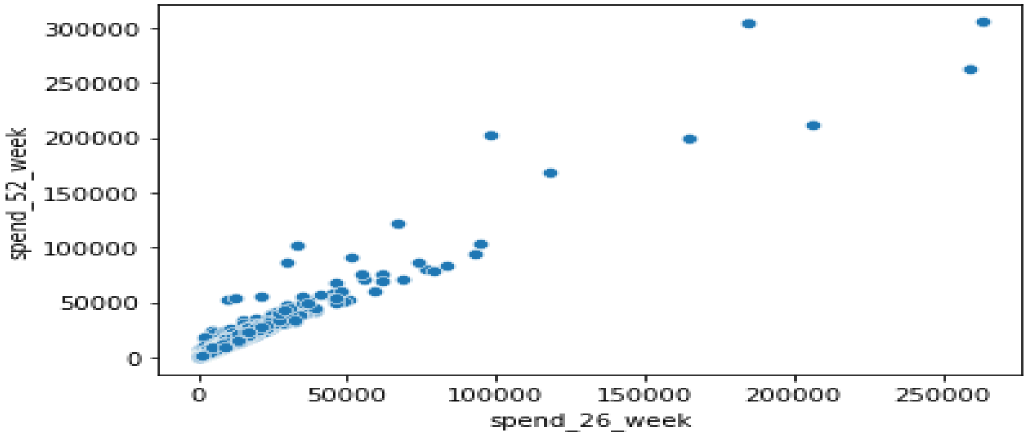


Fig.8

The scatterplot of spends on 52 weeks and 26 weeks, X axis shows the amount spend in 26 weeks and Y axis shows the amount spend in 52 weeks. It can be shown that the purchase amount after 26-week period has linear relationship between amount spend after 52 weeks. Or spend after both weeks has tight correlation between each other. The spread is wider in 0 to 50000 for both X axis and Y axis. It is evident that spend after 26 weeks and 52 weeks has clear linear relationship with each other.

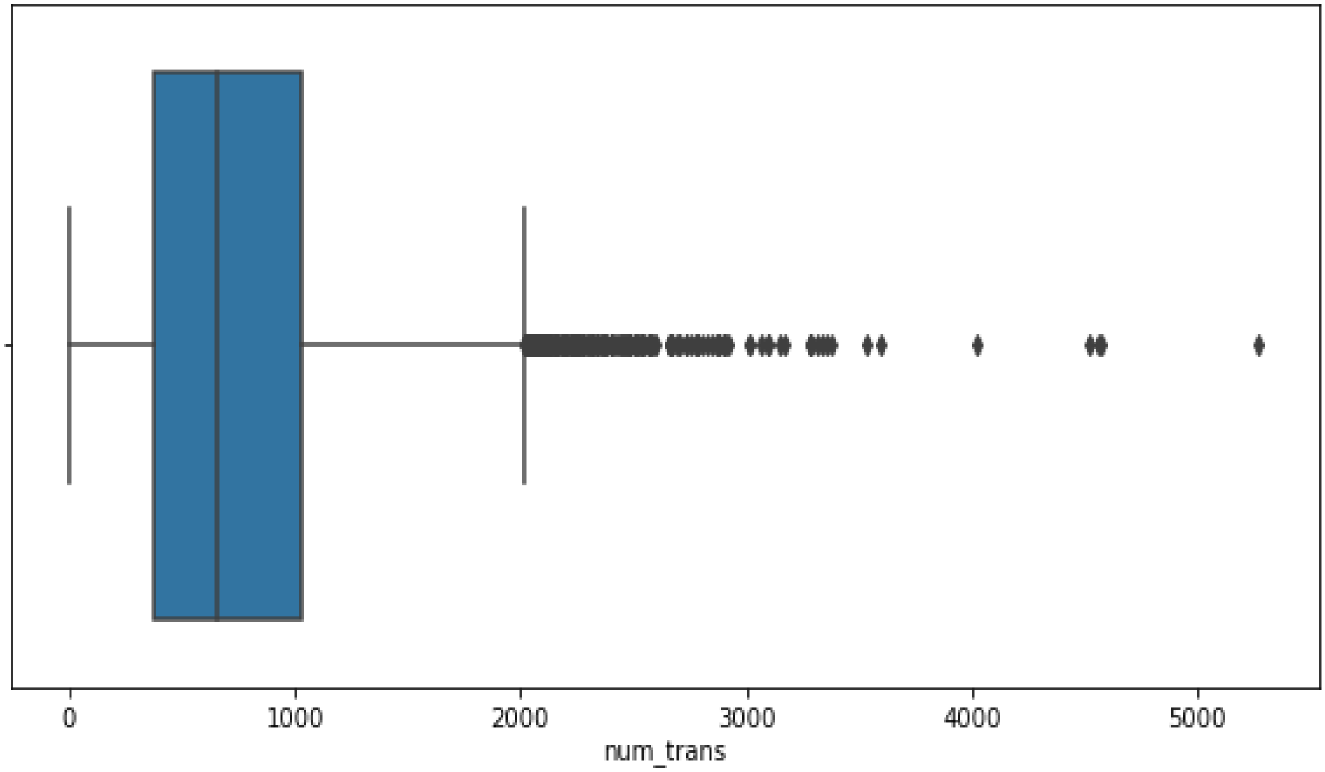


Fig. 4.9: Number of transactions in 26 weeks

The boxplot above shows the number of transactions in 26 weeks is distributed, the boxplot shows the summary statistics like minimum, maximum, median, skewness and outliers. The number of transactions in 26 weeks is left skewness. The median value of number of transactions is just above 500.

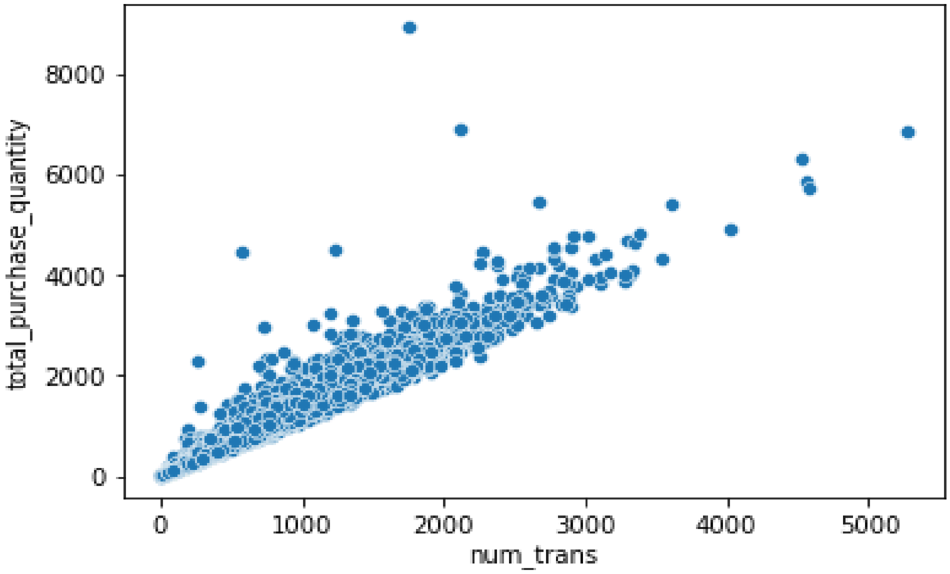


Fig. 4.10 Number of transaction vs total purchase quantity

The scatterplot of total purchase quantity versus number of transactions above in figure 4.12, we can read from the diagram that both features are highly correlated. The number of transactions increases the total purchase quantity also increases. But it could be a coincidence.

It concentrated around 500 to 2500 on X axis and 2000 on Y axis. The spread is wider on X axis than on Y axis.

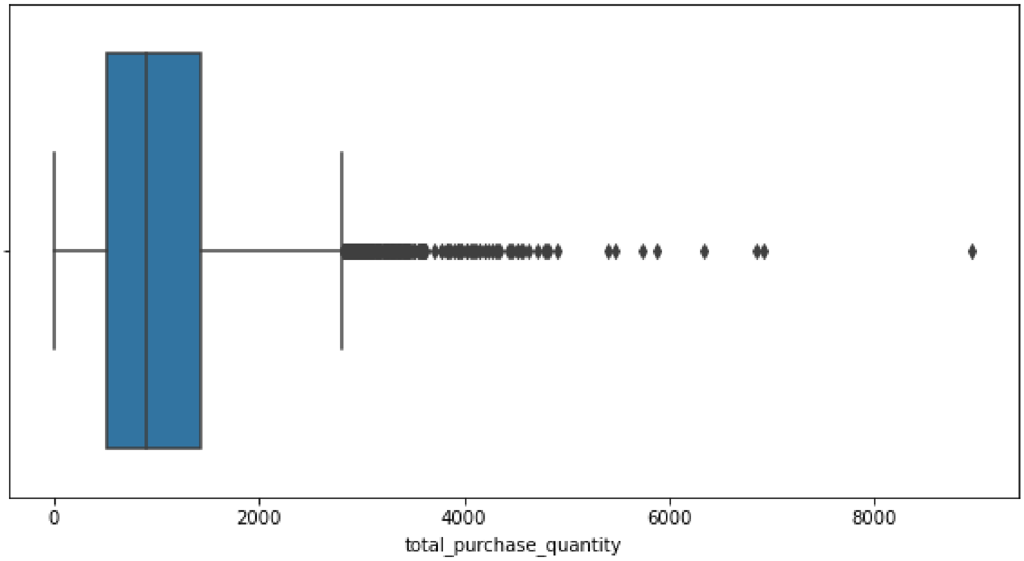


Fig. 4.11 box plot of total purchase quantity

The boxplot of total purchase quantity is shown below, total purchase quantity is the variable contain the total purchases of each customer. We can see that total purchase quantity is left skewed and the median is below 1000. The first quartile and third quartile ranges between around 500 to 1800.

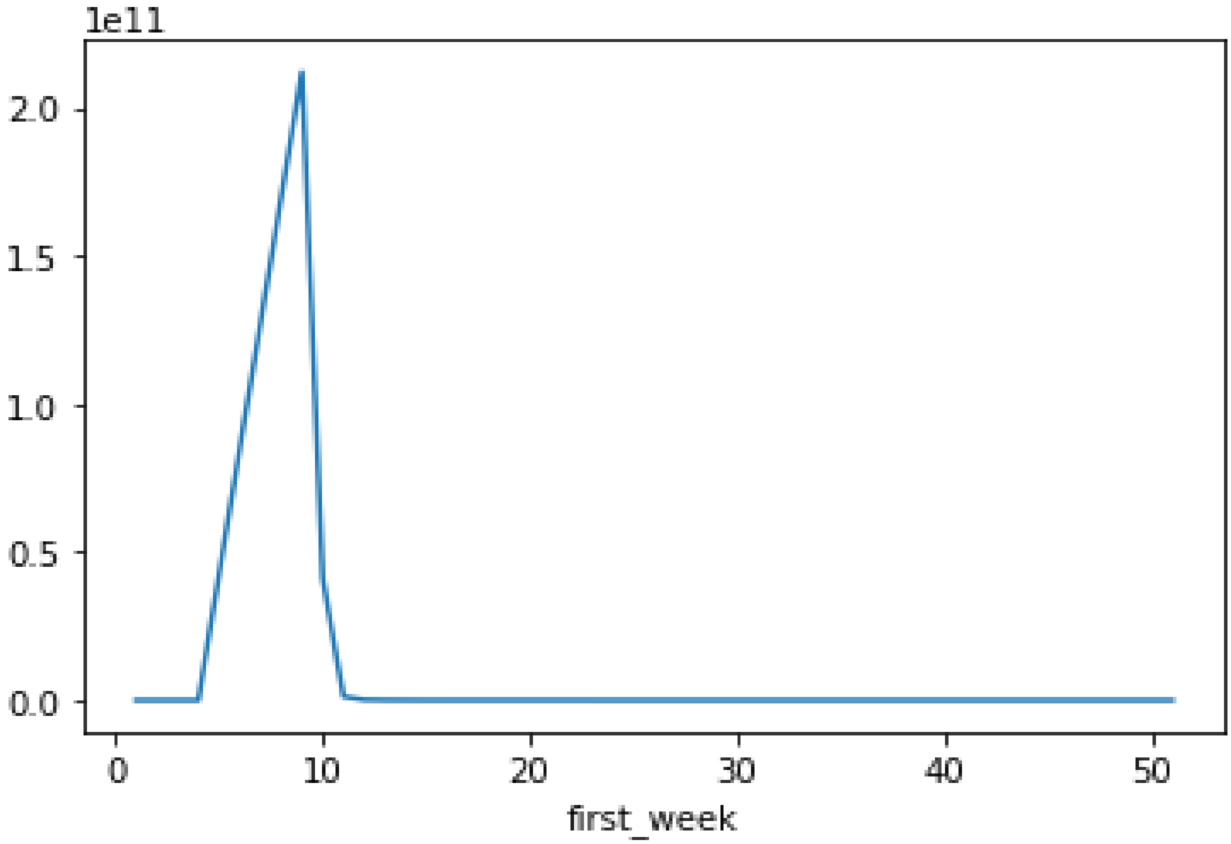
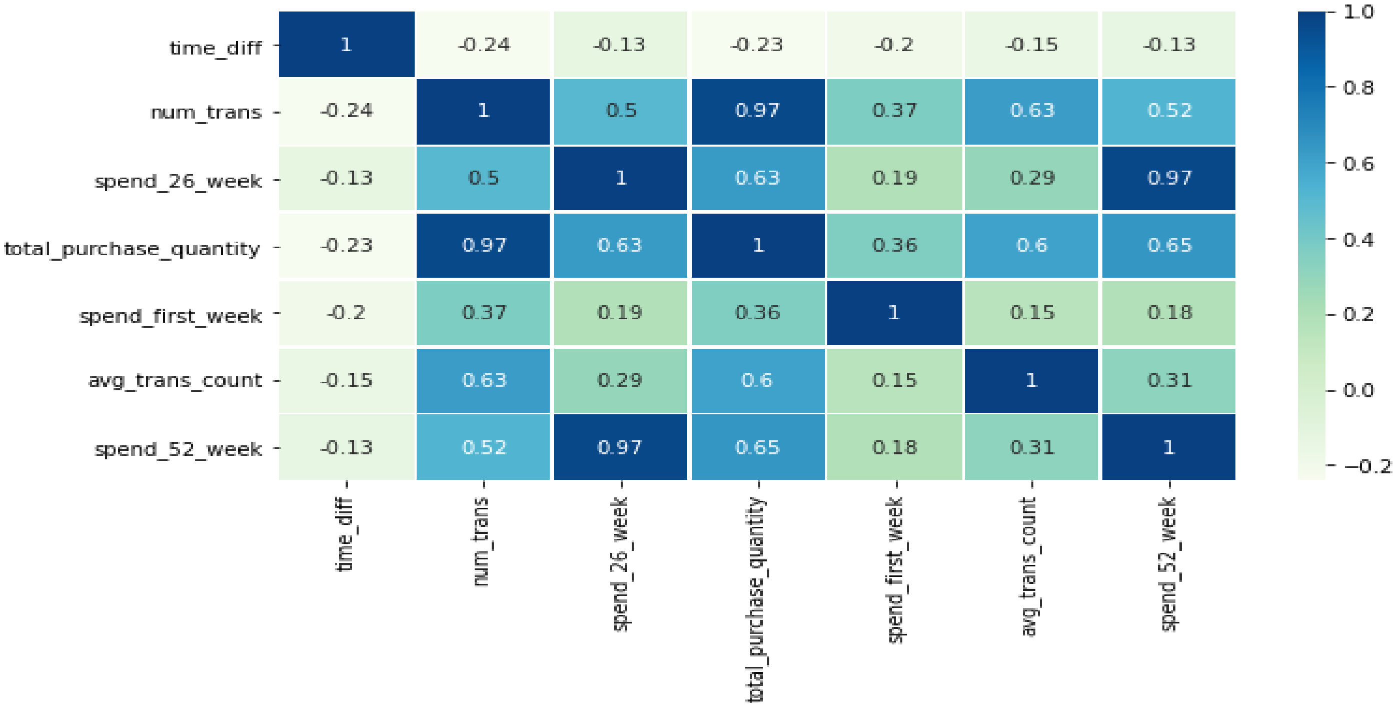


Fig. 4.12 sum of first week transactions

The line graph above figure 4.14 is the sum of spend first week versus first weeks of each customer's purchases. This graph provides a clear picture of the transaction amount of the first week of each customer. The graph says that majority of the purchases occurs in week 9, which is the peak point in the graph. Most of the purchases is in week 9 and 10. Only a few transactions are in the rest of the weeks.

The numerical variables in our dataset's correlations are in the correlation matrix below. This information is crucial since some Machine Learning algorithms difficulty with highly correlated input variables. First, calculate the correlation between each pair of characteristics using the corr() method on a DataFrame. Then pass the resultant correlation matrix to seaborn heatmap (), which creates a color-coded matrix for the input values:

The colour code in the matrix is from light green to blue. The light green represents the least correlation among the features. Whereas the blue represents the highly correlation among the features.



	time_diff	num_trans	spend_26_week	total_purchase_quantity	spend_first_week	avg_trans_count	spend_52_week
time_diff	1	-0.24	-0.13	-0.23	-0.2	-0.15	-0.13
num_trans	-0.24	1	0.5	0.97	0.37	0.63	0.52
spend_26_week	-0.13	0.5	1	0.63	0.19	0.29	0.97
total_purchase_quantity	-0.23	0.97	0.63	1	0.36	0.6	0.65
spend_first_week	-0.2	0.37	0.19	0.36	1	0.15	0.18
avg_trans_count	-0.15	0.63	0.29	0.6	0.15	1	0.31
spend_52_week	-0.13	0.52	0.97	0.65	0.18	0.31	1

Fig. 4.13 correlation heat map

From the matrix figure 4.15, it is clear that spend 52 week is highly correlated with spend 26 week and it is 0.97. Total purchase quantity and number of transactions are correlated with spend 52 weeks with 0.65 and 0.52 respectively. However, spend 52 week is least correlated with average transaction count. Number of transaction and total purchase quantity is also correlated with average transaction count. While average transaction count is least correlated with spend first week. The total purchase quantity is highly correlated with number of transactions. Therefore, from the correlation matrix it is evident that spend 52 week is highly correlated with spend 26-week, total purchase quantity and number of transactions. Spend 52 week has linear relationship with all these features.

4.5 Modelling

Before model building, we'll divide the data into two sets: one for developing models, pre-processing predictors, and exploring correlations between predictors and results (the training set), and another for determining the predictor set/model combination's performance (the test set). To partition the data, the original data set will be split in a stratified way, with random splits made in each of the outcome.

To understand model performance, dividing the dataset into a training set and a test set is a good strategy. Dataset was split into training and test datasets with nine sampling ratios (10:90, 20:80, 30:70, 40:60, 50:50, 60:40, 70:30, 80:20, and 90:10) to analyze the effect of sampling ratio on the performance of prediction algorithms and to establish the best sampling ratio for the dataset. The 70:30 sampling ratio was found to be the best suitable sample ratio for all algorithms when all other sampling ratios were considered. As a result of this finding, all susceptibility mapping models used training and test datasets with a 70:30 sample ratio (Sahin et al., 2020). The dataset was split into 70:30 ratio. Used 70% for training and 30% for testing the various models. Train/Test is a method to measure the accuracy of the model. Train the model means create the model. Test the model means test the accuracy of the model.

In total six regression model and one artificial neural network model were built for the dataset. That is:

- Linear Regression
- Lasso Regression
- Ridge Regression
- Random forest
- Decision tree

Each model was fitted by using the subset of data created as a training set.

The metric used to evaluate a model's way to estimate an outcome is critical and can influence the conclusions. The metric we use to assess model performance is determined by the outcome. When the result is a number, the Root Mean Squared Error is the most commonly used metric. A model is created to determine this number, and it is then used to forecast the outcome. The difference between the observed and expected outcome values is the residuals. To get the RMSE for a model, first compute the average of the squared residuals, then take the square root of that result. Taking the square root returns the metric to its original units of measurement. RMSE is defined as the average distance between a sample's observed and anticipated values. Simply defined, the smaller the RMSE, the more accurately a model can predict the outcomes of samples.

The equation to find RMSE is:RMSE = $\sqrt{\frac{\sum_{1}^{N}(y-\underline{y})^{2}}{N}}$

The coefficient of determination, often known as R2, is another significant statistic. There are numerous algorithms for calculating this number, but the simplest one squares the standard correlation between observed and expected values. The advantage of this statistic is that it has a simple interpretation for linear models: R2 is the proportion of total variability in the result that can be explained by the model. A score around 1.0 implies a nearly perfect fit, whereas a value near zero shows a model with no linear relationship between the predictions and the event. Another feature of this number is that it is unitless, making comparisons between various results simple.

Coefficient of Variation Root Mean Square error is the square of the distance between the predicted and the actual values. Adding all the squared residuals dividing by number of observations finally divide it by average dependent variable value. The statistical equation to find CV(RMSE) is:

$$\text{CV (RMSE)} = 1/y\sqrt{\frac{\sum_{1}^{N}(y-\underline{y})^{2}}{N}}$$

The CVRMSE has the advantage of providing a unitless percentage number that represents the model's efficiency. This metric was designed to eliminate the RMSE's dependence on capacity. For calibrated models created data, the CVRMSE value must be less than 30%. This measure should have a lower value (Chakraborty & Elzarka, 2018).

The score of CVRMSE, 25% indicates a good model fit with acceptable predictive capabilities.

Model	R-Square	CV(RMSE)
Linear regression	0.9592611619339252	0.21879088799777133
Lasso regression	0.9392045533043522	0.26727629274305603
Ridge regression	0.9592636035100083	0.21878433156871227
Random Forest regressor	0.9379998408056174	0.26991145366984853
Decision tree regressor	0.8494625940670322	0.42057855628977625

Table 4-5: R-Square and cvrmse of each Machine learning models

From the table above we can say that linear regression and ridge regression have score around 1.0 implies that both these models are perfectly fit and has a linear relationship between prediction and the event. From the table 2.1 the model accuracy of each model was above 80% and when compared to the regression models in this paper, the CV(RMSE) was found to be below 25% for both linear regression and ridge regression model implying that the models are reliably predictive. From the table it is clear that linear regression and Ridge regression has R-square nearly 96% and CV(RMSE) approximately 22% is the best performing model and is a good fit for predicting CLV.

4.5.1 Modelling using Linear regression

Regression analysis is a statistical method for determining the relationship between variables that have a causal relationship. The primary goal of univariate regression is to examine the relationship between a dependent variable and one independent variable and to create a linear relationship equation between them. Multilinear regression refers to regression models that have one dependent variable and many independent variables. Multilinear regression attempts to synchronically account for the change of the independent variable in the dependent variable. (Uyanık & Güler, 2013). The multivariate regression analysis model is formulated using:

$$\gamma = \beta 0 + \beta 1x1 +..+ \beta nxn + \varepsilon$$

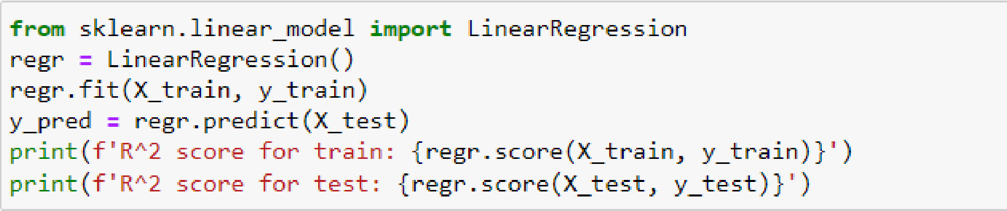
γ = dependent variable

x = independent variable

β = parameter

ε= error

Create a Linear Regression object after importing the Linear Regression module. Then, using the fit () method, fit the model to the train set and use the predict () function to predict on the test set.



```
from sklearn.linear_model import LinearRegression
regr = LinearRegression()
regr.fit(X_train, y_train)
y_pred = regr.predict(X_test)
print(f'R^2 score for train: {regr.score(X_train, y_train)}')
print(f'R^2 score for test: {regr.score(X_test, y_test)}')
```

Figure 4.16: Linear regression model development

4.5.2 Modelling using Lasso Regression

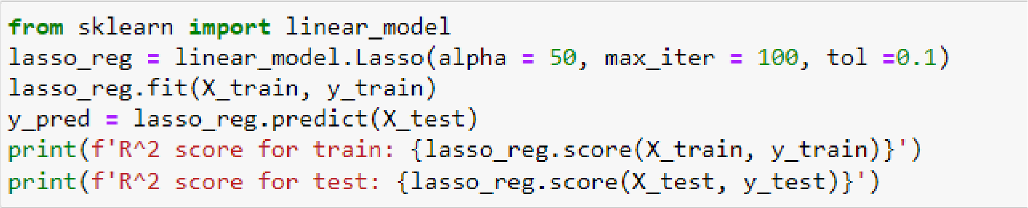
LASSO (Least Absolute Shrinkage and Selection Operator) regression is a popular approach for minimizing and selecting features. (Li et al., 2005) The Lasso regression loss function is defined as:

$$L = \Sigma i(yi - \Sigma p\ \beta pXip)\ \hat{}2 + \lambda\ \ \Sigma\ p\ ||\beta p||1$$

where Xip denotes the p th predictor (feature) in the I th datum, yi denotes the value of the response in this datum, and βp denotes the regression coefficient of the p th feature.

The goal of LASSO regression is to find the variables and regression coefficients that lead to a model with the least amount of prediction error. This is accomplished by placing a constraint on the model parameters that shrinks the regression coefficients towards zero, i.e., requiring the total of the absolute values of the regression coefficients to be smaller than a given value (λ).

Create a Lasso Regression object after importing the Linear module. Then, using the fit () method, fit the model to the train set and use the predict () function to predict on the test set.

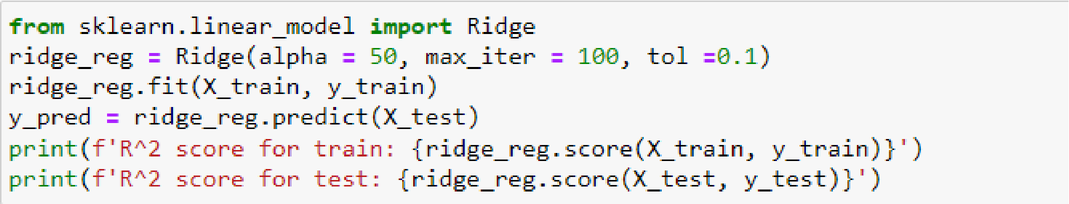


```
from sklearn import linear_model
lasso_reg = linear_model.Lasso(alpha = 50, max_iter = 100, tol =0.1)
lasso_reg.fit(X_train, y_train)
y_pred = lasso_reg.predict(X_test)
print(f'R^2 score for train: {lasso_reg.score(X_train, y_train)}')
print(f'R^2 score for test: {lasso_reg.score(X_test, y_test)}')
```

Figure 4.17: Lasso Regression model development

4.5.3 Modelling using Ridge Regression

Ridge regression is one of the most often used approaches for solving the multicollinearity problem. Due to their large variances, ordinary least squares unbiased estimators may become highly unstable in the presence of multicollinearity, resulting in poor prediction. Ridge regression, which modifies the least squares approach, enables biased estimators of regression coefficients to achieve a significant decrease in variance with an accompanying gain in stability of these values (Mardikyan et al., 2008).



```
from sklearn.linear_model import Ridge
ridge_reg = Ridge(alpha = 50, max_iter = 100, tol =0.1)
ridge_reg.fit(X_train, y_train)
y_pred = ridge_reg.predict(X_test)
print(f'R^2 score for train: {ridge_reg.score(X_train, y_train)}')
print(f'R^2 score for test: {ridge_reg.score(X_test, y_test)}')
```

Figure 4.18: Ridge Regression model Development

Create a Ridge Regression object after importing the Ridge module. Then, using the fit () method, fit the model to the train set and use the predict () function to predict on the test set.

4.5.4 Modelling using Random Forest Regression

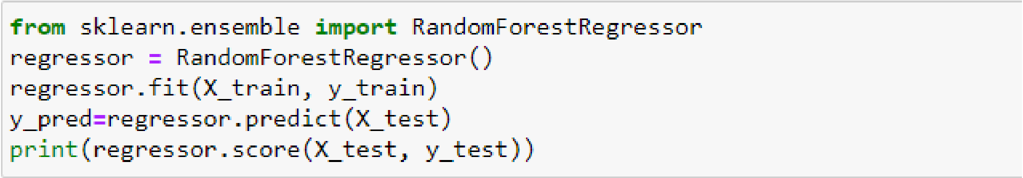
A random forest is a set of tree predictors h (x;θk),k=1…K where x represents the observed input (covariate) vector of length p with associated random vector X and the random vectors are independent and identically distributed. As we concentrate on regression problems with a numerical outcome, Y, however we would come across certain classification difficulties. The observed (training) data is considered to have been randomly selected from a joint distribution of (X, Y) and comprises (np+1)-tuples (x1, y1)….(xn,yn) (Segal, 2004).

The unweighted average over the collection is the random forest prediction for regression.

$h(x) = (1/K) \sum(hx;\theta k)$

when $k = \infty$ the Law of Large Numbers ensures $E X, Y = (Y \bar{h}(X)\text{^}2)$ -> $EX,Y (Y- E\theta (hX;\theta))\text{^}2$

The prediction error for the random forest is indicated by the quantity on the right. The above equation convergence implies that random forests do not overfit.



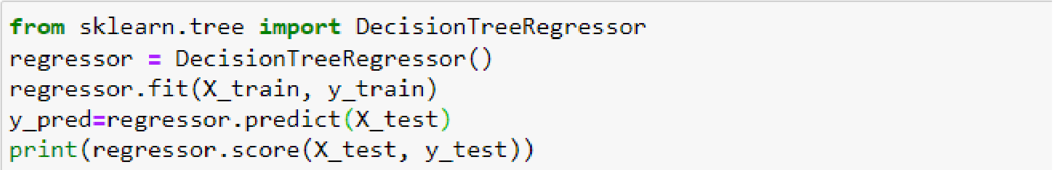
```
from sklearn.ensemble import RandomForestRegressor
regressor = RandomForestRegressor()
regressor.fit(X_train, y_train)
y_pred=regressor.predict(X_test)
print(regressor.score(X_test, y_test))
```

Fig. 4.14 Random Forest model development

Create a Random Forest object after importing the Random Forest Regressor module. Then, using the fit () method, fit the model to the train set and use the predict () function to predict on the test set.

4.5.5 Modelling using Decision Tree Regression

A decision tree's basic concept is to divide down a complex decision into multiple smaller ones, which may result in a solution that is easier to understand. The predictor variables in a decision tree method are data features, while the target variable is the class to be mapped. Decision tree classification is used when the target variable is discrete. Decision tree regression, on the other hand, is used when the target variables are continuous. Because the class proportions of a mixed pixel are measured on a continuous scale from 0 to 1, it makes sense to estimate them using decision tree regression. The underlying assumption in the decision tree regression approach is that correlations between characteristics and target objects are either linear or non-linear (Xu et al., 2005).



```
from sklearn.tree import DecisionTreeRegressor
regressor = DecisionTreeRegressor()
regressor.fit(X_train, y_train)
y_pred=regressor.predict(X_test)
print(regressor.score(X_test, y_test))
```

Fig. 4.15 Decision Tree Regressor Model Development

Create a Decision Tree object after importing the Decision Tree Regressor module. Then, using the fit () method, fit the model to the train set and use the predict () function to predict on the test set.

4.6 Evaluation

The following four error metrics are typically used to evaluate and report the performance of a regression model:

- RMSE – Root Mean Square Error
- MSE – Mean Square Error
- MAE – Mean Absolute Error
- CVRMSE - Coefficient of Variation of Root-Mean Squared Error

In this paper discuss about the R square, RMSE, MAE and CVRMSE of different model.

The table below shows the evaluation matric of machine learning models

MODEL	R SQUARE	MAE	RMSE	CVRMSE
Linear regression	R^2 score for train: 0.94 R^2 score for test: 0.95	821.67	1768.36	0.21
Lasso regression	R^2 score for train: 0.929 R^2 score for test: 0.93	1091.658	2160.241	0.26
Ridge regression	R^2 score for train: 0.94 R^2 score for test: 0.95	821.2	1768.30	0.21
Randomforest regressor	R^2 score for test:0.93	764.02	2181.54	0.26
Decision Tree regressor	R^2 score for test:0.84	1011.57	3399.29	0.42

Table 4-6: Evaluation matric of machine learing models

From the above table it can be concluded that the RMSE and MAE are least for linear regression model and ridge regression model. From R square we can interpret that around 96% of the model prediction are correct and the variation in the error is around 4 unit. Moreover, CVRMSE below 25% indicates a good model fit with acceptable predictive capabilities. When the CVRMSE is greater than 30%, the model predictive power is bad; when the CVRMSE is between 20% and 30%, it is simply fair; when the CVRMSE is between 10% and 20%, the predictive capability is good; and when the CVRMSE is less than 10%, the predictive power can be exceptional. In terms of predictability, the CVRMSE values of all univariate regression models are in the 10% to 20% range, indicating good predictability (Ge et al., 2018). Here linear regression model and ridge regression model has CVRMSE 21% is good fit for the model. From the table 2.1 it can be seen that model accuracy above 80% is a good model for prediction. Here both models have nearly 96% accuracy and the MAE for both regression models are nearly 822 is the least error and is showing excellent predictive capability.

The best model can be obtained after hyperparameter tuning, Data is used to learn model parameters, and hyper-parameters are tuned to achieve the best fit. Because finding the ideal hyper-parameter can be time-consuming, search algorithms such as grid search and random search are employed. Linear regression. get params (). keys (), is the method to find the parameter for tuning linear regression. Whereas Linear regression (). Set params () is the method to find how to set the parameters. Linear regression is the best model after fine tuning. From the RMSE: 1770.559655399599, MAE 824.9533028871023, R-squared: 0.959159842514317, CV_RMSE: 0.21906279069408421, we get the accurate model for predicting CLV.

For the better performance and accurate model fit the model was observed using different matric such as RMSE, MAE, CVRMSE the performance of the model was found using R square.

5. SYSTEM SPECIFICATIONS

5.1 HARDWARE REQUIREMENT

Processor	Core i3 or above
RAM	4GB or above
Hard Disk	500 GB or above
Mouse	Standard
Keyboard	Standard 104 enhanced keyboard

5.2 SOFTWARE REQUIREMENT

Language	Python
Operating system	Windows
Back-end	Python
Front-end	HTML, CSS
Tool	Python IDLE,VS Code
Web Browser	Google Chrome

6. Source code

6.1 Data Pre-processing

6.1.1 importing libraries

```
import os

import numpy as np

import pandas as pd

import tqdm

import datetime as dt

import seaborn as sns

import matplotlib.pyplot as plt

import scipy.stats as stats

from sklearn import metrics

from sklearn import model_selection

from sklearn import preprocessing

import warnings

warnings.filterwarnings("ignore", category=DeprecationWarning)
```

6.1.2 subset the large transaction data

```
#subset the 3GB data to 1000K

chunk_size=10000000

batch_no=1
```

```
for chunk in pd.read_csv('C:\\Users\\Desktop\\clv\\transactions.csv',chunksize=chunk_size):
chunk.to_csv('chunk'+str(batch_no)+'.csv',index=False)
batch_no+=1
```

6.1.3 read transactions sample data

```
df1 = pd.read_csv('C:\\Users\\Anjali\\OneDrive\\research\\clv\\chunk1.csv')
df1.shape
```

6.1.4 renaming attributes

```
df1.columns = ['id', 'chain', 'dept', 'category', 'company',
     'brand', 'date', 'product_size', 'product_measure',
     'purchase_quantity', 'purchase_amount']
df1.info()
```

6.1.5 creating year, month, week, day name from date column

```
df1['date']=pd.to_datetime(df1['date'])
df1['year'] = df1['date'].dt.year
df1['month'] = df1['date'].dt.month
df1['week'] = df1['date'].dt.isocalendar().week
df1['day_name'] = df1[['date']].apply(lambda x: dt.datetime.strftime(x['date'], '%A'), axis=1)
```

6.1.6 creating first and last transaction date

```
df2 = df1.sort_values(['id','date'])\

                                .reset_index(drop=True)

Grouped_data = df2.groupby(by='id')['date']\

                            .agg(['first','last'])        #first and last transaction date by customer id

Grouped_data[["first", "last"]] = Grouped_data[["first", "last"]].apply(pd.to_datetime)

Grouped_data['Days_gap']=(Grouped_data['last']-Grouped_data['first']).dt.days #get day gap- difference between first and last date by customer
```

6.1.7 splitting data to 26 week periods

```
#Time splitting

n_days = 182

max_date = df1['date'].max()

cut_off = max_date - pd.to_timedelta(n_days, unit = "d")   #split data into 26 weeks

temp_in = df1[df1['date'] <= cut_off]

temp_out = df1[df1['date'] > cut_off]

max_date = temp_in['date'].max() #create temporary dataframe by customer id

temp_df = temp_in[['id','date']].groupby('id').apply(lambda x:(max_date-x['date'].max())/pd.to_timedelta(1,"day")).to_frame().set_axis(['time_diff'],axis =1)
```

6.1.8 creating temporary dataframe for modelling

```
temp_df['num_trans']=temp_in[['id','date']].groupby('id').count() #get number of transaction of 26 weeks

temp_df['spend_26_week'] = temp_in.groupby('id').aggregate({'total_purchase': ('sum')}) #customer value of 26 weeks

temp = df1.groupby('id').aggregate({'date': ('min')}).set_axis(['min_date'],axis = 1)

temp_df['first_week']= temp['min_date'].dt.isocalendar().week #get first week of transaction by customer

temp_df['year']= temp['min_date'].dt.year          #det year of first week by customer

temp_df['total_purchase_quantity']= temp_in.groupby('id').aggregate({'purchase_quantity': ('sum')}) #get 26 weeks total purchase quantity by customer

temp_df['count_first_week']=temp_df.groupby(['first_week'])['first_week'].transform('count') #get first week count

temp_df['spend_first_week']=temp_df.groupby('first_week')['spend_26_week'].transform('sum') #get customer value of first week

temp_df['avg_cv_week']=temp_df['spend_first_week']/temp_df['count_first_week']        #average CV by first week

grouped=temp_in[['id','date','week']].groupby('id').apply(lambda g: g[temp_in['week'].min() == temp_in['week']])

temp_df['avg_trans_count']=grouped['date'].groupby('id').count()            #average transaction count by customer id

temp_df['spend_52_week']= df1.groupby('id')['total_purchase'].sum()            #customer value of 52 weeks
```

6.1.9 visualization

```
#most amount of purchases per customer

most_amount = df1.sort_values(by = ['total_purchase'], ascending = False).head()

most_amount[['id','total_purchase']]

#Total purchase by month

df1.groupby('month').sum()['total_purchase'].plot()

#Purchase quantity versus purchase amount

sns.scatterplot(df1["purchase_quantity"],df1["purchase_amount"])

plt.show()

# Transaction by day

plt.figure(figsize=(10,5))

sns.countplot(df1['day_name'])

plt.show()

# Transaction by Month

df1.groupby(by="id")["month"].value_counts()

df1['month'].value_counts().head(30).plot.bar(figsize=(12,5))

plt.ylabel('frequency')

plt.xlabel('month')

# First week Transaction

plt.figure(figsize=(10,5))

sns.countplot(temp_df['first_week'])

plt.show()
```

```
# Number of transaction in 26 weeks

plt.figure(figsize=(10,5))

sns.boxplot(temp_df['num_trans'])

plt.show()

#Total spend in 26 weeks

sns.distplot(temp_df["spend_26_week"])

plt.show()

#Total spend in 52 weeks

sns.distplot(temp_df["spend_52_week"])

plt.show()

# Number of transaction vs total purchase quantity

sns.scatterplot(temp_df["num_trans"],temp_df["total_purchase_quantity"])

plt.show()
```

6.1.10 Correlation heat map

```
plt.figure(figsize=(10,5))

sns.heatmap(temp_df.drop(columns=['first_week','year','count_first_week','avg_cv_week']).corr(), annot = True, linewidths=.5, cmap="GnBu")

plt.show()
```

6.2 Modeling

6.2.1 importing libraries

```
import os

import numpy as np

import pandas as pd

import tqdm

import datetime as dt

import seaborn as sns

import matplotlib.pyplot as plt

import scipy.stats as stats

from sklearn import metrics

from sklearn import model_selection

from sklearn import preprocessing

import warnings

warnings.filterwarnings("ignore", category=DeprecationWarning)
```

6.2.2 reading csv for modeling

```
dataset_1 = pd.read_csv('final_data4.csv') # read final dataset for modelling

dataset_1.isnull().sum()                #check missing values

dataset_1 = dataset_1.fillna(0)          #fill null values with zero

dataset_1.shape
```

6.2.3 separating input and output features

```
# Separate input features and output feature

X = dataset_1.drop(columns = ['spend_52_week','first_week', 'year']) # input features

y = dataset_1.spend_52_week                                   # output feature

from sklearn.model_selection import train_test_split

X_train,X_test, y_train, y_test = train_test_split(X, y, test_size=0.30,  random_state=42)   #split to train and test data
```

6.3 Modeling

6.3.1 Linear regression

```
from sklearn.linear_model import LinearRegression

regr = LinearRegression()                                 #linear regression object

regr.fit(X_train, y_train)

y_pred = regr.predict(X_test)                             #predict on test set

print(f'R^2 score for train: {regr.score(X_train, y_train)}')      #r-square for train

print(f'R^2 score for test: {regr.score(X_test, y_test)}')         #r-square for test
```

6.3.1.1 Evaluation of linear regression

```
from sklearn.metrics import mean_squared_error,r2_score,mean_absolute_error
```

```
y_pred = regr.predict(X_test)

print('RMSE:',np.sqrt(mean_squared_error(y_test,y_pred)))        #RMSE

print("MAE",mean_absolute_error(y_test,y_pred))                  #MAE

print('R-squared:',r2_score(y_test,y_pred))                     #R-SQUARE

rmse=np.sqrt(mean_squared_error(y_test,y_pred))

y_mean= y_test.mean()

print('CV_RMSE:',rmse/y_mean)                                   #CVRMSE
```

6.3.2 Lasso Regression

```
from sklearn import linear_model

lasso_reg = linear_model.Lasso(alpha = 50, max_iter = 100, tol =0.1)    #lasso regression object

lasso_reg.fit(X_train, y_train)

y_pred = lasso_reg.predict(X_test)

print(f'R^2 score for train: {lasso_reg.score(X_train, y_train)}')      #r-square for train

print(f'R^2 score for test: {lasso_reg.score(X_test, y_test)}')        #r-square for test
```

6.3.2.1 Evaluation of lasso regression

```
from sklearn.metrics import mean_squared_error,r2_score,mean_absolute_error

y_pred = lasso_reg.predict(X_test)

print('RMSE:',np.sqrt(mean_squared_error(y_test,y_pred)))             #RMSE

print("MAE",mean_absolute_error(y_test,y_pred))                       #MAE
print('R-squared:',r2_score(y_test,y_pred))                    #R-SQUARE

rmse=np.sqrt(mean_squared_error(y_test,y_pred))

y_mean= y_test.mean()

print('CV_RMSE:',rmse/y_mean)                                  #CVRMSE
```

6.3.3 Ridge Regression

```
from sklearn.linear_model import Ridge

ridge_reg = Ridge(alpha = 50, max_iter = 100, tol =0.1)          #Ridge regression object

ridge_reg.fit(X_train, y_train)

y_pred = ridge_reg.predict(X_test)

print(f'R^2 score for train: {ridge_reg.score(X_train, y_train)}')      #r-square for train

print(f'R^2 score for test: {ridge_reg.score(X_test, y_test)}')         #r-square for test
```

6.3.3.1 Evaluation of ridge regression

```
from sklearn.metrics import mean_squared_error,r2_score,mean_absolute_error

y_pred = ridge_reg.predict(X_test)

print('RMSE:',np.sqrt(mean_squared_error(y_test,y_pred)))          #RMSE

print("MAE",mean_absolute_error(y_test,y_pred))                    #MAE

print('R-squared:',r2_score(y_test,y_pred))                        #R-SQUARE

rmse=np.sqrt(mean_squared_error(y_test,y_pred))

y_mean= y_test.mean()
```

```
print('CV_RMSE:',rmse/y_mean)                         #CVRMSE
```

6.3.4 Random forest regression

```
from sklearn.ensemble import RandomForestRegressor

regressor = RandomForestRegressor()             #randomforest object

regressor.fit(X_train, y_train)

y_pred=regressor.predict(X_test)

print(regressor.score(X_test, y_test))          #r-square for test
```

6.3.4.1 Evaluation of random forest regression

```
y_pred=regressor.predict(X_test)

print('RMSE:',np.sqrt(mean_squared_error(y_test,y_pred)))  #RMSE

print("MAE",mean_absolute_error(y_test,y_pred))        #MAE

print('R-squared:',r2_score(y_test,y_pred))            #R-SQUARE

rmse=np.sqrt(mean_squared_error(y_test,y_pred))

y_mean= y_test.mean()

print('CV_RMSE:',rmse/y_mean)                     #CVRMSE
```

6.3.5 Decision tree regression

```
from sklearn.tree import DecisionTreeRegressor

regressor = DecisionTreeRegressor()             #decision tree object
```

```
regressor.fit(X_train, y_train)

y_pred=regressor.predict(X_test)

print(regressor.score(X_test, y_test))          #r-square for test
```

6.3.5.1 Evaluation of decision tree regression

```
y_pred=regressor.predict(X_test)

print('RMSE:',np.sqrt(mean_squared_error(y_test,y_pred))) #RMSE

print("MAE",mean_absolute_error(y_test,y_pred))       #MAE

print('R-squared:',r2_score(y_test,y_pred))          #R-SQUARE

rmse=np.sqrt(mean_squared_error(y_test,y_pred))

y_mean= y_test.mean()

print('CV_RMSE:',rmse/y_mean)                      #CVRMSE
```

7. Conclusion

This study provides a test of different supervised machine learning algorithms to see how well they perform. On the final datasets, these machine learning models were utilized. The initial part of this thesis was a literature review to summarize existing knowledge on CLV prediction in Data Analytics, as well as the tools and methodologies employed in this work, such as CRISP-DM, Machine Learning, and other regression models.

The null values in the data have been replaced with zero, and the missing values have been eliminated.

The dataset was divided into two groups: one for training the dataset (70%) and another for testing the dataset (30%).

After that, each of the supervised regression models: linear regression, lasso regression, ridge regression, random forest, decision tree, adaboost, and ANN were used to train the dataset. Each model may forecast accuracy using metrics, indicating which model outperformed the others or was the best fit for the task. The ability of linear regression, lasso regression, ridge regression, random forest, decision tree, adaboost, and ANN modelling techniques to predict CLV performance from past transaction data is demonstrated in this study.

This project shows that we can accurately predict CLV utilizing a 26-week period to predict 52-week customer value. And there is a clear linear relationship between 26 weeks value and 52 weeks value. All of the modelling results suggest that this approach is correct.

In general, a customer with a 26-week purchase history will make another purchase in 52 weeks or less than a year, indicating that they are loyal to the company. This is also a linear relationship

8. Future scope

Although this project focus on four regression models, the performance of models might have been examined by using more data and training model performance on a large dataset.

We implemented a Holdout approach to train the regression data models, in which 70% of the data was used for training and 30% for testing. Instead, a different split (such as 80/20 or 90/10) or even a test of all of them may have been used.

More regression models, a larger dataset, and a larger number of records, such as offers, should all be added in future study to evaluate how they affect CLV prediction.

9. References

- customer lifetime value of credit card customers. *Journal of Database Marketing & Customer Strategy Management*.
- Almuqren, L., Alrayes, F.S. & Cristea, A.I. 2021. An empirical study on customer churn behaviours prediction using arabic twitter mining approach. *Future Internet*.
- Coussement, K. & de Bock, K.W. 2018. A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *European Journal of Operational Research*, 269(2).
- de Caigny, A., Coussement, K., de Bock, K.W. & Lessmann, S. 2020. Incorporating textual information in customer churn prediction models based on a convolutional neural network. *International Journal of Forecasting*, 36(4): 1563–1578.
- Çavdar, A.B. & Ferhatosmanoğlu, N. 2018. Airline customer lifetime value estimation using data analytics supported by social network information. *Journal of Air Transport Management*, 67.
- Cenggoro, T.W., Wirastari, R.A., Rudianto, E., Mohadi, M.I., Ratj, D. & Pardamean, B. 2021. Deep Learning as a Vector Embedding Model for Customer Churn. In *Procedia Computer Science*. Elsevier B.V.: 624–631.
- Chakraborty, D. & Elzarka, H. 2018. Performance testing of energy models: are we using the right statistical metrics? *Journal of Building Performance Simulation*, 11(4).
- Chamberlain, B.P., Cardoso, Â., Bryan Liu, C.H., Pagliari, R. & Deisenroth, M.P. 2017. Customer lifetime value prediction using embeddings. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.
- Chen, P.P., Guitart, A., del Río, A.F. & Periáñez, A. 2019a. Customer Lifetime Value in Video Games Using Deep Learning and Parametric Models. In *Proceedings - 2018 IEEE Internnce on Big Data, Big Data 2018*. Institute of Electrical and Electronics Engineers Inc.: 2134–2140.
- Chen, P.P., Guitart, A., del Río, A.F. & Periáñez, A. 2019b. Customer Lifetime Value in Video Games Using Deep Learning and Parametric Models. In *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*. Institute of Electrical and Electronics Engineers Inc.: 2134–2140.
- Chen, S. 2018. Estimating Customer Lifetime Value Using Machine Learning Techniques. In *Data Mining*.

- Cheng, C.H. & Chen, Y.S. 2009. Classifying the segmentation of customer value via RFM model and RS theory. *Expert Systems with Applications*, 36(3 PART 1).
- Dachyar, M., Esperanca, F.M. & Nurcahyo, R. 2019. Loyalty Improvement of Indonesian Local Brand Fashion Customer Based on Customer Lifetime Value (CLV) Segmentation. In *IOP Conference Series: Materials Science and Engineering*.
- Dingli, A., Marmara, V. & Fournier, N.S. 2017. Comparison of deep learning algorithms to predict customer churn within a local retail industry. *International Journal of Machine Learning and Computing*, 7(5).
- Donkers, B., Verhoef, P.C. & de Jong, M.G. 2007. Modeling CLV: A test of competing models in the insurance industry. *Quantitative Marketing and Economics*, 5(2).
- Ekinci, Y., Uray, N. & Ülengin, F. 2014. A customer lifetime value model for the banking industry: a guide to marketing actions. *European Journal of Marketing*, 48(3/4).