CLV

Mehdi Farhangian

12/21/2019

The best marketing strategy is to maintain customers who are profitable and remain loyal for a Lifetime. In order to obtain and maintain these customers, we need to measure an important metric that is called customer lifetime value. Calculating lifetime value has had lots of applications such as performance measurement, targeting customers, marketing resources allocation, product offering, pricing, and customer segmentation.

However, as we explained in our previous article, it is easier said than done and almost everyone is doing it wrong. In order to measure calculate CLV, Gupta et al. described six modelling approaches:

- RFM Models: Which are models based on Recency, Frequency, and Monetary.
- Probability Models: Which are models based on Pareto/NBD model and Markov chains.
- Econometric Models: Which are models based on Pareto/NBD in conjunction with some factors such as customer acquisition, customer * * * retention, and customer margin and expansion.
- Persistence Models: Which are models based on modelling the behaviour of its components Which are acquisition, retention, and cross-selling.
- Computer Science Models: Which are models based on some theories such as utility theory.
- Diffusion/Growth Models: Which are models based on customer equity (CE).

RFM has been used in marketing to predict customer behaviour for more than 50 years and is widely known as one of the most powerful techniques in marketing. In this article,] provide an extended framework in which customers can be clustered in RFM segments, and customers' lifetime value is calculated based on these clusters.

##Step 1: Data preprocessing

Understanding and preparation of data is the most important and time-consuming part of customers' lifetime value modelling. Usually, in our databases, we have data about the purchase behaviour of our customers, as well as sociodemographic data (e.g. age, gender, etc). The purchase behaviour is the main indicator of the lifetime value of customers and can be used to predict the customers' behaviour in the future. One of the advantages of RFM compared to other methods is its availability. A firm can use RFM analysis immediately and without requiring any additional data, as it has a database of its customers containing their purchase history.

Through using the example below, we explain how customer lifetime value can be calculated. Firstly, let's generate a random dataset in R that contains customers' purchase history as follows.

```
N <- 1000
df <- data.frame(CustomerID = sample(1:500, size = N, replace = TRUE),
Date = format(seq(as.Date("2015/01/01"), by='day', length = N, replace = TRUE)),
Revenue = sample(100:500, size = N, replace = TRUE))
```

##Step 2 RFM Method

The most popular approach for measuring the customer's' lifetime value is RFM. RFM refers to a modelling technique that uses the following three factors from client records:

Recency: Period since last purchase. Frequency: How many purchases an individual made during the observation period. Monetary: Cumulative total spent by client during observation period. Bruce Hardie developed a spreadsheet to implement RFM that is easy to use and can be seen here

Grouping the RFM can be done through clustering algorithms which are a form of unsupervised machine learning. A popular method for clustering is to use Hartigan's Rule which "essentially compares the ratio of the within-cluster sum of squares for a clustering with k clusters and one with k + 1 clusters, accounting for

	Method		
	Pareto/NBD (P/NBD)	Beta geometric/NBD (BG/NBD)	Beta geometric/beta binomial (BG/BB)
	Schmittlein et al. (1987)	Fader et al. (2005)	Fader et al. (2004)
Phenomena			
Customer alive	$\tau = \text{lifetime} \sim \text{Exp}(\mu)$	$p = \text{probability "die" after each purchase}$	$q = \text{probability "die" after each purchase}$
Heterogeneity	$\mu \sim \text{gamma}(s, \beta)$	$p \sim \text{Beta}(a, b)$	$q \sim \text{Beta}(\gamma, \delta)$
Purchase rate	$x = \text{number of purchases in time } t \sim \text{Poisson}(\lambda)$	$x = \text{number of purchases in time } t \sim \text{Poisson}(\lambda)$	$p = \text{probability of purchase in each period}$
Heterogeneity	$\lambda \sim \text{gamma}(r, \alpha)$	$\lambda \sim \text{gamma}(r, \alpha)$	$p \sim \text{Beta}(\alpha, \beta)$
Summary			
Lifetimes	Exponential	Geometric	Geometric
Purchases	Poisson	Poisson	Bernoulli
Interpurchase	Exponential	Exponential	Geometric

Figure 1: A caption

the number of rows and clusters. If that number is greater than 10, then it is worth using k + 1 clusters." . In the example below, we have two columns of frequency and recency of my customers and wish to cluster them based on these two factors by using R:

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
R3 <- group_by (df, CustomerID) %>% summarise( Monetary= sum(Revenue), Frequency = n(),
Recency = 1/(as.numeric (difftime(format(Sys.time(), "%Y-%m-%d"), last(Date),units = c("days")) )))
```

After data preparation, we need to score our RFM variables. In this article, we assume all three variables have the same weight. It should be noted that the weight of RFM variables vary depending on industry characteristics. Furthering this we use the Min-max normalisation method to convert all the data, to a date range between 1 to 5. This method performs a linear transformation on the original data and can be calculated using the following formula:

Distribution based Non-Contractual Models

An alternative to RFM is to use a more complicated approach which is the distributed based non-contractual model. Three stochastic models are the most popular ones for calculating CLV, namely BG/NBD, BG/BB, Pareto/NBD. A comparative summary of these models is in the following table:

Bruce Hardie has a number of excel spreadsheets and explanations for these models:

- BG/NBD Model

- BG/BB Model

- Pareto/NBD Model

An R package called BTYD can be used to calculate these models.

In addition, Lifetimes is a Python library to calculate CLV. We use an example to show how this package can be used for measuring CLV. In this example you can use cdnow_customers.csv located in the datasets/ directory.

We need to convert our dataset into a dataset that contains Recency, Frequency and Monetary columns.

```r
library(dplyr)
R3 <- group_by (df, CustomerID)%>% summarise( Monetary= sum(Revenue), Frequency = n(),
Recency = 1/(as.numeric (difftime(format(Sys.time(), "%Y-%m-%d"), last(Date),units = c("days")) )))
```

After data preparation, we need to score our RFM variables. In this article, we assume all three variables have the same weight. It should be noted that the weight of RFM variables vary depending on industry characteristics. Furthering this we use the Min-max normalisation method to convert all the data, to a date range between 1 to 5. This method performs a linear transformation on the original data and can be calculated using the following formula:

We normalise our dataset in R as follows:

```r
normalize <- function(x) {
return ((((x - min(x)) / (max(x) - min(x))) * 5)}

dfNorm <- as.data.frame(lapply(R3[2:4], normalize))
```

##Step 3: Clustering

Clustering is the process of grouping a set of physical or abstract objects into similar groups. In this article after preprocessing and normalizing the data, we want to group customers with a similar purchase behaviour So knowing this, clustering techniques are employed to cluster customers according to RFM values. Other customers with similar lifetime values should be grouped together. It should be noted that the results produced from clustering, should be constantly evaluated. Additionally when performing a cluster, make sure that clusters are significantly different from each other, and are also easily interpretable. We can use different algorithms such as K-means and the Two-step algorithm to compare the accuracy of the results with each other. However, the mathematics behind clustering algorithm is not in the scope of this article.

Number of Clusters

The first step in clustering is determining the optimal number of clusters for a given dataset. There are multiple methods that can be used to determine the optimal segments of a given dataset. This Wikipedia article has a review of some of them.

The elbow method is one of the most popular methods for finding the optimal number of clusters, that look at the percentage of variance, explained as a function of the number of clusters. It can be computed using the following function and then plotted for our dataset.

```r
wss <- (nrow(dfNorm)-1)*sum(apply(dfNorm,2,var))
for (i in 2:15) wss[i] <- sum(kmeans(dfNorm,centers=i)$withinss)
plot(1:15, wss, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares")
```
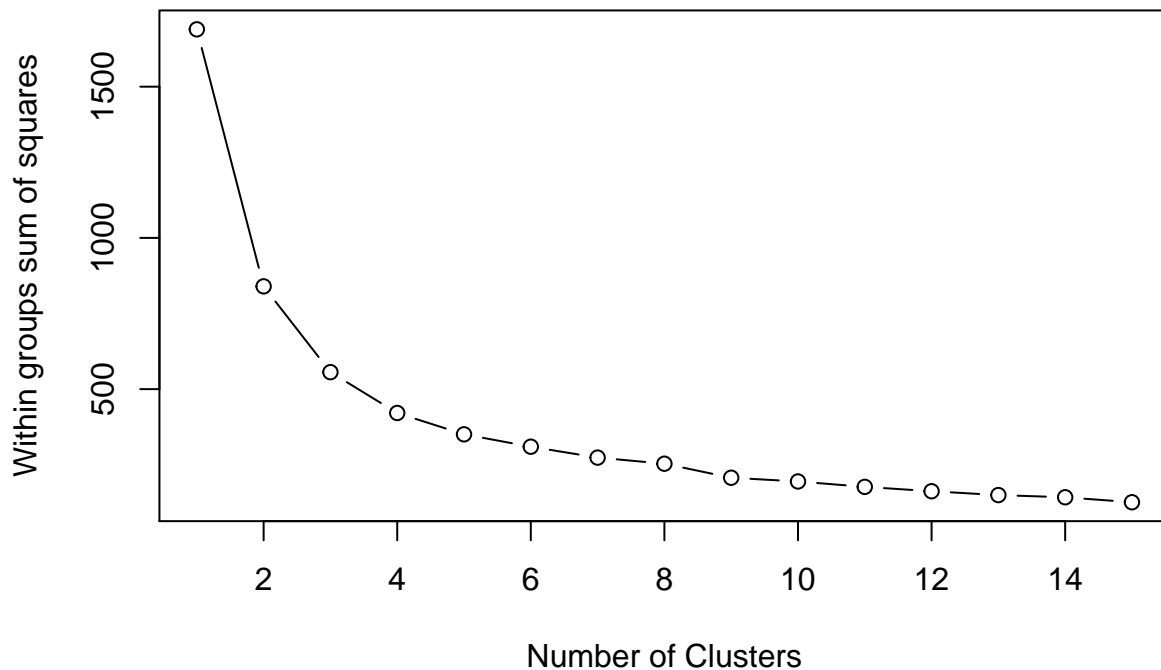
In this example, we see an 'elbow' at 5 clusters. By adding more clusters than that we get relatively smaller gains. So 5 clusters are chosen for this dataset.

After finding the optimal number of clusters, we use the k-means method the cluster our dataset:

```r
mydataCluster <- kmeans(dfNorm, 5, nstart = 20)
mydataCluster$cluster <- as.factor(mydataCluster$cluster)
```

Visualize the clusters

Visualising the cluster provides more interpretability for a business. We can plot the results by selecting two of the variables as follows:

```r
library (ggplot2)
ggplot(dfNorm, aes(Recency, Monetary, color = mydataCluster$cluster)) + geom_point()
```
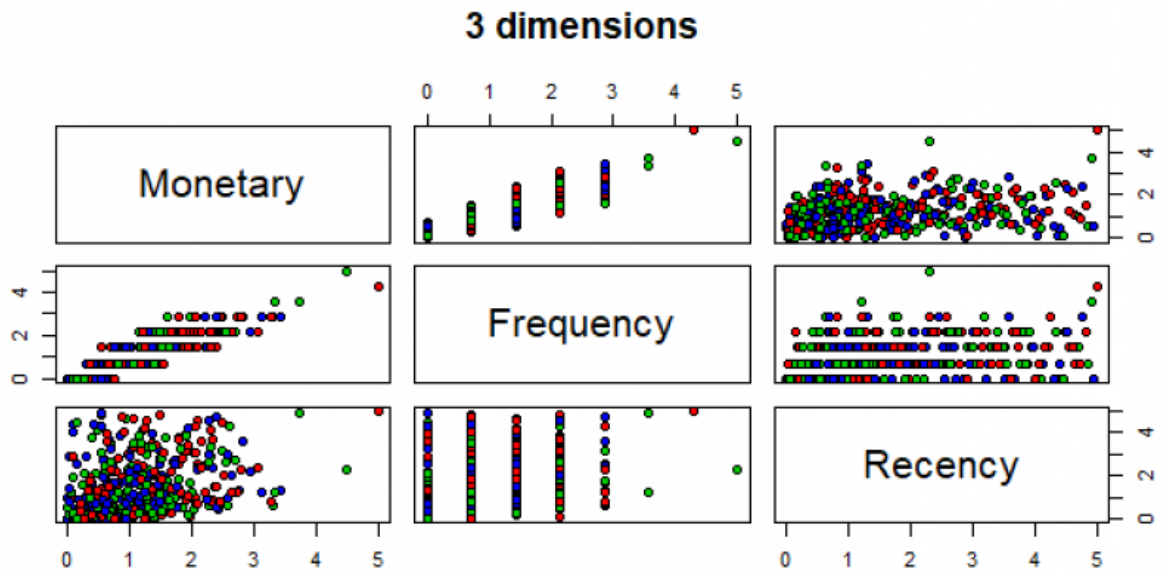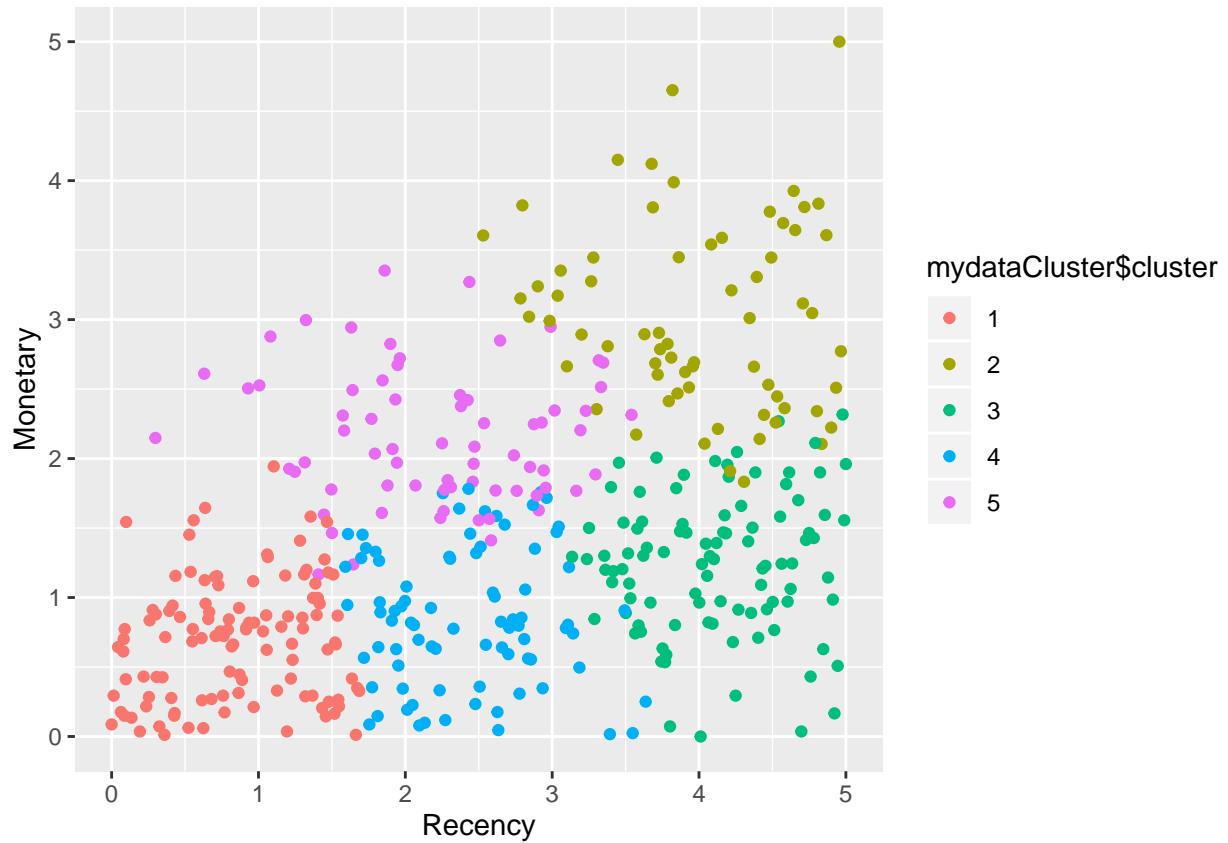
Figure 2: A caption

In order to visualise more than two variables, we can use various techniques. We believe the best approach is producing a matrix of scatterplots using the pairs function in R.

Step 4: Generate rules

This step is extremely valuable and the rules provided can play an important role in our strategies. However this step is not necessary, and if you are merely interested in calculating the lifetime value of customers, this step can be skipped and you can go to step 5.

In order to provide more insights, we add a column that contains the number of clusters in our dataset.

```
out <- cbind(dfNorm, clusterNum = mydataCluster$cluster)
```

We also added more variables to our dataset such as sociodemographic data (e.g. age, gender, etc.) and also online behaviour data such as (time on site, number of events and so on).

After adding variables and labelling the number of clusters to each customer, we want to mine some rules in our dataset to see the relationship between the selected variables, with the cluster of customers. Some techniques such as C.50 can be used for producing decision trees and rules.

In this article, we use association rule mining and Apriori algorithm to mine rules in our dataset. The following three terms help you to have a better understanding of association rule mining:

- Support= The indication of how frequently a variable appears in a dataset.
- Confidence = The indication of how often the rule has been found to be true
- Lift = The performance of a rule. A higher lift means a stronger rule.

The following code is used in this example to produce the rules

Applying this model to our dataset resulted in the following rules:

- Gender = female, age = 20-30, time on site > 50 then class = 5
- Gender = male, age >40 and time on site = 300-500, then class =5

We can generate more rules by changing the support, confidence level, and also the variables in the appearance list.

Step 5: Calculate the net present value per cluster Lastly, we measure the lifetime value of each cluster. A basic formula is used for Lifetime value in this article as follows:

$LTV = CM_i(\frac{R}{1+d-R}) - AC_i$

Where

- CM= customers' margin for each cluster
- R = retention rate of customers for each cluster
- d = discount rate
- AC= customer acquisition cost for each cluster

We measure the lifetime value for each cluster separately.

Customer Retention rate can be calculated as follows

$AC = \frac{CE-CN}{CS} * 100$

- CE = Number of customers at the end of the period for each cluster
- CN = Number of customers acquired during the period for each cluster
- CS = Number of customers at the start of the period for each cluster