Estimating Customer Lifetime Value in the Gaming Industry Using Incomplete Data

R. Ali [1]†, S. Abrahams [2] A. Berryman [2] C. Bleak [3] N.A. Hamzah [4] T.F. Khang [4] P.G. Hjorth [5] C.M. Ng [4] Y. Tian [2] J.A. Ward [1] and H. Yang [2]

[1] *University of Leeds, Leeds, UK*
[2] *University of Oxford, Oxford, UK*
[3] *University of St Andrews, Fife, Scotland*
[4] *University of Malaya, Kuala Lumpur, Malaysia*
[5] *Technical University of Denmark, Kongens Lyngby, Denmark*

(*Communicated to* MIIR *on 18 November 2021*)

Study Group: ESGI 162, Leeds, 20-24 July 2020

Communicated by: J.A. Ward

Industrial Partner: Innovation Embassy

Presenter: Erik Micheelsen .

Team Members: S. Abrahams, University of Oxford; R. Ali, University of Leeds; A. Berryman, University of Oxford; C. Bleak, University of St Andrews; N.A. Hamzah, University of Malaya; T.F. Khang, University of Malaya; P.G. Hjorth, Technical University of Denmark; C.M. Ng, University of Malaya; Y. Tian, University of Oxford; J.A. Ward, University of Leeds' H. Yang, University of Oxford;

Industrial Sector: Finance

Tools: R, Python, SQL and MATLAB

Key Words: Gambling, Customer Lifetime Value, Exploratory Data Analysis, Clustering, Predictive Modelling

MSC2020 Codes: 62,91

† Corresponding Author: gyral@leeds.ac.uk

Abstract

We were asked by Innovation Embassy to work with a large dataset centred around gambling investment, with the task of making a predictive function for computing Customer Lifetime Value (CLV), and also to see if there are ways of detecting fraudulent financial practices and addictive gambling patterns. We had moderate success with the data as it stands, but we were partly held back for two main reasons: the ability to discern a solid definition of CLV due to highly inconsistent data and data that contained many large and incomputable gaps. The team used a number of linear regression models to find CLV functions based on key variables. We also describe a short and explicit list of ways where the base data can be improved to support effective calculation of CLV.

1 Executive summary

After some consideration, we have chosen our definition of CLV as *the sum of the deposits made by a customer*. As the reader can surely see, this choice is already fraught with difficulties, and reflects the apparent main obstacles to meaningful analysis: inconsistent and incomplete data. Due to our great difficulties with the data, as indicated here, while we have figures to report, our confidence levels in these figures are low. Particularly, our opinion is that the current results should not be taken as conclusive in any way.

1.1 Key results/products

Here is a short list of calculation results we can offer:

(1) CLV for the mean customer is 1035. However, the vast majority of customers have CLV's well below this figure, as will be clear from the immediately following points.

(2) Roughly 80% of revenue is brought in from roughly 10% of the clients, and this progression appears to loosely scale.

(3) Unsurprisingly (given the above), clustering identified two groups of customers with different CLV. Customers in the group with higher CLV tended to have longer duration and very large total turnover. In any case, splitting the data into two groups based on high and low CLV improved the performance of the regression models.

(4) Of initial parameters for market aggregation data, `brand` matters. For strategic planning, careful attention should be focused on developing `brand` before a marketing push.

(5) Predicting total deposits for a collection of customers, rather than single customers, may be useful for executive level planning. Using a Random Forest Model with `brand`, `country`, and `duration` as predictors, the estimated total deposits of a customer test set only exceeded the true total deposits by about 9% (€24.6 million vs. €22.6 million).

A qualifier to the final result mentioned above should be stated: the parameter `duration`

runs from first deposit through the date this study began. It is natural that one can make better predictions when the duration parameter is long, as one effectively is predicting the full result, based on all available data, as opposed to making future predictions based only on initial data.

Detection of meaningful predictive parameters is the main difficulty in our analysis, and this process was greatly hindered by base data that is inconsistent and incomplete. While we had a measure of success imputing missing data using "K-nearest neighbours", it was by no means sufficient to consider those results to be either satisfying or trustworthy. In any case, in the main body of the report we describe carefully the regressions we run, and the various methods we employed to try to work around the various large gaps in the (broken) data.

We are confident that with more integral data, our analytical methods would provide more fulfilling results, so we provide as well all of the essential code we employed, for analysis and use by future teams working with better base data.

1.2 Improving the data

A positive outshoot of the above is that we have some actionable recommendations for improving the data, so that more reliable CLV prediction functions might be generated in the future (both at individual and market campaign aggregation scales), and also to support greater utility in detecting fraudulent or addictive practice. Companies wishing to have more reliable CLV calculations, and to more readily detect money laundering and addicted gamblers (so as to protect these individuals), should be more than willing to supply the requested more detailed information in the future.

We believe it would behoove Innovation Embassy and Partners to develop a template of minimum requirements that providers can "live up to." Fulfilling these reporting requirements could be taken as concrete evidence that providers are working to detect and stop money laundering, and the participation of addicted gamblers. This might not only be morally satisfying for providers, but also be of assistance to them in demonstrating that they are working to satisfy local legal requirements.

The data as it stood could have been improved in various ways:

(1) Clarity of variable definitions – often times even the currency was not obvious. As listed, definitions provided are likely useful for inhouse use but not for external evaluation.
(2) Inclusion of withdrawal data.
(3) Consistency between providers in values of data reported.
(4) Data about the business model of different providers.

For the final point, we understand that different providers will have different fee structures: various percentages of winning bets, where both sides of the bet are between customers, versus, e.g., the customer staking directly against the house. If there is no

data on how fees are drawn from the gamblers, then there is no way to accurately predict the profit to the provider from the customer.

This is a key point and of tremendous importance for any prediction of CLV and also for detection of addiction or fraudulent financial practice: being able to "follow the money": when we cannot see how much money is withdrawn, how often, and with consistent and full reporting of turnover. It then becomes impossible to discern (compute) where the money goes.

This means we cannot understand if it went to the provider as profit, or, if it was lost by one customer to another, or if it was withdrawn and saved by the depositing customer.

Developing this further, studies indicate that a flag for a potential addicted gambler is ever increasing time and money spent on gambling [10][11]. In an online setting, increasing time and money would equate to increasing rates and sizes of deposits. When withdrawal data is withheld, it becomes difficult to identify addiction, and impossible to identify money laundering and other fraudulent practices.

We built a number of regression models to find CLV functions based on key variables. We attempted to derive CLV models at the scale of individual customers as well as for market campaign aggregates. We were hindered by incompleteness of data in various ways. This has led to results where we initially cannot discern a meaningful CLV prediction as originally defined due to the difficulty with the data as provided.

However, we have found how the data can be improved in various ways for future teams to improve the CLV definition:

(1) Clarity of variable definitions. Although definitions were given to us, they are for inhouse use and not clearly defined (purified) for external understanding.
(2) Inclusion of withdrawal data. This point is crucial together with variable (and unknown) provider fees.
(3) Consistency between between providers in values of data reported.
(4) Data about the business model of different providers (particularly their fee structures).
(5) The sum of the deposits made by a customer was used as a proxy for CLV, although other definitions could be defined to improve predictions
(6) Linear regression models were built using two different sets of independent variables.
(7) For the first linear regression model, Exploratory Data Analysis (EDA) was used to select the most relevant variables to predict CLV.
(8) The second linear regression model used only variables derived from the first three deposits made by a customer
(9) There are significant outlier customers in terms of Net Revenue (both positive and negative) and Value of Deposits.
(10) We found that 80% of the total deposits were made by only 10% of the customers.

(11) Clustering identified two groups of customers with different CLV. Customers in the group with higher CLV tended to have longer duration and very large total turnover.

(12) Splitting the data into two groups based on high and low CLV improved the performance of the regression models.

(13) Predicting total deposits for a collection of customers, rather than single customers, may be useful at executive policy level planning.

Using a Random Forest Model with Brand, Country, and Duration as predictors, the estimated total deposits of a customer test set only exceeded the true total deposits by about 9% (€24.6 million vs. €22.6 million).

2 Introduction

Innovation Embassy wishes to find a way to quantify from early data for a given customer, the monetary value of a customer over either a set time, or possibly the total time the customer will be using a provider - in this case a betting website. The latter has given rise to the term Customer Lifetime Value *Customer Lifetime Value (CLV)*. The CLV is essentially the total (net) profit over the lifetime of the customer.

The dataset provided to the study group by Innovation Embassy includes individuals who sign up to a betting-operator and bet various amounts over time, at inconsistent intervals, beginning the data series as a First Time Depositor (FTD). Over time, wins, losses for the customer (these are respectively losses and wins for the betting operator) are recorded daily and logged as a time series. Each betting operator logs (1) a customer report (each customer has a unique identifier) and (2) a campaign report (each campaign has a unique identifier).

The customer report can be used to track how much an individual customer plays, while the campaign report provides information on each marketing campaign (e.g., an internet link found on a web page containing an advertisement).

One of the main problems facing the study group is that the largest betting-operators provide only the campaign reports, and not the customer reports. For this reason majority of the data is incomplete. The key to discerning the betting-operators who only provide campaign reports is by first identifying a relation given by the operators who provide both campaign and individual customer reports.

The study group is also asked to identify ways of uncovering non-standard betting patterns (this could for instance be high-value bets or extremely large deposits).

Innovation Embassy asked the study group to:

(1) Define a customer profile from intact data

(2) Define the CLV by using the customer profile and determine the uncertainty of this estimate

(3) Make short time predictions for the Customer Value

3 Data Structure

Innovation Embassy is working with a client that collects data on web betting operators; specifically, they collect data on online marketing campaigns and customer activity. The data is provided by the betting operators for the purpose of tracking the success of the campaigns and the behavior of customers. There are a number of datasets made available for this study group, two of which are *campaign reports*, containing data on marketing campaigns together with two *customer reports*, containing data on customer transactions. And also one large data set of only *campaign report*, which can be made discernible once the relationship between the *customer reports* and *campaign report* has been found.

3.1 Customer Reports

For this project, with the aim of predicting the CLV, we focus just on the two customer reports, which are named: 'netrefer_customer_finance' and 'netrefer_customer_alias'. Both customer reports contain a row of data for each day that each customer is active on a each betting website, for example, if one customer bets on three different days this will appear as three separate rows. 'netrefer_customer_finance' contains the entries described below. 'netrefer_customer_alias' contains similar information as 'netrefer_customer_finance', but is enriched with additional information, as described below. In this report we will indicate in each section which dataset has been used.

For 'netrefer_customer_finance', the entries are:

- **date**: date that the customer was active,
- **report_type**: indicates if this is a customer or marketing report – always 'customer' in this dataset,
- **marketing_source**: name of the marketing campaign that lead the customer to the betting website,
- **id**: unique identification code given to each customer,
- **brand**: name of the betting operator,
- **country**: country where customer signed up,
- **deposits**: value of deposit – amount that the customer deposited to their account in that day,
- **net_revenue**: revenue made by the operator from this customer on this day,
- **net_revenue_mtd**: revenue made by the operator from this customer between the first day of the current calendar month and the current day of the calendar month,
- **alias**: definition unknown,
- **customer_type**: same as 'country',
- **customer_signup_source**: type of device that was used – either 'Desktop' or 'Mobile web',
- **signup_date**: date that customer first used the site
- **transactions**: number of transaction made by the customer on this date (deposits and withdrawls or bets too),
- **turnover**: amount that the customer bet or "put on the stake" on that day,

The 'netrefer_customer_alias' report contains additional information such as the exchange rates to Euros at the time of conversion. For entries that have the same definition as entries in 'netrefer_customer_finance', some of the names differ slightly. For 'netrefer_customer_alias' the entries are:

- **date**: date that the customer was active,
- **url**: url of the page where the customer clicked a link to the betting website,
- **campaign_name**: name of the marketing campaign that lead the customer to the betting website,
- **customer_id**: unique identification code given to each customer,
- **brand_report**: name of the betting operators,
- **country_report**: country where customer signed up,
- **alias_report**: definition unknown,
- **customer_signup_source**: type of device that was used, e.g mobile phone,
- **turnover**: amount that the customer bet or "put on the stake" on that day,
- **vod**: value of deposit – amount that the customer deposited to their account in that day,
- **total_net_revenue**: revenue made by the company from this customer on this day,
- **channel**: the definition of "channel" is not consistent across the betting companies so this entry will not be analysed,
- **country**: same as 'country_report',
- **campaignname**: same as 'campaign_name',
- **gbp2eur**: exchange rate for GBP to EUR on date specified,
- **dkk2eur**: exchange rate for DKK to EUR on date specified,
- **usd2eur**: exchange rate for USD to EUR on date specified.

For both customer report datasets, there is uncertainty over the exact definitions of the entries and whether they are consistent between providers. For example, it is not clear whether the entries for 'turnover', 'deposits' and 'net_revenue' had been already converted to Euros or not; since no currency was given, we make the assumption that the values given are all in Euros.

3.2 Preliminary Data Analysis

The customer report data is incomplete and many lines have "null" entries. The analysis in this section is on 'netrefer_customer_alias'. Analysing the data, we find the following:

- There are 1,348,421 rows.
- There are 64,824 distinct 'customer_id'.
- 1,121,426 rows have non-null 'vod', of which there are 56,688 distinct 'customer_ids's.
- The total sum of (non-null) 'vod' is 68,533,433.
- 1,301,124 rows have non-null 'total_net_revenue'.
- 656,761 rows have positive 'total_net_revenue'.
- 229,296 rows have negative 'total_net_revenue'.
- 415,067 rows have zero 'total_net_revenue'.

- The total sum of (non-null) 'total_net_revenue' is 10,213,633.
- There are 1,119,911 rows where 'vod' and 'total_net_revenue' are non-null.
- The total sum of 'vod' when 'vod' and 'total_net_revenue' are non-null is 68,470,709.
- The total sum of 'total_net_revenue' when 'vod' and 'total_net_revenue' are non-null is 9,115,338.
- Earliest date: 2007-01-01.
- Latest date: 2020-07-19.
- There are 122 distinct 'brand_report'.
- Sum of 'vod' where 'total_net_revenue' is zero is 14,185,271.
- Sum of 'vod' where 'vod'¿1000 is 17,896,763, which consists of 1,894 distinct ids (26% of total 'vod', 3% of 'customer_id').
- Sum of 'vod' where 'vod'¿10000 is 1,112,274, which consists of 26 distinct ids (1.6% of total 'vod', 0.04% of 'customer_id').

3.3 Aggregation of Customer Data

We aggregate the data for each customer, collecting lines with the same 'customer_id'. Due to the amount of missing entries, we take a subset of the data containing only customers with sufficiently complete data.

Aggregating the data allows us to create relevant new variables for each customer, these are:

- **sum_deposits**: sum of 'deposits' for a customer,
- **deposits_first**: first entry for 'deposits' for a customer,
- **sum_turnover**: sum of 'turnover' for a customer,
- **sum_net_revenue**: sum of 'net_revenue' for a customer,
- **sum_net_revenue_mtd**: sum of 'net_revenue_mtd' for this customer,
- **date_min**: first date customer is active,
- **date_max**: last date customer is active,
- **duration**: time from first activity date, to 22 July 2020 (date when the analysis is done) in days,
- **time_duration**: number of days between 'date_min' and 'date_max',
- **first_duration**: number of days between first and second entries for 'deposits',
- **date_count**: number of days that the customer logged into the operator site, i.e. number of lines for a single customer
- **deposits_mean**: 'sum_deposits' divided by 'date_count',
- **deposits_first**: first entry for 'deposits' for a customer,
- **first_country**: first entry for 'country' for a customer,
- **first_customer_type**: first entry for , 'customer_type', for a customer,
- **first_marketing_source**: first entry for 'marketing_source' for a customer,
- **first_brand**: first entry for 'brand' for a customer,
- **first_customer_signup_source**: first entry for 'customer_signup_source' for a customer,
- **net_revenue_min**: minimum value of 'net_revenue' for a customer,

- **net_revenue_max**: maximum value of 'net_revenue' for a customer,

3.4 Defining CLV

As the provided definition was not possible to calculate given the provided data, an initial practical definition was chosen 'sum_deposits' as a proxy for CLV.

'sum_deposits' is not the only way to define CLV and other definitions could equally be considered. For example, the lifetime value of a customer to a company is closely linked to the net revenue that this customer brings, therefore 'sum_net_revenue' could be used as the CLV. However we suspect that the 'net_revenue' has not been consistently recorded by the betting operators due to the large amount of zero or missing entries (NULL) in this column. It has been suggested that some betting operators only record positive 'net_revenue' values, however we cannot verify this.

'sum_deposits' is the total amount deposited by the customer to the betting company over the time that they have been active on the website. We do not have any withdrawal data, which, alongside deposits, would allow us to calculate the amount lost by the customer; deposits minus withdrawals would be another option for the definition of CLV. However, without withdrawal data, for the purposes of this study, we make the assumption that total deposits represents the value that the customer brings to the company, and therefore we use this as the CLV.

4 Exploratory Data Analysis (EDA)

4.1 Plots

Figure 1 shows that there are three variables with missing values.

This plot helped us to make informed decisions about how to handle these missing values and assess whether the Missing At Random (MAR) assumption would be plausible. Further details on how these missing values were handled can be found in the subsection 6.2.1.
The data and code used to produce Figure 1 are included in section A.6 of the Appendix.

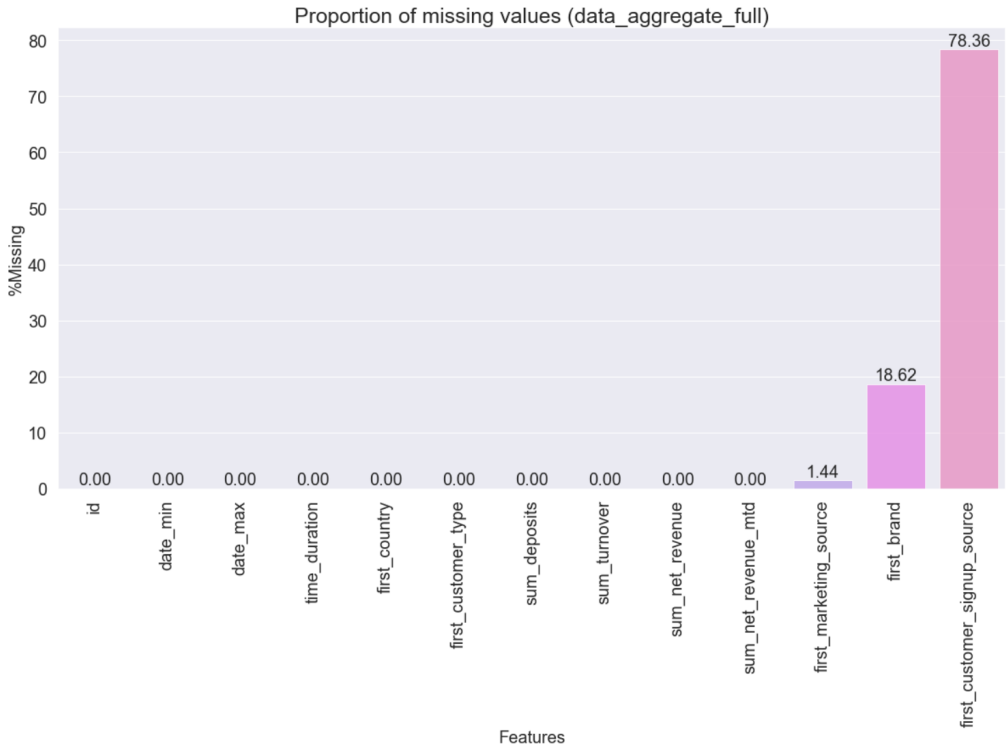

Figure 1. Percentage of missing values for each variable

Figure 2 is a scatter plot of the sum of 'vod' against the sum of 'total_net_revenue' for each 'customer_id' in the 'netrefer_customers_alias' table.
The data and code used to produce Figure 2 are included in section A.1 of the Appendix. In Figure 2, blue markers indicate customers with sum of 'total_net_revenue' equal to zero and red markers indicate customers with sum of 'vod' equal to zero. It seems unlikely that customers would have sum of 'total_net_revenue' exactly equal to zero, although this may depend on the types of games they have played. The red markers in Figure 2 suggest that there are players who have contributed to non-zero sum of 'total_net_revenue' despite having not deposited any money. One possible explanation is that the players had made a deposit prior to the earliest date in the data. However, it may also suggest that our understanding of these quantities does not accurately reflect what is being reported, or that reporting practices may vary between betting providers. The grey lines indicate where the magnitude of the sum of 'total_net_revenue' is equal to the sum of 'vod'. Note also in Figure 2 that there are some significant outliers, i.e. customers who have made extremely large total deposits and one who has an extremely large sum of 'total_net_revenue' loss.

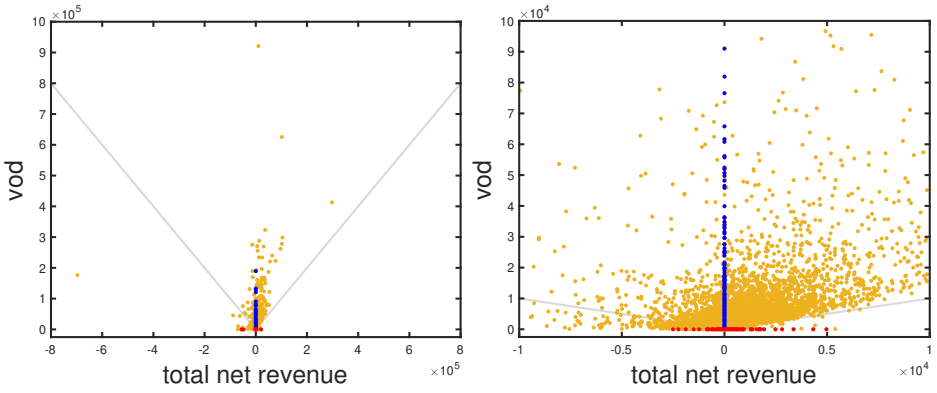

Figure 2. Scatter plot of the sum of 'vod' against the sum of 'total_net_revenue' for each 'customer_id' in the 'netrefer_customers_alias' table. Blue markers indicate customers with sum of 'total_net_revenue' equal to zero and red markers indicate customers with sum of 'vod' equal to zero.

Figure 3 shows the pairwise comparison for continuous variables. This plot allows us to see any relationships and the spread of each data point. It takes the continuous variables, places them on both the x and y axes, plots a scatter plot where they meet, and fits a linear regression model including the confidence interval band. The diagonal subplots are the univariate histograms (distributions) for each variable. For example, it illustrates that there is a positive linear relationship between 'sum_deposits' and 'sum_net_revenue'. The data and code used to produce Figure 3 are included in section A.8 of the Appendix.

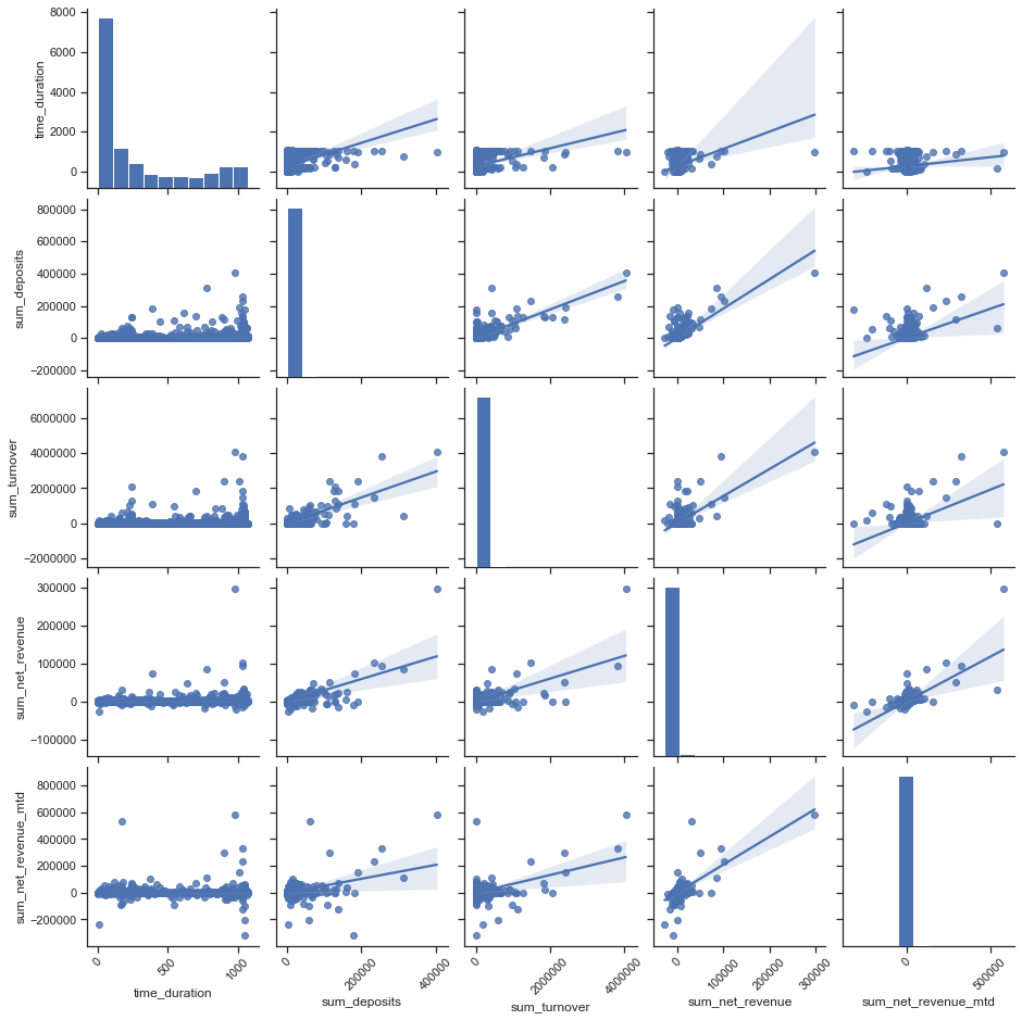

Figure 3. Pairplot (Distribution of continuous variables)

Figure 4 reveals that the amount of the $\log_{10}(\text{Total deposits} + 1)$ for some brands are different from that of other brands, judging from the median and spread from the boxplots. This indicates that the brand might play a role in attracting higher total deposits and

hence, higher revenue from the customers. The data and code used to generate Figure 4 are provided in section A.7 of the Appendix.

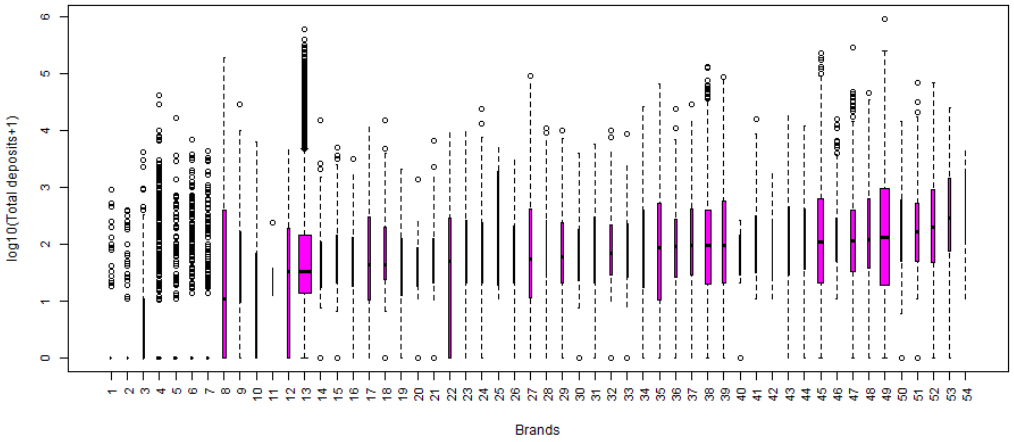

Figure 4. Amount of total deposits by brands

4.2 Correlation matrix

To analyse the relationship between variables, in Figure 5 we plot a heat map showing the Pearson correlation between each pair. This is used to identify important variables and choose suitable predictors for the linear regression analysis. Since we choose 'sum of deposit' as the dependent variable, we can see that the most relevant variables, i.e. those that are likely to be chosen as the independent variables in the linear regression model are 'deposit_first', 'sum_turnover', and 'sum_net_revenue_mtd'.
The code used to generate Figure 5 is provided in the section A.2 of the Appendix.

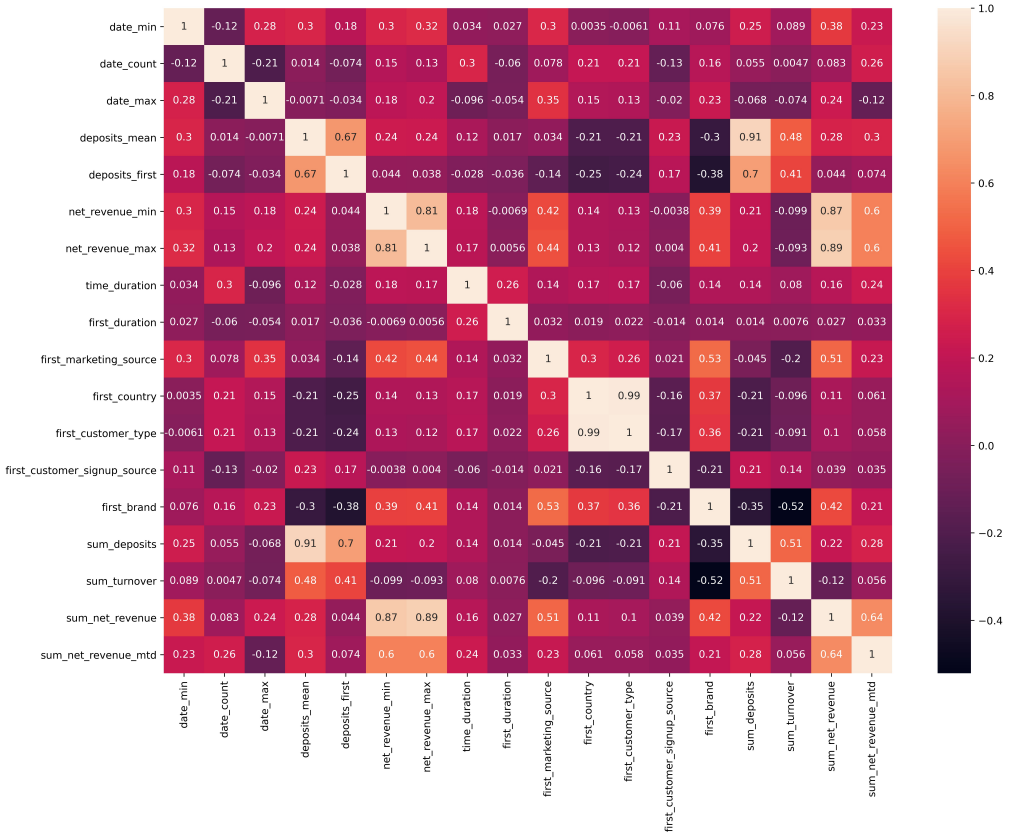

Figure 5. Correlation matrix between the variables

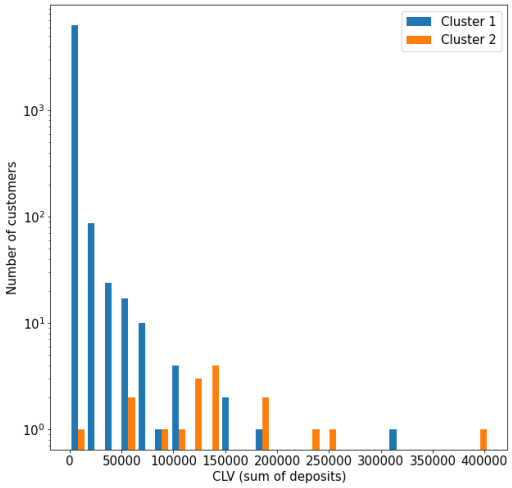

Figure 6. Histogram of customers clustered using the k-prototypes algorithm with two clusters. Customers in Cluster 2 with higher CLV were identified as having longer durations of play and very large total turnover.

4.3 Clustering

In order to gain more understanding of the relationship between customer profiles and CLV, we looked at clustering the customers according to their profiles. Since the customer profiles include categorical and numerical variables (e.g. 'country' and 'net_revenue'), it will not be enough to use the popular k-means clustering algorithm as this can only handle numerical variables. There exists an extension to the k-means algorithm that incorporates categorical variables, as introduced by Huang [8], called the *k-prototypes* algorithm.

The k-prototypes algorithm is similar to the k-means algorithm when dealing with numerical variables but uses a matching dissimilarity measure for categorical variables. The dissimilarity measure for numerical variables is the squared Euclidean distance and the dissimilarity measure for categorical variables is the number of mismatching categories between two instances. A linear combination of these two dissimilarity measures is used to update the cluster prototypes during the algorithm.

In Figure 6, we can see that two clusters found using the k-prototypes algorithm enable us to identify customers with higher CLV (sum of deposits). These customers with higher CLV, in Cluster 2, also tend to have longer durations of play and also very large total turnover. This result is encouraging when moving forward to trying to predict customer lifetime value because this shows that the customer profiles are relevant to identifying differences in CLV.

We employed the 'elbow method' to decide how many clusters to use, shown in Figure 7. The elbow method is an informal method used to determine the number of clusters present in a dataset. We can see in Figure 7 that this method suggests to use three clusters. However, we chose two clusters because this gave us a clearer split between

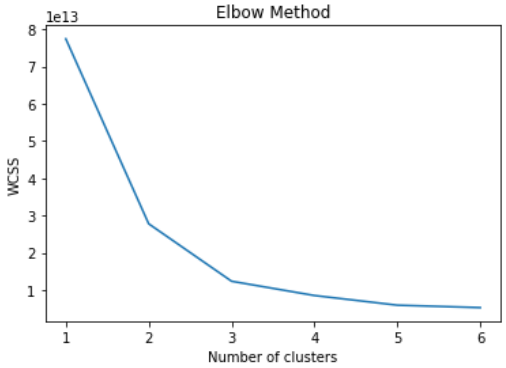

Figure 7. Plot to show how the WCSS (within cluster sum of squares) for different numbers of clusters used in the k-prototypes algorithm.

customers with different CLV that can be leveraged to improve our model's predictive power.

The data and code used to generate Figure 6 and Figure 7 are provided in section A.4 of the Appendix.

4.4 80/20 Rule

We want to test the '80/20 rule', i.e. whether 80% of profits come from 20% of customers. We consider the total net revenue, denoted ρ_i for each customer i in the database who has non-null 'vod' and 'total_net_revenue'. There are $N = 56,562$ such customers. For a given threshold $\tau > 0$ we computed the sum of total net revenues whose magnitudes are larger than τ, i.e.

$$\sum_{|\rho_i|>\tau} \rho_i.$$

We then computed the percentage net revenue above the threshold τ,

$$\rho_\tau = 100 \times \frac{\sum_{|\rho_i|>\tau} \rho_i}{\sum_i \rho_i}.$$

We also computed the percentage of customers whose total net revenue is greater than τ,

$$N_\tau = 100 \times \frac{1}{N} \sum_i \mathbf{1}_{\{|\rho_i|>\tau\}}.$$

In Figure 8 we plot the percentage net revenue above threshold ρ_τ in blue and the percentage of customers above threshold N_τ in red. Note that the horizontal axis is scaled logarithmically. The data and code used to produce Figure 8 are included in section A.3 of the Appendix. In Figure 8, the horizontal grey lines correspond to 80% and 20% and the vertical grey lines indicate where the intersections with the percentage net revenue above threshold and the percentage customer above threshold. If this data agreed with the 80/20 rule then there would only be one vertical grey line. Here there are two and we can see that actually 80% of the deposits are made by more like 10% of the customers.

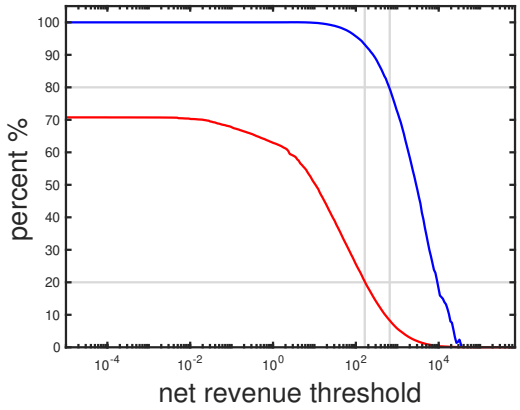

Figure 8. Percentage net revenue above threshold ρ_τ (blue) and percentage of customers whose total net revenue is above threshold N_τ (red) against the total net revenue threshold τ. The horizontal grey lines indicate 80% and 20% and the vertical grey lines indicate the intersections with the percentage net revenue above threshold and percentage of customers above threshold respectively. The 80% intersection with the percentage net revenue above threshold lies to the right of the 20% intersection with percentage of customers above threshold, indicating that fewer than 20% of customers (roughly 10%) make up 80% of the total net revenue.

5 Feature Selection

Feature selection is the process of selecting (automatically or manually) variables that are most relevant to the given predictive modeling problem. Eliminating irrelevant features will result in reduced computation time and improved model accuracy [6].

The Boruta algorithm is a wrapper built around the random forest algorithm. Although there are many different feature selection techniques to choose from, we used the Boruta algorithm for following reasons:

- It works well with both classification and regression problems.
- It takes into consideration multi-variable associations.
- It uses all-relevant variable selection method (takes into account all the variables that are relevant to the outcome variable). Whereas, most of the other feature selection methods use a minimal optimal method (take into account only a small subset of features that yields a minimal error).
- It deals with interactions between variables.

A brief explanation of how this algorithm works is given below [7]:

(1) Generate duplicate copies of all independent variables.
(2) Shuffle the values of added attributes in order to remove their correlations with the dependent variable.
(3) Perform a random forest model on the extended dataset and create a variable importance measure (the default is Mean Decrease Accuracy) to assess the importance of each feature, where higher means more important.
(4) Calculate the z-score (mean of accuracy loss divided by standard deviation of accuracy loss).
(5) Find the maximum z-score among the permuted copies.
(6) Label the features as "unimportant" if their importance measure is significantly lower than the permuted copies.
(7) Label the features as "important" if their importance measure is significantly higher than the permuted copies.
(8) Repeat the procedure until all features are either labelled "unimportant" or "important".

Figure 9 shows the relative importance of each independent variable. The x-axis lists each independent variable. Green colour indicates the independent variables that are relevant to the given predictive modeling problem. Red colour indicates independent variables that are irrelevant. Independent variables that may or may not be relevant to predicting the response variable are shown in yellow. For example, according to this plot 'sum_turnover' is the most important predictor of 'sum_deposits'. The correlation matrix shown in Figure 5 also illustrates a positive correlation between 'sum_deposits'

and 'sum_turnover'.

The data and code used to produce Figure 9 are included in section A.9 of the Appendix.

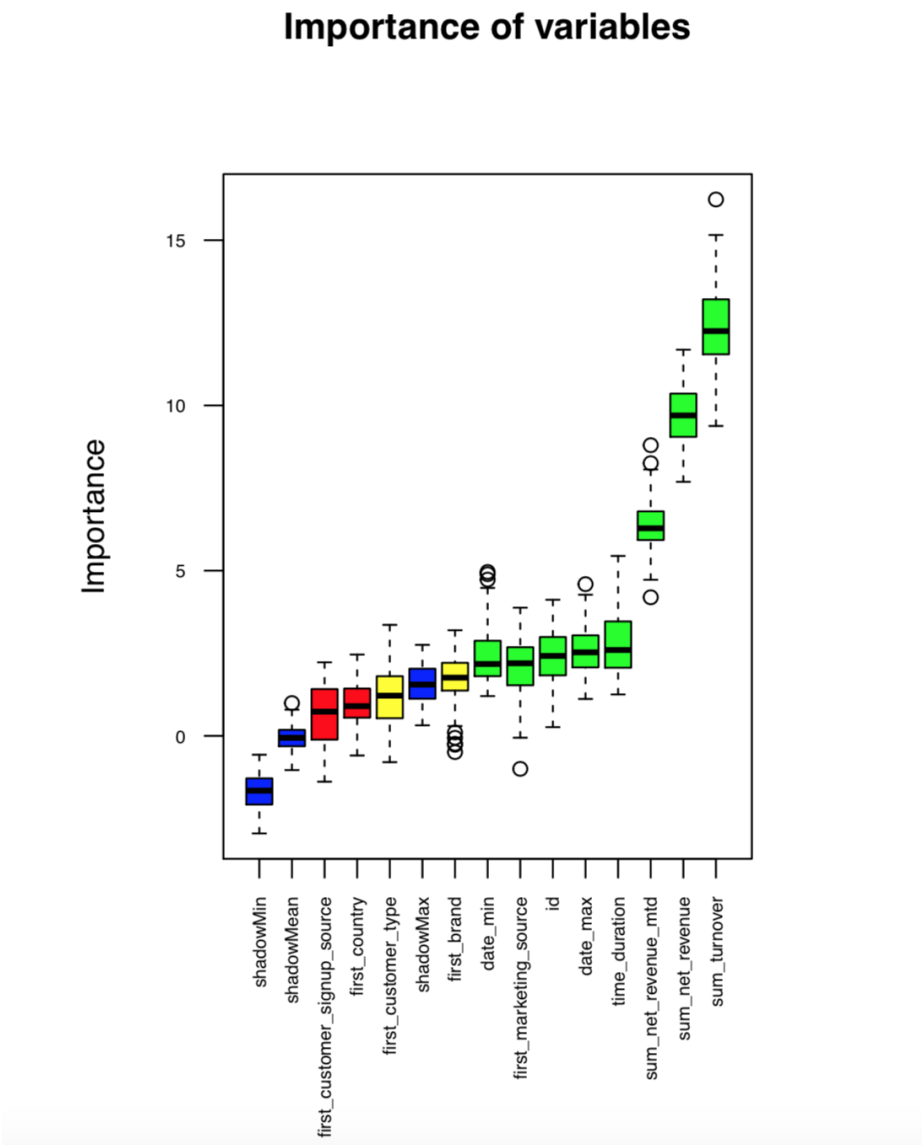

Figure 9. Feature Importance plot (Boruta package)

6 Predictive Modelling

6.1 Linear Regression

Linear regression is a statistical method that is used to predict a dependent variable based on one or more independent variables [2].

The equation for linear regression is given below:

$$y_i = B_0 + B_1 x_{1,i} + B_2 x_{2,i} + ... + B_k x_k, i + \mathcal{E}_i,$$

Where,

- y_i is the dependent variable
- $x_{i,i} \ldots \ x_{k,i}$ are the explanatory variables
- B_0 is the intercept
- $B_1, \ldots B_k$ are the coefficients
- $\mathcal{E}_i$ are the residuals (random error).

6.1.1 *Model Selection*

To choose our model, here we use the coefficient of determination, R^2, which shows the proportion of the variance in the dependent variable that is predictable from the chosen explanatory variable(s). It is defined through the total sum of squares, SS_{tot}, which is proportional to the variance of the data, and residual sum of squares, SS_{res}, which is proportional to the variance explained by the residual, as follows.

$$\begin{aligned} SS_{\text{tot}} &= \sum_{i=1}^{n} (y_i - \bar{y})^2, \\ SS_{\text{res}} &= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2, \\ R^2 &= 1 - \frac{SS_{\text{tot}}}{SS_{\text{res}}}, \end{aligned}$$

where $\bar{y} = \sum_{i=1}^{n} y_i$ is the mean of the observed data and $\hat{y}_i$ is the fitted value. It is easy to verify that $SS_{\text{res}} \geq SS_{\text{tot}}$, hence $R^2 \in [0, 1]$. A higher value of R^2 means that the model has more explanatory power, and a value of 1 indicates that the regression predictions perfectly fit the data.

There are drawbacks in using R^2, e.g. where one might keep adding variables (e.g. 'kitchen sink' regression) to increase the R^2 value. Hence, we also consider the adjusted R^2 which is almost the same as R^2 but penalizes the statistic when extra variables are included in the model.

Sections 4 and 5 helped us to make an informed decision about which variables to include in our predictive models. We explore almost all combinations of possible explanatory variables, and compare the goodness of fit of the models through R^2 and adjusted R^2. Note we also construct the corresponding categorical variables from some numerical ones,

for example, we split the turnover into three groups: less than 0, less than 80% and the remaining, since these variables are not completely reliable but their ranges are. Finally, we choose the model with the following independent and dependent variables:

- Independent variables: first deposit, time duration, turnover (indicator), net revenue (indicator), simple splitting based on deposits(left) /clusters(right), and cross terms
- Dependent variable: sum of deposits

6.1.2 *Model Performance*

Here we provide the results from the best model without clustering. We split the dependent variable into two groups, the one with $y > 1000$ and the other with $y \leq 1000$. The fitted value (y_{fit} in Figure 10) is close to the true value (y_{true} in Figure 10), in the sense that the scatter plot is close to the identity line $y = x$. The values of R^2 for the two groups are 0.103 and 0.449, respectively, indicating that the model explains more variance for the second group.

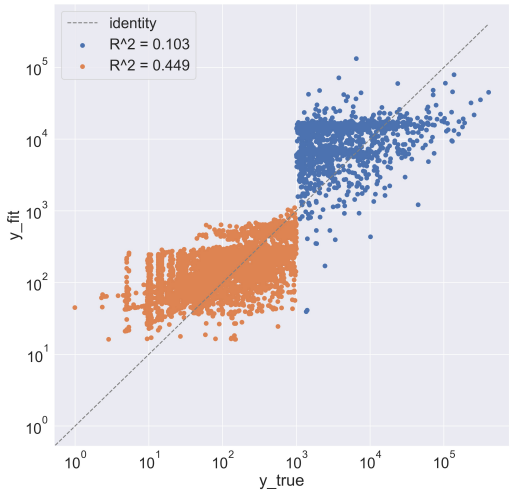

Figure 10. Performance of the regression model.

Then we incorporate the results from clustering (see subsection 4.3), where we treat the clustering index as an extra (categorical) independent variable. The model has a better performance than the previous one, with $R^2 = 0.542$ (see Figure 11).

The data and code used to generate Figure 10 and Figure 11 are provided in section A.5 of the Appendix.

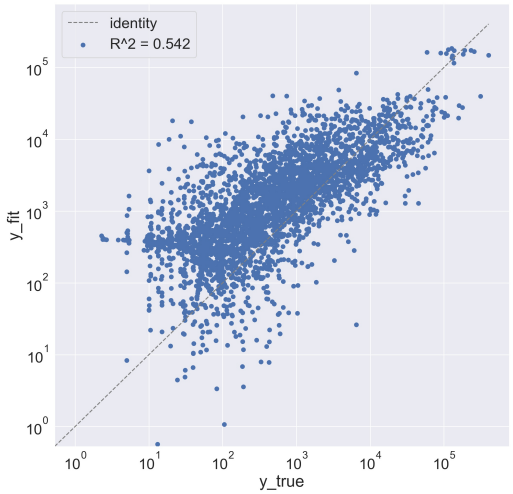

Figure 11. Performance of the regression model with the clustering index as an extra (categorical) explanatory variable.

6.2 Multiple Linear Regression with Cross-validation (imputed data)

The subsection 6.1 covered linear regression with and without the clustering index as an extra categorical variable (all the missing values were removed), we wanted to explore how the model will perform on unseen and imputed data. Hence, this subsection focuses on linear regression with cross-validation on imputed data. We make the MAR assumption and impute the variables with less than 20% missing values [9]. Subsequently, to test how the model would perform on unseen data, we use the k-fold cross-validation approach. A detailed explanation of this technique is provided in 6.2.2

.

6.2.1 *Imputation using KNN*

Assumptions:
MAR [3]: there could be some variations between missing and observed values, but these can be fully explained by other observed variables.

The K-Nearest Neighbor (KNN) algorithm is an imputation technique that is very efficient and simple to implement [1]. In this method, K nearest or similar data points are selected using all the non-missing features and then the average of the selected data points is computed to fill in the missing feature.

Figure 1 illustrated the percentage of missing values for each variable:

(1) **first_customer_signup_source**: 78.36%. It was deemed too excessive for imputation to be meaningful, hence it was removed.
(2) **first_brand**: 18.62%. It was imputed using the KNN algorithm.
(3) **first_marketing_source**: 1.44%. It was imputed using the KNN algorithm.

The data and code used to perform imputation are included in section A.10 of the Appendix.

6.2.2 *5-Fold Cross-validation*

Cross-validation is a re-sampling technique used to evaluate machine learning models on a limited dataset. It takes one parameter called "k", which is the number of subsets a given dataset is to be split into (in this case k=5). This technique is very popular as it is easy to implement and interpret. The key advantage of using cross-validation is that it prevents us from overfitting the data.

A brief explanation of the various stages of 5-Fold Cross-validation is provided below:

(1) Split the dataset into 5 subsets;

(2) For each subset:

- Take one subset as the hold-out or test dataset.
- Take the remaining data as the training dataset.
- Fit the model on the training dataset and evaluate its performance on the test dataset.
- Keep the evaluation scores, but discard the model.

(3) Use the sample of model evaluation scores to evaluate the model's performance.

Figure 12 shows the performance of the most parsimonious cross-validated model, which was generated using 'first_brand', 'time_duration', 'sum_turnover' and 'sum_net_revenue'. The x-axis shows the true values and the y-axis illustrates the fitted values. Model's R^2 (a detailed explanation of R^2 is provided in 6.1.1) value is 0.18 (i.e. the model explains approximately 18% of the variation).
The data and code used to produce Figure 12 are included in section A.11 of the Appendix.

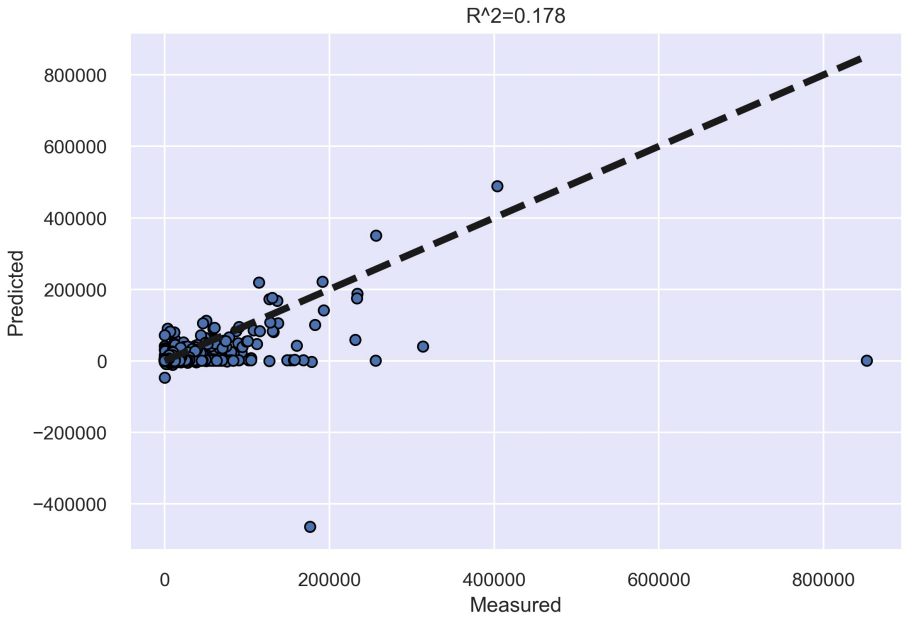

Figure 12. Predicted VS Actual Values plot (cross-validated model)

6.3 Linear Regression with deposit data only

As a parallel exercise to the linear regression described above, we carry out linear regression using only information about the first three deposits that a customer makes. Innovation Embassy is interested in knowing if CLV can be predicted using only information about a customer's first few deposits as this is the information that the betting companies have early on in the customer's lifetime; for this reason, in this section, we aim to predict CLV using variables derived from the data on the first three deposits made by customers.

To make the data a manageable size, we extract a subset that includes just the past 3 years: 'netrefer_customers_finance_past_3_years.csv'. See Appendix for SQL command used to extract this data subset. From 'netrefer_customers_finance_past_3_years.csv' we select a further subset that includes only customers that made at least 3 deposits. In addition to 'deposits_first' and 'duration_first', we define a number of further customer variables based only on the first three deposits made:

- **deposits_second**: second entry for 'deposits' for this customer,
- **deposits_third**: third entry for 'deposits' for this customer,
- **mean_two_deposits**: mean of 'deposits_first' and 'deposits_second',
- **mean_three_deposits**: mean of 'deposits_first', 'deposits_second' and 'deposits_third',
- **time_ratio**: days between first entry for 'deposits' and second entry for 'deposits' divided by days between second entry for 'deposits' and third entry for 'deposits',

Following the same method as section 6.1, we carry out linear regression but choosing only dependent variables from those listed above. All linear regression in this section is carried out using the using `Python` package `statsmodels.formula.api`. In the correlation matrix in Figure 13 we can pick out independent variables that correlate most strongly with 'sum_deposits' which is our proxy for CLV. This gives the following model:

- Independent variables: 'deposits_first' and 'mean_three_deposits'
- Dependent variable: 'sum_deposits' (i.e CLV)

Linear regression on this subset of data (customers with at least three deposits) gives the following model for CLV

$$CLV = 2.0 \times \text{'deposits\_first'} + 12.4 \times \text{'mean\_three\_deposits'} \tag{6.1}$$

with an R^2 value of 0.121.

As in section 6.1, we again split the dependent variable into two groups, with the aim of improving the performance of the regression model. We take one group with $y \leq 1000$ and the other with $y > 1000$ where $y =$'sum_deposits'. Carrying out linear regression on the first group, $y \leq 1000$, gives

$$CLV = -0.1 \times \text{'deposits\_first'} + 4.8 \times \text{'mean\_three\_deposits'} \tag{6.2}$$

with an R^2 value of 0.615. As the contribution from 'deposits_first' is small, this vari-

able does not appear to be significant in predicting CLV for this group. Taking only 'mean_three_deposits' as the dependent variable gives

$$CLV = 4.7 \times \text{'mean\_three\_deposits'} \tag{6.3}$$

while retaining an R^2 value of 0.615.

Carrying out linear regression on the second group, $y > 1000$, gives

$$CLV = 2.1 \times \text{'deposits\_first'} + 13.6 \times \text{'mean\_three\_deposits'} \tag{6.4}$$

with an R^2 value of 0.123.

From figures 14 and 15, we can compare visually the true values of CLV (y_{true}) to the values predicted by the regression models (y_{fit}). Splitting the data into two groups significantly increased the R^2 value for the group $y \leq 1000$ from $R^2 = 0.121$ to $R^2 = 0.615$. The increase in R^2 value means that the model could explain more of the variance in the data. From the plot we see that although the R^2 value has improved, the points still deviate significantly from the line $y_{\text{fit}} = y_{\text{true}}$, mostly lying below this line; this indicates that our model for this group generally under-predicts CLV. For the group $y > 1000$, the R^2 value stayed approximately the same.

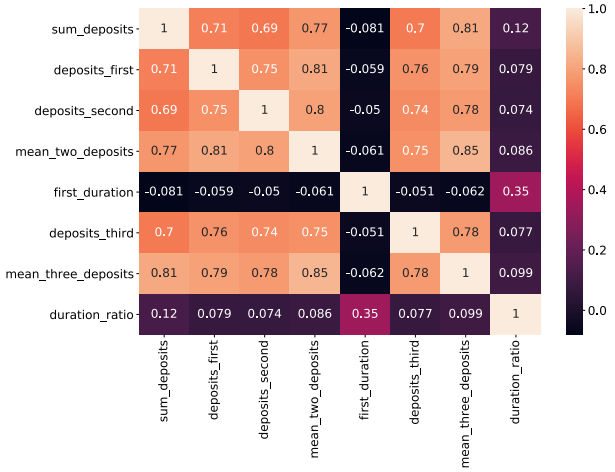

Figure 13. Matrix of correlation between the deposits variables, with 0 being no correlation and 1 being perfect correlation. The diagonal values are 1 as each variable perfectly correlates with itself. The first column is of most interest as it indicates the correlation of 'sum_deposits' with each variable.

The data and code used to generate Figure 13, Figure 14 and Figure 15 are provided in section A.12 of the Appendix.

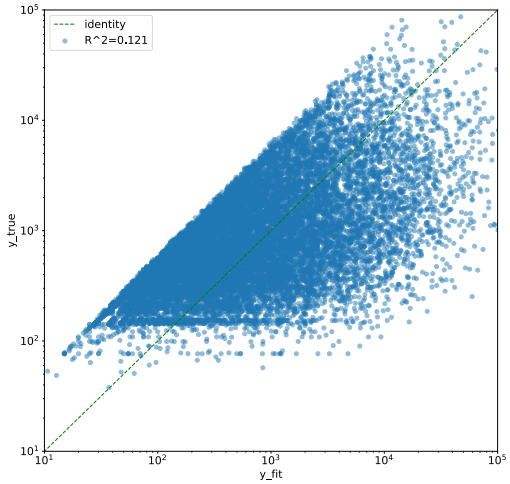

Figure 14. Plot showing the performance of the deposit data regression model. y_{true} is the value for CLV from the data i.e 'sum_deposits' for every customer in this data subset. y_{fit} is the value for CLV as given by equation 6.1. The dotted line is the line $y_{\text{fit}} = y_{\text{true}}$; the closer the plotted points are to this line, the better the regression model.

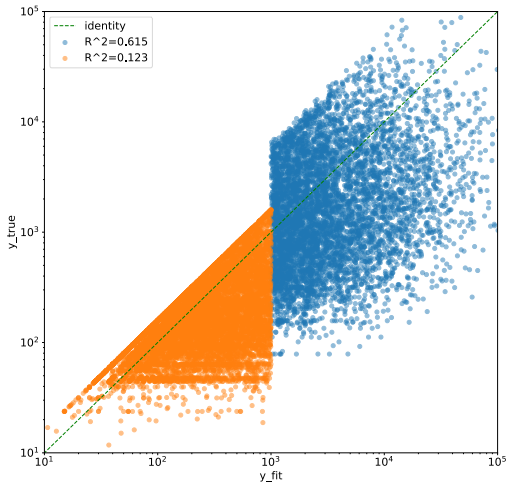

Figure 15. Plot showing the performance of the deposit data regression model with data split by $y =\leq 1000$ (orange) and $y > 1000$ (blue). y_{true} is the value for CLV from the data i.e 'sum_deposits' for every customer in this data subset. y_{fit} is the value for CLV as given by equation 6.3 for $y \leq 1000$ and by equation 6.3 for $y > 1000$. The dotted line is the line $y_{\text{fit}} = y_{\text{true}}$; the closer the plotted points are to this line, the better the regression model.

6.4 Statistical methodology for estimating total deposits using random forests

6.4.1 *Dataset characteristics*

With suitable data pre-processing (done using R Version 3.6.1; [4]), we considered an initial data matrix with 64,284 rows (unique customer id) and 5 columns consisting of 5 variables: (i) total deposits; (ii) count; (iii) country; (iv) brand; (v) duration. For each customer, we created a new variable - mean deposits, which is a customer's total deposits divided by the frequency of their play (count). The code used to perform this analysis is included in the section A.13 of the Appendix.

About 13% (8,136/64,284) of customers had missing values for total deposits. These were removed, leaving 56,688 customers. About 18% (10,026/56,688) of them never made any deposits. With such a high proportion of zeros, we felt that linear regression was unlikely to model variation in mean total deposits well.

To make progress, we converted mean deposit into a categorical variable ("client") with three levels: mean deposits of €0 (class 0), more than €0 but less than €25 (class 1), €25 or more (class 2). The proportions for each of these three classes were 18% (10,026/56,688), 44% (25,089/56,688), and 38% (21,573/56,688), respectively. For refinement, it is possible to adjust the threshold for defining each classes. Generally, this threshold should result in proportions of class 2 and 3 that are approximately balanced (class 0 being special to mark individuals with poor revenue potential).

For brand, we used the top 16 brands accounting for 80% of all brand shares, and grouped the rest of the brands as "Others". Similarly, for countries, we took the top ten most frequent countries (Norway, Sweden, Denmark, ..., Poland, Ireland) and coded the rest as "Others". The duration variable contains the number of days from the time a customer registered with an operator until 22 July 2020.

The final data matrix used consisted of 56,688 rows and 4 columns, with client as the response variable, and brand, country, and duration as predictor variables.

6.4.2 *Statistical modelling*

From a training data set, we could obtain estimates of the mean count in each of the three classes ($\bar{f}_i$, $i = 0, 1, 2$), as well as the 5% winsorized mean of mean deposit ($\bar{x}_i$, $i = 0, 1, 2$). Let n_i be the (unknown) number of class 0, 1, and 2 customers in a future sample of n ($n = \sum_{i=0}^{2} n_i$) customers. An estimate of the total deposit for this set of customer is given by

$$\hat{T} = \sum_{i=0}^{2} \hat{n}_i \bar{f}_i \bar{x}_i = \hat{n}_1 \bar{f}_1 \bar{x}_1 + \hat{n}_2 \bar{f}_2 \bar{x}_2,$$

where $\hat{n}_i$ indicates estimate of the number of class $i, i = 0, 1, 2$ customers. Note that the simplification is because of $\bar{x}_0 = 0$. Assuming independence of mean deposit between different customers, the standard deviation of $\hat{T}$ can be computed as $\sqrt{n}\mathrm{SD}_{\mathrm{train}}$, where

$\text{SD}_{\text{train}} = \sqrt{\sum_{i=1}^{2} p_i \text{Var}_i}$, with p_i and Var_i being the relative frequency and the sample variance of mean deposit of class i customers in the training sample, respectively.

Estimates of n_i in the test sample can be obtained as the predicted number of class 0, 1, and 2 customers using some suitable classification algorithm. For this purpose, we used random forests (1000 trees) as implemented using the GUIDE (Generalized, Unbiased, Interaction Detection and Estimation; Version 35.2) algorithm [5]. Two thirds of the data set were randomly assigned into a training set, and one third into a test set. To enable reproducibility, a seed number (121) was used. Variable importance scoring was also implemented using GUIDE.

6.4.3 *Results*

Brand, country and duration were all important (in this order) for node splitting.

Predicted	True class			Total
	0	1	2	
0	2517	1304	1498	5319
1	162	4229	1098	5489
2	663	2830	4595	8088
Total	3342	8363	7191	18896

Table 1: Classification matrix of prediction outcome for the test sample using GUIDE random forests.

Based on the classification matrix (Table 1), the accuracy of class label prediction was about 60% (11,341/18,896), which is relatively higher than baseline prediction accuracy using majority class (class 1: 44%). The estimates of n_i were obtained from the row marginals of the classification matrix, thus $\hat{n}_0 = 5,319$, $\hat{n}_1 = 5,489$, $\hat{n}_2 = 8,088$. From the training sample, we had $\bar{f}_1 = 23.1$, $\bar{f}_2 = 23.7$, $\bar{x}_1 = 10.9$, $\bar{x}_2 = 120.4$. The estimated total deposits for the test sample was about €24.5 million, with left and right limits being €21.4 million and €27.6 million, respectively ($\pm$ 2 standard deviations). The actual total deposits for the test sample was about 22.5 million Euros.

In the hypothetical event that all samples were correctly predicted, the estimated total deposits for the test sample would be about €22.6 million, almost equal to the actual total deposits value. In contrast, naive prediction using majority class label yields just about €4.8 million, a significant underestimate.

7 Conclusions, limitations and future work

7.1 Conclusions

We have fitted linear regression models with and without clustering index as an extra categorical variable to the data. Our linear regression indicates that first deposit, time duration, turnover (indicator), and net revenue (indicator) are significant predictors of customer lifetime value, which is consistent with the correlation matrix given in Figure 5. A better performance of the model with clustering index implies that a more intelligent classification of the customers could help improve the prediction.

Linear regression with cross-validation revealed that first brand, time duration, sum turnover and sum net revenue were the key predictors of CLV='sum_deposits'. The Boruta package, which was used to perform feature selection also identified sum turnover, sum net revenue and time duration as the most important features.

Another linear regression model was built using only customer deposit data. The performance of this model was improved by splitting the customers by 'sum_deposits'$\leq$ 1000 and 'sum_deposits'$>$ 1000. Splitting allowed us to find an improved model for the group 'sum_deposits'$\leq$ 1000, which is given by the equation 6.3.

Using random forests to estimate the number of customers falling in the three mean deposits categories, we were able to produce a model that allows estimation of total deposits value of a collection of customers, using just three predictor variables - brand, country, and duration. For the test sample consisting of 18,896 customers, our estimate of total deposits (€24.6 million) exceeded the true total deposits (€22.5 million) by just about 9%. While this model is not able to address the problem of estimating CLV (represented as total deposits) of individual customers based on a set of features, it is useful for predicting the sum of total deposits value from a group of customers. For macro-level planning, such a model may be useful.

7.2 Limitations and future work

The choice of 'sum_deposits' to represent CLV could be reconsidered to improve further models. 'sum_deposits' is highly dependent on the duration of time that a customer has been active at the time that the data is sampled. For example, according to our definition of CLV as 'sum_deposits', a customer that signed up a few days ago but deposits large amounts could have a similar CLV to a customer that has been active for a year but deposits small amounts. If instead we took data for both customers over a year of activity, we would likely see that the high-depositing customer is of greater value to the betting operator. It is therefore suggested that in further work, a method of normalising 'sum_deposits' should be developed to account for the duration that a customer has been active with the betting operator.

8 List of Acronyms

CLV Customer Lifetime Value

EDA Exploratory Data Analysis

FTD First Time Depositor

KNN K-Nearest Neighbor

MAR Missing At Random

References

[1] Zahriah Binti Sahri and Rubiyah Binti Yusof [*Support Vector Machine-Based Fault Diagnosis of Power Transformer Using k Nearest-Neighbor Imputed DGA Dataset*]. Journal of Computer and Communications, 2:22-31, 2014

[2] Robert Kabacoff [*R in action: Data analysis and graphics with R*]. https://www.manning.com/books/r-in-action Manning, 2011

[3] Krishnan Bhaskaran and Liam Smeeth [*What is the difference between missing completely at random and missing at random*]. International Journal of Epidemiology, 43(4):1336–1339, 2014

[4] R Core Team [*R: A Language and Environment for Statistical Computing*]. R Foundation for Statistical Computing: Vienna, Austria (2019).

[5] Wei-Yin Loh [*Improving the precision of classification trees*]. The Annals of Applied Statistics, 3(4):1710-1737, 2009

[6] Jie Cai, Jiawei Luo, Shulin Wang and Sheng Yang [*Feature selection in machine learning: A new perspective*]. Neurocomputing, 300:70-79, 2018

[7] R Documentation, [*Feature Selection With The Boruta Algorithm*]. https://www.rdocumentation.org/packages/Boruta/versions/7.0.0/topics/Boruta

[8] Huang, Zhexue [*Extensions to the K-Means Algorithm for Clustering Large Data Sets with Categorical Values*]. Data Mining and Knowledge Discovery 2.3 (1998): 283-304.

[9] Cheema Jehanzeb [*Some General Guidelines for Choosing Missing Data Handling Methods in Educational Research*]. Journal of Modern Applied Statistical Methods, 13(2):53-75, 2014

[10] Smarkets, [*Identifying a Problem Gambler*]. https://help.smarkets.com/hc/en-gb/articles/212653605-Identifying-a-problem-gambler

[11] Paul Delfabbro, Anna Thomas and Andrew Armstrong [*Observable indicators and behaviors for the identification of problem gamblers in venue environments*]. Journal of Behavioral Addictions, 5(3):419-428, 2016