

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ADMINISTRAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ADMINISTRAÇÃO

Guilherme Brandelli Bucco

DEVELOPMENT OF A STOCHASTIC MODEL TO
ESTIMATE CUSTOMER VALUE

Porto Alegre

2019

Guilherme Brandelli Bucco

**DEVELOPMENT OF A STOCHASTIC MODEL TO
ESTIMATE CUSTOMER VALUE**

Tese apresentada ao Programa de
Pós-Graduação em Administração da
Universidade Federal do Rio Grande
do Sul como requisito parcial para
obtenção do título de Doutor em Ad-
ministração

Supervisor: Prof. PhD. João Luiz Becker

Porto Alegre

2019

CIP - Catalogação na Publicação

Bucco, Guilherme Brandelli
Development of a Stochastic Model to Estimate
Customer Value / Guilherme Brandelli Bucco. -- 2019.
73 f.
Orientador: João Luiz Becker.

Tese (Doutorado) -- Universidade Federal do Rio
Grande do Sul, Escola de Administração, Programa de
Pós-Graduação em Administração, Porto Alegre, BR-RS,
2019.

1. customer lifetime value. 2. customer-base
analysis. 3. customer-base risk. 4. probabilistic
models. I. Becker, João Luiz, orient. II. Título.



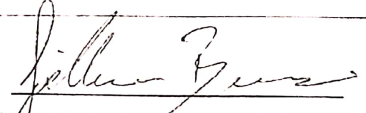

ATA DE DEFESA DE TESE

Aos 3 dias do mês de janeiro do ano de 2019, às 16h, na sala 202 da Escola de Administração da UFRGS, reuniu-se em ato público a Banca Examinadora de Tese de Doutorado do(a) aluno(a) Guilherme Brandelli Bucco, orientado pelo(a) Prof(a). Dr(a). João Luiz Becker (PPGA/UFRGS) e composta pelos professores examinadores abaixo relacionados, ocasião em que se realizou a arguição da tese intitulada “**Development of a stochastic model to estimate customers lifetime value**”. Concluídos os trabalhos, foram atribuídos os seguintes conceitos definitivos:

NOME	CONCEITO
Prof(a). Dr(a). Tiago Pascoal Filomena (UFRGS)	<input checked="" type="checkbox"/> Aprovado <input type="checkbox"/> Não Aprovado
Prof(a). Dr(a). Guilherme Liberali Neto (Erasmus University)	<input checked="" type="checkbox"/> Aprovado <input type="checkbox"/> Não Aprovado
Prof(a). Dr(a). José Afonso Mazzon (USP)	<input checked="" type="checkbox"/> Aprovado <input type="checkbox"/> Não Aprovado

Em anexo os pareceres individuais dos avaliadores.

OBSERVAÇÃO: Após o ato público da defesa da tese este documento deve ser encaminhado para a secretaria acadêmica a fim de ser feita a verificação do cumprimento dos requisitos regimentais pelo aluno e encaminhamento para homologação pela Comissão de Pós-Graduação. Assim sendo, esta ata não pode ser caracterizada como instrumento final do processo de concessão do título de doutor.

 Aluno(a): Guilherme Brandelli Bucco	 Prof(a). Dr(a). João Luiz Becker Presidente da Banca Examinadora
--	---

ACKNOWLEDGEMENTS

Agradeço aos meus pais, Maria Lúcia e Jorge Luiz, e à minha irmã Larissa, por estarem ao meu lado desde muito antes do início da minha trajetória acadêmica. A influência de vocês foi determinante. Sem vocês eu não teria chegado perto de onde cheguei. Agradeço pelo constante apoio, paciência sem fim e amor incondicional. Vocês tornaram essa caminhada muito mais tranquila. Esse trabalho também é de vocês.

Agradeço ao meu orientador, prof. Dr. João Luiz Becker, pela imensa paciência durante o longo processo de doutoramento. Agradeço por compartilhar generosamente a sua sabedoria e conselhos, por me apoiar e me ajudar a manter a calma nas épocas de estresse mais acentuado. Sua presença foi fundamental para que eu chegasse aqui. Agradeço também ao prof. Dr. Gui Liberali, que me supervisionou durante minha estadia na Erasmus University. Muito obrigado pelo conhecimento compartilhado, por me fazer sentir em casa e pelos ensinamentos sobre a academia, os quais levarei comigo sempre.

Agradeço aos membros da banca, prof. Dr. Gui Liberali, prof. Dr. José Afonso Mazzon e prof. Dr. Tiago Pascoal Filomena. Muito obrigado pelas inestimáveis contribuições para o aperfeiçoamento desta tese, e pela sua inspiração.

Agradeço à Universidade Federal do Rio Grande do Sul pelo ensino de qualidade, e aos meus professores no PPGA, em especial aos da área de Pesquisa Operacional: Dr. Denis Borenstein, Dra. Denise Lindstrom Bandeira, Dr. Tiago Pascoal Filomena e Dr. João Luiz Becker. Sou imensamente grato pelos seus ensinamentos.

Agradeço à minha turma do doutorado e aos amigos que fiz durante essa trajetória. Em especial, agradeço ao Pablo Guedes, Leonardo Sant'anna, Lucas Casagrande, Patrícia Tometich, Guillermo Cruz e Tito Grillo. Os momentos de descontração tornaram a caminhada mais leve. Agradeço também à Carol Dalla Chiesa e ao Marcus Bonugli, que tornaram a minha estadia em Rotterdam ainda mais estimulante e proveitosa. Muito obrigado, acima de tudo, pela amizade. Agradeço em especial à Marina, pelo apoio e paciência nos momentos finais.

Agradeço, por fim, à Capes pela ajuda financeira por meio da bolsa de estudos.

ABSTRACT

Companies have become increasingly consumer-oriented, aligning their around them. Faced with the potential source of consumer generated revenue, managers should seek to increase the value of their customer base. Customer relationship management provides companies with the means to do this. Customer lifetime value (CLV) is, according to Kumar and Shah (2009), a metric aimed to predict future cash flows that each consumer can provide to the company. This thesis aims to develop a stochastic model to estimate CLV. We found in the literature related to CLV a variety of methods, built to suit the specificities of the relationship between companies and their clients. In order to better situate the present thesis, we used the classification proposed by Fader and Hardie (2009), between contractual and non-contractual relationships. Besides this, the model fits in the classification proposed by Gupta et al. (2006) as a probabilistic model. The selected probabilistic model for this work is the renewal reward process. In order to perform the estimation in the non-contractual setting, we selected variables interpurchase and ticket value. For the contractual case, we seek to identify patterns in this type of relationship through the number of transactions and the total value spent within pre-defined periods. We do not assign probability distributions *a priori* for such variables, since we understand that, in order to obtain better CLV estimates, the model should represent the consumer behavior as best as possible. Thus, the distribution family chosen for each variable depends on the specific application. We also consider the possibility that client is active or inactive in a given period, either for having interrupted, temporarily or permanently, the relationship with the company, or for because he died. Finally, we consider the correlation between clients, so that the construction of a customers portfolio, based on the present model estimates, consider the systemic risk of the base. We developed a method which allows the inclusion of the correlation between clients in the model solution. It wasn't possible to find an analytical solution to the model. So, we proposed a solution based on discrete events simulation. We performed tests with two real world datasets, representing the two types of relationships. It was possible to estimate the distributions of CLVs for each client individually, considering the aforementioned characteristics.

Keywords: customer lifetime value; customer-base analysis; customer-base risk; probabilistic models.

RESUMO

As empresas têm se cada vez mais voltadas para o consumidor, alinhando as suas ações em torno deles. Diante da potencial fonte de geração de receita dos consumidores, os gestores devem buscar aumentar o valor da sua base de clientes. A gestão do relacionamento com clientes provê às empresas meios para isso, dentre eles, o *customer lifetime value* (CLV). De acordo com Kumar e Shah (2009), CLV trata-se de prever os fluxos de caixa futuros que cada consumidor pode prover. Diante disso, esta tese tem como objetivo desenvolver um modelo estocástico para estimativa do CLV. Encontramos na literatura relacionada a CLV uma diversidade de métodos, construídos de modo a se adequar melhor às especificidades da relação entre empresas e seus clientes. No intuito de melhor situar a presente tese, recorreremos à classificação proposta por Fader e Hardie (2009), entre relacionamentos contratuais e não-contratuais. Além desta, o modelo se enquadra na classificação proposta por Gupta et al. (2006) como modelo probabilístico. O modelo probabilístico selecionado para este trabalho é o processo estocástico de renovação com recompensa. Para efetuar as estimativas de CLV em relacionamentos não-contratuais, selecionamos as variáveis intervalo entre compras e valor da compra. Para o caso contratual, buscamos identificar padrões nesse tipo de relacionamento por meio da quantidade de transações e o valor total gasto dentro de períodos pré-definidos. Não atribuímos distribuições de probabilidade *a priori* para tais variáveis, pois entendemos que, para que se obtenha melhores estimativas de CLV, o modelo deve representar o comportamento do consumidor de maneira mais fidedigna possível. Assim, a escolha da família de distribuições para cada variável depende da aplicação específica. Consideramos também probabilidades de o cliente estar ativo ou inativo em um dado período, seja por ter interrompido, temporariamente ou permanentemente, a relação com a empresa, seja por ter falecido. Por fim, consideramos a correlação entre os clientes presentes na base, de modo que a construção de um portfólio de clientes baseada nas estimativas do presente modelo considerem o risco sistêmico da base. Desenvolvemos um método que permite a inclusão da correlação entre clientes na solução do modelo. Não foi possível encontrar solução analítica para o modelo. Assim, propusemos uma solução baseada em simulação de eventos discretos. Efetuamos testes com duas bases de dados reais, representando os dois tipos de relacionamentos considerados. Foi possível estimar as distribuições de CLVs para cada cliente individualmente, considerando as características mencionadas.

Palavras-chave: customer lifetime value; análise de base de clientes; risco em base de clientes; modelos probabilísticos.

LIST OF FIGURES

Figure 1 – Purchase behavior of a single customer	11
Figure 2 – Customer-company relationship classification	12
Figure 3 – Renewal process	20
Figure 4 – Proposed solution method for the model	31
Figure 5 – Convergence run time for the bootstrapping procedure to generate the correlation matrix	35
Figure 6 – Correlations matrix heat map for the groceries data set	35
Figure 7 – CLV distribution histograms for a few customers	37
Figure 8 – Number of runs needed for convergence with different treatments, for the non-contractual setting	38
Figure 9 – Convergence run time for different treatments, for the non-contractual setting	39
Figure 10 – Convergence run time for different numbers of customers, for the non-contractual setting	39
Figure 11 – Correlations matrix heat map for the credit card data set	41
Figure 12 – CLV distribution histograms for a few customers	43
Figure 13 – Number of runs needed for convergence with different treatments, for the contractual setting	43
Figure 14 – Convergence run time for different treatments, for the contractual setting	44
Figure 15 – Convergence run time for different numbers of customers, for the contractual setting	45

LIST OF TABLES

Table 1 – Some CLV models found in the literature	19
Table 4 – Transactions descriptive statistics from a groceries store	33
Table 5 – Running time for the correlations bootstrapping procedure with groceries data	34
Table 6 – Running time and number of runs for the convergence of simulations with non-contractual data	36
Table 7 – Mean CLV and standard deviation for a few customers of the non- contractual data set	37
Table 8 – Transactions descriptive statistics from a credit card data set	40
Table 9 – Runtime and number of runs for the contractual setting simulations . .	42
Table 10 – Mean CLV and standard deviation for a few customers of the contractual data set	42

CONTENTS

1	INTRODUCTION	10
1.1	Objectives	11
2	THEORETICAL REVIEW	12
2.1	Customer value - related work	12
2.2	Customer Lifetime Value	16
2.3	Renewal Reward Process	19
2.4	Simulation Modeling and the Operations Research Method	23
3	MODEL DEVELOPMENT	26
3.1	Model Description	26
3.2	Proposed Solution	28
4	EMPIRICAL ANALYSIS	33
4.1	Non-contractual Setting	33
4.2	Contractual setting	40
5	CONCLUSIONS	46
	BIBLIOGRAPHY	48
	APPENDIX A – FUNCTIONS USED TO SIMULATE THE CLV MODEL WITH THE NON-CONTRACTUAL SETTING	51
	APPENDIX B – FUNCTIONS USED TO SIMULATE THE CLV MODEL WITH THE CONTRACTUAL SETTING	63

1 INTRODUCTION

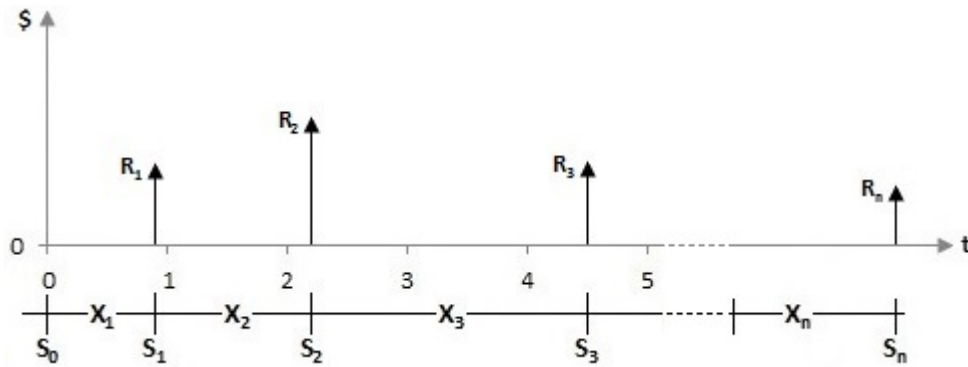
Companies have become more customer-centric (KUMAR; SHAH, 2009), aligning their actions around their current and potential clients. Faced with the revenue generation source of customers, marketing managers should seek to increase the value of the customer base. This can be achieved by several ways, either by attracting new customers, retaining existing ones while reducing the costs of keeping within the firm, and increasing sales. In any case, the estimation of expected earnings from each customer is an important issue.

Customer relationship management (CRM) provides companies the opportunity to contact the right customer at the right time and through the appropriate marketing contact (NESLIN et al., 2013). Since it is the customer who makes a significant contribution to corporate revenue, the prediction of those who can generate the best return on marketing contact expenditures becomes central to CRM. In a seminal paper, Bell et al. (2002) note in their review on the paradigm of customer value management that consumers should be seen as assets, and since the time of the publication of their paper, a literature aimed at customer value measurement was developed.

Customer Lifetime Value (CLV) is a metric aimed to help companies manage their customer base. According to Kumar and Shah (2009), it seeks to predict the future cash flows of each customer incorporating, in a single equation, the elements of revenue, expenditure and behavior that guides the customers profitability. Such flows are then discounted from capital costs in order to obtain the net present value of a customer expected future cash flows. The customer lifetime value, or present value of the expected future cash flows of a given consumer, is the key metric in CRM (ROMERO; LANS; WIERENGA, 2013).

Figure 1 illustrates a typical purchase behavior of a customer in which each of her transaction is depicted in a time frame. In the figure, R_n represents the contribution margins generated by the clients, S_n the instant of occurrence of the n^{th} transaction, and X_n the interpurchase time between the $n - 1^{th}$ and the n^{th} transactions. Most applications that can be found in the literature define a finite horizon, like 36 months in Kumar and Shah (2009), to predict future cash flows from customers. However, we argue that if we are to estimate the *actual* lifetime value of a given customer, we must consider the whole duration of the relationship between customer and companies.

Figure 1 – Purchase behavior of a single customer



Source: the author.

The problem considered here is therefore to identify a model that allows an adequate representation of this behavior so that one can obtain good predictions of the future cash flows that each client can provide to the company, and thus, making it possible to estimate a real *lifetime* value for each customer. The goal of this dissertation is to contribute to the body of research on CLV by developing a stochastic model aimed to estimate CLV in two different customer-company settings.

1.1 OBJECTIVES

The main goal of this dissertation is to develop a stochastic model to estimate CLV. The specific objectives are:

1. determine the relevant variables to the individual CLV measurement, adjustable to the type of company, product and service;
2. estimate a probabilistic model that best fit the variables for the individual CLV estimation, according to the setting;
3. estimate a model that consider the possible correlations between customers, in order to obtain a good risk assessment of the clients base;
4. select the right stochastic model to accommodate the selected variables;
5. illustrate the model using real-world data.

2 THEORETICAL REVIEW

In this chapter we present the aspects related to the measurement of customer value. Classifications of the relationship between customers and companies are discussed, and how the models represent such relationships. Then, characteristics of the problem of measuring customer value are presented. The theories and methods that support the proposed solutions are discussed in the final two section.

2.1 CUSTOMER VALUE - RELATED WORK

Fader and Hardie (2009) note the importance for companies of using the best methods to estimate the customer value, based on the different types of business environments in which they operate. Gupta et al. (2006) point out that it is possible that the approach used to model CLV is context-dependent. Following these recommendations, in this section we describe nature of the relationship between customers and companies and the types of transactions opportunities, according to the literature, in order to identify in which situations the proposed models can be generalized.

We found in the related literature, two main ‘axes’ that differentiate the relationship between customers and companies that affects customer value measurement: (1) relationship type (contractual and non-contractual), and (2) transaction opportunities (discrete and continuous). Figure 2 presents examples of products and services classified according to these criteria. We now will discuss them.

Figure 2 – Customer-company relationship classification

		Relationship type	
		Contractual	Non-contractual
Transactions opportunities	Discrete	Insurance Subscriptions	Conferences
	Continuous	Mobile phone Banking services	Office supplies Groceries

Source: Fader and Hardie (2009).

The existence of any formal contract guiding the relationship between companies and consumers defines, among other things, the duration of this relationship, the prices charged and possibly the period of payment due. Because of the implied impact in predicting the expected revenues from consumers, customer value research classify such relationships as “contractual” and “non-contractual”. Our model differs from those found

in the literature, in terms of such classification, by including a set of variables adjustable to the specific relationship type.

Bolton (1998) investigated the role of consumer satisfaction in the relationship duration with continuous service providers, in mobile industry, that is, in a contractual relationship. An important point made by the author is that small increments in customer retention rates (increases in the relationship duration) have a considerable effect on companies revenues, because customer retention costs are lower than the costs of acquiring new ones. In the case studied by the authors, the contractual setting, it was confirmed the proposition that satisfaction directly affects the relationship duration.

Even if in a contracted service situation the consumer intention is clearly expressed through the service subscription, such as when opening a bank account or when hiring a telephone service, customer desertion is always possible. Wirtz et al. (2014) contrasted the intention to change service provider and the change act itself. Variables such as price, quality and monetary costs of change are overvalued in relation to the act of change. Conversely, non-monetary costs of change (such as the effort to cancel the contract and search for a new supplier) are undervalued. The authors observed that, having identified the drivers of provider change behavior, it is possible to reduce customer desertion, so that costs associated with the acquisition of new clients (to compensate for desertions) are reduced, and two desirable characteristics of the contractual environment are maintained – greater predictability of relationship duration and revenues.

Fader and Hardie (2009) note that, in a contractual environment, the typical issues of managerial concern are relate to the identification of customers who are most at risk of defection in the next period and how much longer those clients can be expected to remain doing business with the company. Moreover, they emphasize that the definitive characteristic of the contractual setting is that the desertion is observable by the company. Such characterizations are important, guiding the researchers in the use of models appropriate to the relationship setting studied.

The problem of retention and desertion was addressed by Fader and Hardie (2007), in a contractual setting. The authors defined retention rates, for a period t , as the proportion of active clients in the period $t - 1$ that remained active at the end of t . Assuming that the probability of the customer renewing the contract with the company is constant and equal to $1 - \theta$, then the probability that it remains client up to a time T can be modeled by a Geometric distribution:

$$P(T = t|\theta) = \theta(1 - \theta)^{t-1}.$$

The heterogeneity of clients, implying a θ parameter for each of them, was modeled by the Beta distribution:

$$f(\theta|\alpha, \beta) = \frac{\theta^{\alpha-1}(1 - \theta)^{\beta-1}}{B(\alpha, \beta)},$$

where $B(.,.)$ Is the Beta function. Given these two assumptions, the probability of a randomly chosen client renewing his contract until the time T is

$$P(T = t|\alpha, \beta) = \frac{B(\alpha + 1, \beta + t - 1)}{B(\alpha, \beta)}.$$

Such a model is called *shifted-beta-geometric distribution* and was used extensively in the contractual context to model the probability of desertion.

In the case of non-contractual relationships, the problems faced by modelers differ. In this case, companies should ensure that the relationship remains active, since consumers generally split their purchases between different companies (REINARTZ; KUMAR, 2000). According to Fader and Hardie (2009), the primary objectives in the non-contractual case are to differentiate consumers who have terminated their relationship with the firm from those who are only in a gap between purchases and make predictions about the amount of business that can be expect from each consumer in the future.

Fader and Hardie (2009) point out the NBD (negative binomial distribution) model, developed by Ehrenberg (1959), as the standard model for repetitive behavior in the non-contractual case. Such distribution was adjusted to the amount of items purchased.

One of the most important extensions of this model is Pareto/NBD (SCHMITTLEIN; MORRISON; COLOMBO, 1987). In this model, the authors sought to estimate the likelihood of customers being *active* – that is, of having the potential to make purchases with the company –, growth of the customer base, identification of customers with higher probability of being active and the expectation of future business with them. In this model, they assumed the following, as summarized by Gupta et al. (2006): (1) while active, the number of transactions is characterized by a Poisson distribution, (2) the heterogeneity in the transaction rate over consumers is adjusted to a gamma distribution, (3) the time a customer remains active is exponentially distributed, (4) the heterogeneity in the desertion rates are adjusted to a gamma distribution, and (5) the transaction and desertion rates vary across consumers. It is important to note that according to such a model, after a long period without purchases, it is considered that the customer deserted, that is, he will not buy again.

Reinartz and Kumar (2000) and Schmittlein and Peterson (1994) validated the Pareto/NBD model using data from a retailer and a company that markets office products, respectively. The second work proposed, in addition, an extension of the model, incorporating the financial volume handled by the clients. Each purchase ticket value was adjusted to a normal distribution, whose average parameter varied throughout the consumers, also distributed according to a normal distribution. Finally, it was assumed independence of this component in relation to transaction and retention rates.

Arguing about difficulties in implementing the Pareto/NBD model, Fader, Hardie and Lee (2005) developed the beta-geometric/NBD (BG/NBD) model. According to

the authors, the only difference in terms of the assumptions of the two models lies in the fact that the BG/NBD assumes that customer defects occur immediately after the last purchase, unlike the Pareto/NBD case, in which desertions can occur at any time. More specifically, it is assumed that (1) after a transaction, the client becomes inactive with probability distributed according to the *shifted geometric* distribution, and (2) the heterogeneity of that probability is beta distributed. The other model assumptions are identical to assumptions 1 and 2 of the Pareto/NBD model. The results found in the comparison between the two models, using data from compact disc (CD) retail, indicate an equivalence between the models. As a limitation, the authors indicate the need to extend the model, such as Pareto/NBD, to adjust the quantity purchased.

We found some relaxation of the interpurchase imposition being exponentially distributed, even in the non-contractual context. Allenby, Leone and Jen (1999) developed a dynamic interpurchase model, aimed to get an indication of when specific consumers are more likely to become less active. The interpurchase time was adjusted to a generalized gamma distribution, consumer heterogeneity was adjusted to a generalized inverse gamma distribution and the temporal variation in expected interpurchase time was adjusted by relating covariates in a multiplicative model. They tested the model with data from financial instruments broker (stocks, bonds, funds, etc.), using different parameterizations and distributions. They identified the characteristics of each client, to classify them into three possible states: super active, active and inactive.

We cite, in addition to this, the research of Wu and Chen (2000), which also relaxed such imposition. In this work, the authors used the Erlang distribution, a generalization of the exponential distribution, to adjust the interpurchase time. Following the modeling construction practice of the previous models, that is, to consider heterogeneity among clients, the authors adjusted the heterogeneity in the rate of purchases to a gamma distribution. When compared with the NBD and Pareto/NBD models and using panel data from a tea company, the model resulted in better predictive performance.

One of the problems associated with measuring customer value is determining the amount of business that customers make with the company, that is, the fraction of their consumption purchased from the company – *share-of-wallet* (SOW). Perkins-Munn et al. (2005) point out the difficulties of obtaining such information in many of the business categories. Thus, they sought to examine the relationship between satisfaction with various attributes of performance, repurchase and SOW. Samples were collected from companies, from two distinct industries: transportation companies that purchased trucks in the observed period, and pharmacists. As result, the authors observed that the repurchase behavior can be used as proxy for the SOW, and the repurchase can be explained by the probability of repurchase, general satisfaction and general efficacy.

Information about SOW was used, for example, by Kumar and Shah (2009). Such

information was useful to measure the vulnerability of expected future cash flows, since a high SOW customer will most likely buy back, will have high retention and high satisfaction, consonant with the findings of Perkins-Munn et al. (2005).

Given the diversity of characteristics inherent to the company-consumer relationship, several modifications were proposed in traditional models. Romero, Lans and Wierenga (2013) sought to develop a more generic stochastic model, which includes the incidence of purchases (i.e. whether or not a consumer will buy in a period) and the variables purchase frequency and monetary value. In addition, they incorporate: dynamic purchasing patterns, situations in which the purchase frequency and monetary value are related, and the possibility that the customer may retake the relationship with the firm after a period of inactivity. The transition between the different states of activity and inactivity was adjusted to a hidden Markov model, the purchase frequency follows a Poisson process, whose parameters are modeled by a modified, state-dependent distribution. The purchase value follows a gamma distribution, whose parameters were adjusted to a distribution of the same family. Finally, monetary value and frequency could be dependent, throughout the different states. Using two databases, the model performed better than all benchmark models.

Park, Park and Schweidel (2014) proposed a model that considers multiple product categories, in a non-contractual environment. The interpurchase time of each products in the shopping basket has been adjusted to an exponential distribution. The authors argue that for the consumer, the combination of categories in their shopping basket can be informative about the time until the next purchase, depending on the relationship nature between these categories.

It is possible to perceive, in the cited works, a concern with the probability estimation that a certain client is inactive. This is aligned with the fundamental distinction of the CLV problem in a non-contractual context.

2.2 CUSTOMER LIFETIME VALUE

We emphasize that the decision of which variables are used to predict the CLV of a given client depends on the modeler intention. In the cases we mentioned, in the non-contractual context, the modeling included transactional variables (instant of purchase and its value). The work of Kumar and Shah (2009) sought to establish the link between customer equity (CE, determined by the sum of all CLVs in the customer base) and the company's market capitalization. The objective is to identify the specific consumers drivers that influence the market value. Thus, CLV was specified as a function of transaction-specific variables (e.g., purchase number, recency of last purchase), and firmographic (e.g. industry type, number of employees, and annual revenue) and demographic variables. The

method used was a system of seemingly unrelated regression equations (SUR).

For cases where transactions occur on a discrete time basis in non-contractual relationships, we point to the Fader, Hardie and Shang (2010) model. The relationship treatment as occurring in discrete time may be due to different causes, according to the authors: (1) transactions can only occur at fixed regular intervals, (2) transactions may be associated with specific events, and (3) the organization may opt, for convenience, to treat the transactions occurrence in discrete time, although in fact occur in a continuum.

The objective of his work was to develop a model to predict future purchasing patterns, called beta-geometric/beta-Bernoulli (BG/BB). Assuming that the customer relationship at a given time is 'alive' or 'dead' with some probability (Bernoulli), and that such probability differs according to the dataset heterogeneity (to a Beta distribution), such assumptions lead to a beta-Bernoulli model. Further assuming that the relationship has ended at the beginning of a transaction opportunity with a certain probability (whose heterogeneity is also beta distributed), i.e. after a sequence of several purchase opportunities, Bernoulli distributed, the customer decides to terminate the relationship, then such an assumption leads to the beta-geometric model, since the occurrence of the first 'success' (desertion, in this case) in a Bernoulli experiment is distributed according to the geometric distribution. The authors were able to derive an analytical formula for the model and made a comparison with the Pareto/NBD model, from where it was possible to observe a better performance of the BG/BB model. We point out the fact that, although the model does not assume exponentially distributed intervals, the assumption of desertion used the distribution in the discrete case that has the same property: absence of memory. Such a supposition may be fragile in cases where the consumer's experience with the company has an influence over his future decision to consume again with the company.

In addition to the two classification axes discussed above, we highlight a differentiation presented by some authors in their models, namely *always-a-share* and *lost-for-good* clients. Dwyer (1997) states that in the first category includes customers who supply their needs by buying from multiple suppliers, adjusting their participation in each of them at their discretion. An example is the office supplies consumer. In the second category are the clients who normally establish long-term relationships with their suppliers, since there is a high exchange cost. Such clients generally seek, as the author claims, to solve more complex problems. If the customer terminates their relationship, such account is forfeited. The author points out telecommunication systems customers as an example.

Gupta et al. (2006) state that the modeling decision possibly depends on the context. The authors use as an example the cell phone industries and banks. In these cases, consumers usually maintain purchase with a single company, which must involve lost-for-good models. In other contexts, such as consumable, airline and B2B industries, consumers tend to purchase from more than one company, so always-a-share approaches

are more appropriate.

Berger and Nasr (1998) comment on such a classification, stating that retention models are best suited for lost-for-good situations, in which customers who buy back with the company are treated as new clients. In the always-a-share cases, the migration patterns are more appropriate. In this case, the last purchase recency is used to predict the probability of repurchase.

In addition to these classifications, we present the one produced by Gupta et al. (2006). In their review of CLV work, they identified a variety of models according to their approach: (1) RFM, in which groups of consumers are created based on three variables – recency, frequency and monetary value –, based on their previous purchases; (2) probabilistic, where observed behaviors are seen as the realization of an underlying stochastic process governed by behavioral characteristics and through which it is sought to describe and predict behavior, rather than explaining it through covariates; (3) econometric, using regression models, such as probit and logit; (4) persistence models, which use time series and dynamic models, using VAR, unit root tests and cointegration; (5) computer science models, using data mining and machine learning tools (neural networks, decision trees, CART, SVM); and, (6) models of growth and diffusion.

The methods diversity used for CLV estimation is large, as the authors pointed out above. This finding is in line with the recommendations made by Bechwati and Eshghi (2005), namely, that the general model for computing CLV depends on the expected cash flows pattern which, in turn, depends on the business nature: “it is not one size fits all” (BECHWATI; ESHGHI, 2005).

Faced with the number of CLV models that can be found in the literature, we selected some of them in Table 1, which we’ll discuss next.

Most CLV models, like the one presented in the seminal paper from (SCHMITTLEIN; MORRISON; COLOMBO, 1987), assume the exponential distribution for the interpurchases in a non-contractual, continuous setting. This choice is based on the easy analytical treatment it provides. Interpurchases adjusted to an exponential distribution implies a Poisson process. Our model differs in terms of not assuming specific distribution families for the random variables. Even if we cannot provide a closed form to the computation of expected CLV, nor to the CLV distribution, we provide a solution good enough for the managerial purposes.

Heterogeneity in transaction rates, in dropout rates and other inputs for the models constitute another characteristic in some models. In our case, we deal with heterogeneity by measuring lifetime of the individuals, in a disaggregate manner.

Berger and Nasr (1998) argue in favor of stipulating a finite length to the projection period, specially in some industries like high-technology, because looking beyond some

Table 1 – Some CLV models found in the literature

Reference	Suppositions
Schmittlein and Peterson (1994)	(1) Consider retention rates (exponential), (2) number of transactions in a given period, Poisson distributed, (3) heterogeneity in transaction rates as a gamma distribution, (4) heterogeneity in dropout gamma distributed, (5) independence between both rates, (6) purchases value normally distributed.
Berger and Nasr (1998)	(1) Finite length to projection period, (2) constant values to contribution margin, promotional costs and retention rates.
Kumar and Shah (2009)	(1) Finite length to projection period, (2) consider the influence of marketing contacts on customers expenditure, (3) model level of marketing contacts, probability of purchase and contribution margin by means of predictor variables.
Romero, Lans and Wierenga (2013)	(1) Allow purchase patterns to vary over time, (2) purchases occur in discrete periods (weeks), (3) while active, customers purchases follow a zero-truncated Poisson distribution, (4) when active, the monetary value of a customer is gamma distributed, (5) periods of active and non-active modeled by a partially-hidden Markov model.
Sunder, Kumar and Zhao (2016)	(1) Consider contribution margin from each brand separately, (2) quantity purchased dependent on demographic variables and seasonal effects.

Font: authors archives

threshold involves too much guesswork. We account for dropout rates, as well as actual dying, in such a way that defining a finite horizon is not only unnecessary, but arbitrary. Their model was developed for a non-contractual setting, with discrete transaction opportunities.

Finally, some models define the CLV relevant variables in terms of demographic variables. Our model does account for that, because we want to present a model that depends on a set of transactional variables that are, in almost all companies cases, readily available.

2.3 RENEWAL REWARD PROCESS

The assignment of a probability distribution to the interpurchase time allows us to model the problem of measuring the customer value using a renewal process. Figure 1, which illustrates the set of transactional variables associated with a particular client, can be interpreted as a rewarding renewal process model. This is the model of this dissertation, to bring a stochastic model tool to the arena of estimate CLV.

Back to figure 1, we call each of company visits that becomes a sale as an “event”,

for which we assign the random variable S_n . This variable corresponds to the instant of occurrence of the n^{th} event (n^{th} purchase, therefore). We consider also a sequence of independent, nonnegative random variables $\{X_n, n = 1, 2, \dots\}$ with a common distribution F . These variables represent the time interval between the $(n - 1)^{\text{th}}$ and the n^{th} events (hence, between purchases). Consider that $S_0 = 0$ corresponds to the beginning of a stochastic counting process and that the instant of occurrence of the n^{th} event can be calculated by summing the intervals between events up to n^{th} :

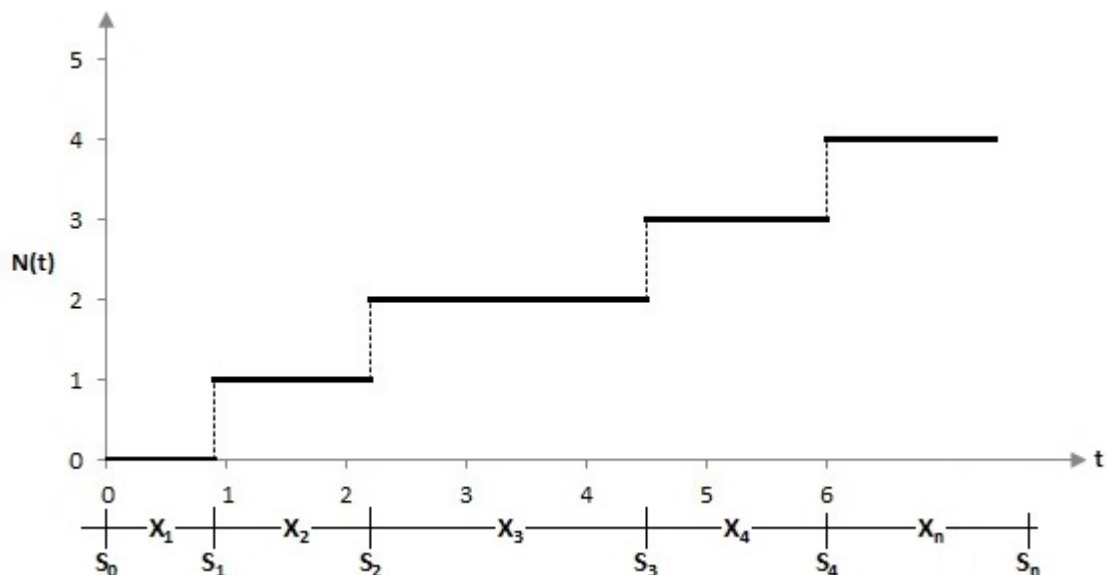
$$S_n = \sum_{i=1}^n X_i.$$

The number of events up to instant t equals n when the n^{th} event occurs before, or exactly, in t . Thus, we have that $N(t)$, the number of events occurring until t , is given by eq. 1.

$$N(t) = \sup\{n : S_n \leq t\} \quad (1)$$

The stochastic process whose occurrences count $\{N(t), t \geq 0\}$ is equal to eq. 1 is called a *renewal* process. Figure 3 illustrates such process. Since the time between events, also called *renewals*, are independent and identically distributed, with each renewal the process probabilistically restarts. In this section, we use the definitions of Ross (1996).

Figure 3 – Renewal process



Source: the author

The distribution of $N(t)$ can be obtained, according to Ross (1996), starting from the observation that the number of renewals up to a time t is greater than or equal to n if,

and only if, the n^{th} renewal occurs until time t . That is,

$$N(t) \geq n \Leftrightarrow S_n \leq t.$$

The probability of occurrence of n events up to time t will therefore be

$$P(N(t) = n) = P(N(t) \geq n) - P(N(t) \geq n + 1).$$

Since variables X_i , $i \geq 1$ are independent and have a common F distribution, we have that

$$S_n = \sum_{i=1}^n X_i$$

is distributed according to F_n , the n^{th} convolution of F with itself. Thus, the distribution of $N(t)$ can be obtained by means of eq. 2.

$$P(N(t) = n) = F_n(t) - F_{n+1}(t) \quad (2)$$

An important information about such processes is the expected number of renewals, which can be expressed in relation to the probability distribution of time between renewals: F . Define $m(t) = E[N(t)]$ a *renew function*. The relationship between the renewal function and F is given by eq. 3.

$$m(t) = \sum_{n=1}^{\infty} F_n(t) \quad (3)$$

As $n \rightarrow \infty$, the values of F_n tend to zero, as they refer to the distribution of infinite renewals within a finite time interval t . Thus, this function converges. This proof can be found in Ross (1996).

Suppose that with each renewal, a reward is received. Such a reward represents, in the modeling problem addressed in this dissertation, the purchase value. We denote R_n the reward received in the n^{th} renewal, and assume that R_n , $n \geq 1$ are independent and identically distributed, even though they may be dependent on X_n . Such a process is called the *renewal reward* process. The total reward received until time t is calculated, according to Ross (1996), by means of the eq. 4.

$$R(t) = \sum_{n=1}^{N(t)} R_n \quad (4)$$

We note that such an equation does not consider the differences in the monetary rewards over time. Because of this, an appropriate discount is required. Eq. 5 presents the present value of the total reward received until the time t , according to a discount rate i .

$$Z(t) = \sum_{n=1}^{N(t)} R_n e^{-iS_n} \quad (5)$$

The expected value of rewards up to a certain time t is dependent on both the distribution of each reward R_n , and the distribution between renewals, X_n . This value converges, when $t \rightarrow \infty$, to the expected reward value of each renewal divided by the expected time between renewals, as expressed by eq. 6, where $E[R] = E[R_n]$ e $E[X] = E[X_n]$.

$$\frac{E[R(t)]}{t} \rightarrow \frac{E[R]}{E[X]} \quad (6)$$

Again we are faced with a situation in which one does not consider reward differences in time. The deduction of discounted rewards expected value up to t , which we present, can be found in Serfozo (2009). Consider Y_n random variables. Its expected value conditioned to occurrence at time $S_n = s$ can be equated to a function $g(s)$, independent of n . That is, $E[Y_n|S_n = s] = g(s)$. The Eq. 7 presents the expected value of the sum of these variables Y_n . The proof is presented by Serfozo (2009).

$$E\left[\sum_{n=1}^{N(t)} Y_n\right] = \int_{(0,t]} g(s) dm(s), \quad (7)$$

where $m(s)$ is the renewal function (Eq. 3).

A variables exchange is performed:

$$Y_n = R_n e^{-iS_n}.$$

The variable Y_n will hence represent the value of the n^{th} present value reward. Its conditioned expected value is

$$E[Y_n|S_n = s] = E[R_n e^{-iS_n}|S_n = s] = e^{-is} E[R_n|S_n = s] = e^{-is} f(s).$$

By doing the substitution in eq. 5, we have that

$$Z(t) = \sum_{n=1}^{N(t)} Y_n.$$

Substituting this sum and the conditioned expected value of Y_n na Eq. 7, we have

$$E[Z(t)] = \int_{(0,t]} e^{-is} f(s) dm(s), \quad (8)$$

which is the present value of expected rewards up to t .

Eq. 8 lets us compute the expected customer's value for a given time t . Kumar and Shah (2009) stipulated a time horizon of 36 months for their model, justifying that this period offers a good trade-off between precision and prediction horizon in predicting customer value. However, it may be useful to estimate the customer value without having to arbitrate a value for t . Thus, we take the estimation horizon to infinity ($t \rightarrow \infty$), which

implies $N(t) \rightarrow \infty$. Although the number of renewals, $N(t)$, becomes infinite, the discount e^{-iS_n} applied to each of them causes the present value of renewals to converge to zero ¹, when $t \rightarrow \infty$. By means of this operation it is possible to estimate the expected value for which the customer's estimated cash flow converges, which we will denote as Z . Eq. 9 computes this result.

$$E[Z] = \lim_{t \rightarrow \infty} \int_0^t e^{-is} f(s) dm(s) \quad (9)$$

We propose a solution for this equation by means of simulation of the stochastic process. Law and Kelton (2000) note that analytical solutions for certain mathematical models can be extraordinarily complex, requiring vast computational resources. Very complex systems make it impossible to reach analytical solutions. In these situations, according to the authors, the models can be studied through simulation.

2.4 SIMULATION MODELING AND THE OPERATIONS RESEARCH METHOD

Simulation is a set of techniques for using computers to imitate or simulate, the operations of various kinds of real-world facilities or processes (LAW; KELTON, 2000; LAW, 2015). A set of *entities* that act and interact together toward the accomplishment of some logical end is called a system. The system has a set of *states*, a collection of variables necessary to describe it at a particular time.

In order to study the system scientifically, we have to make a set of assumptions about how it works. These assumptions, which take the form of mathematical or logical relationships, constitutes a *model*, used to try to gain some understanding of how the corresponding system behaves. If these relationships are simple enough, it may be possible to obtain exact information on questions of interest, the so-called *analytic* solution. However, most real-world systems, as the system presented in eqs. 5 and 9, are too complex to allow realistic models to be evaluated analytically, and these models must be studied by means of simulation. In a *simulation*, we use a computer to evaluate a model numerically. Data are gathered in order to estimate the desired true characteristics of the model.

According to the authors, for the simulation study purpose, a system can be categorized as discrete, in which the state variable changes instantaneously at separated points in time, and as continuous, in which the state variable changes continuously over time. A simulation can be categorized as *static*, where the system is represented at a particular time, or as *dynamic*, where the system represented evolves over time.

¹ The condition for this convergence is that $|i| < 1$.

The simulation method called *discrete-event simulation* concerns, according to Law and Kelton (2000), in modeling a system as it evolves over time by a representation where state variables change instantaneously at separate points in time, that is, the system can change at only a countable number of times. Those points in time are the ones in which events occur.

The system modeled in this work has a set of two entities: customers and companies. Back to figure 1, we can see that the system state variables evolve instantaneously in time. Therefore, this system can be studied by a discrete-event simulation. As we can see in the following Chapter, where the model will be presented, no analytical solution could be derived to it, due to the complexity of assumptions we made on the model. So, we have to resort on simulation.

Law and Kelton (2000) points out that the model description is just part of the overall effort to analyze complex systems. A typical simulation will be composed by several steps. The authors enumerate ten steps to design a sound simulation study. Firstly, the problem of interest must be formulated, and a plan for the study must be made. After that, data must be collected and the model must be defined. Data will be used to the model's parameters and to estimate input probability distributions. Regarding the model details, it should depend on project objectives, performance measures, data availability, time and money constraints, amongst others.

The conceptual model validity test will follow. This step is necessary to verify that the model assumptions are correct and complete. If the conceptual model is valid, then a computer program is constructed and verified.

Then, a series of pilot runs should be conducted, for validation purposes. After that, the model and an existing system, if one exists, are compared, in order to verify validity. Also, in this step, sensitivity analysis should be conducted, to determine what model factors have a significant impact on performance measures.

The seventh step is to design experiments. In this step, the length of each run is determined, the length of a warm up period, if any, and number of independent runs that facilitates construction of confidence intervals. After that, production runs should be made and its output analyzed. These steps are necessary to determine the performance of certain system configurations and to compare alternative system configurations. Finally, in the last step, the simulation model must be documented, presented and the results, used.

Together with the simulation modeling, which was used to construct the solution for the model, we present the usual phases of an Operations Research study, according to Hillier and Lieberman (2015). Firstly, we must define the problem of interest and gather relevant data. The problem defined in the present dissertation is to have good estimations of the customer lifetime value. We gathered data from two companies to test our model.

After that, a mathematical model must be formulated to represent the problem. In the present case, our model's mathematical representation is discussed in Chapter 1. After we have a well defined model, we develop a computer-based procedure for deriving solutions to the problem. The code used in the present dissertation is made available in Appendices A and B. The solutions we derived from it are presented in Chapter 4. The fourth phase is to test the model and refine it as needed. We conducted series of tests, which are presented and discussed also in Chapter 4. The fifth and sixth phases are to prepare for the ongoing application of the model as prescribed by management, and to implement it, respectively.

3 MODEL DEVELOPMENT

In this chapter, we present the stochastic model developed for the CLV estimation. In the first section, we define the notation that will be used throughout the chapter and present the equations for its computation. In the second section, we present the proposed solution for the model.

3.1 MODEL DESCRIPTION

Firstly, we define the random variables (r.v.) and parameters needed for the model development.

C	: Set of customers.	Parameter
CLV_c	: Lifetime value of customer c , $c \in C$, discounted to present value.	R.v.
S_{ic}	: Time of the i^{th} purchase made by customer c , $c \in C$.	R.v.
X_{ic}	: Time between the i^{th} and $i - 1^{th}$ purchases made by customer c , $c \in C$. $X_{ic} = S_{ic} - S_{(i-1)c}$.	R.v.
$F_{X_{ic}}$: Cumulative distribution function (CDF) of the variable X_{ic} .	Parameter
$m_c(t)$: Expected number of purchases from customer c , $c \in C$, until time t .	Parameter
R_{ic}	: Reward received from customer c , $c \in C$ in his i^{th} purchase.	R.v.
$F_{R_{ic}}$: CDF of the variable R_{ic} .	Parameter
$A_c(t)$: Binary variable representing the state of customer c , $c \in C$, at time t . $A_c(t) = \begin{cases} 1 & \text{if active at time } t, \\ 0 & \text{otherwise.} \end{cases}$	R.v.
$Pa_c(t)$: Probability of customer c , $c \in C$, being active at time t . $Pa_c(t) = P(A_c(t) = 1)$.	Parameter
$Ra_{ic}(t)$: Reward received from customer c , $c \in C$ on his i^{th} purchase, conditioned to being active at time t .	R.v.
r	: Discount rate.	Parameter
B	: Correlation matrix between variables X_c and R_c , $i \in C$.	Parameter
T	: Cholesky factor of B .	Parameter

Our CLV model can be used to estimate value in non-contractual and in contractual settings. For the contractual case, we need to define additional notation, as follows.

P	: Set of periods starting times.	Parameter
S_{ic}^B	: Beginning of the period of purchase i made by customer c , $c \in C$. $S_{ic}^B = \max\{p \in P p < S_{ic}\}$.	Parameter
S_{ic}^E	: End of the period of purchase i made by customer c , $c \in C$. $S_{ic}^E = \min\{p \in P p \geq S_{ic}\}$.	Parameter
d_{ic}	: Normalized time of the i^{th} purchase made by customer c , $c \in C$, normalized within the period it occurred. $d_{ic} = \frac{S_{ic} - S_{ic}^B}{S_{ic}^E - S_{ic}^B}$	R.v.
$F_{d_{ic}}$: CDF of the variable d_{ic} .	Parameter
Q_{cp}	: Number of purchases made by customer c , $c \in C$ in the period starting at p , $p \in P$.	R.v.
$F_{Q_{cp}}$: CDF of the variable Q_{cp} .	Parameter
W_{cp}	: Sum of rewards received from customer c , $c \in C$, in the period starting at p , $p \in P$.	R.v.
$F_{W_{cp}}$: CDF of the variable W_{cp} .	Parameter
y_{ic}	: Normalized reward of the i^{th} purchase made by customer c , $c \in C$. $y_{ic} = \frac{R_{ic}}{W_{cS_{ic}^B}}$.	R.v.
$F_{y_{ic}}$: CDF of the variable y_{ic} .	Parameter

We consider a counting process in continuous time, as the theory presented in Chapter 2. Some authors compute the lifetime value of a customer considering a future horizon of 36 months, as in Kumar and Shah (2009), justifying that such a period offers good trade-off between precision and prediction horizon in estimating the customer value. However, any predefined finite period (such as 36 months) is not only arbitrary, but do not consider the actual *lifetime* value, only a part of it. If we are to estimate the true lifetime value of a customer, we need to consider that the relationship with the company might end at some unknown point in time, which is captured by the active probability $Pa_c(t)$. This model parameter depends on the probability that the customer still intends to purchase with the company, and depends on the probability that he or she is actually alive.

So, an infinite horizon should be considered. Because of this, we must apply a discount rate r to each reward received from each customer, and consider that he or she can turn inactive at some time t .

The expected present value received from customer c is computed using eq. 10.

$$E[CLV_c] = \int_0^{\infty} e^{-rt} E[Ra_c(t)] dm_c(t), \quad (10)$$

The conditioned reward $Ra_{ic}(t)$ is computed with eq. 11.

$$Ra_c(t) = R_c A_c(t) \quad (11)$$

$$A_c(t) = \begin{cases} 1 & \text{if active at time } t, \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

Given that $A_c(t)$ has support in the $\{0, 1\}$ set, it is natural to suppose that it is Bernoulli distributed with parameter $Pa_c(t)$. Note that the expected value of $Ra_{ic}(t)$ might not be (and in some cases, is not) stationary. This is because $Pa_c(t)$ may depend on t . For example, the case in which we consider the customer's mortality to compute $Pa_c(t)$, as his or her mortality rate is age dependent.

For the computation of CLV_c distribution, we take eq. 5 on page 21 and substitute the notation to match the one defined in this section. Given that we want to consider the probability of customer c being active, we substitute the reward for the conditioned reward $Ra_{ic}(t)$. So, the distribution of CLV_c , conditioned on being active, is given by eq. 13.

$$CLV_c = \sum_{n=1}^{N(t)} Ra_c(t) e^{-rt} \quad (13)$$

We also consider that $CLV_c, \forall c \in C$, might be correlated with $CLV_d, \forall d \in C, c \neq d$. This is an important consideration when we advance to the portfolio optimization stage, when the portfolio risk can be assessed by the covariance between assets (customers, in the present case): $Cov(CLV_c, CLV_d) = \rho_{c,d} \sqrt{V(c)V(d)}$. So, if we know the distribution of CLV_c and we consider that two customers $c \in C$ and $d \in C$ are correlated with each other ($\rho_{c,d} \neq 0$), then the portfolio risk computation is straightforward.

Eq. 13 is the CLV distribution for each customer, considering its transactional variables distributions, the probability of being active and the correlation with the other customers. Therefore, this equation is central, for it contains all the information needed to compute all managerial relevant information. Having the distribution, we can compute the expected CLV, its variance, the customer-base risk, and also, the probability that the customer is still alive in a given moment in the future, amongst other interesting managerial applications.

3.2 PROPOSED SOLUTION

In this section we present a simulation-based solution to the CLV_c equation and to the model that considers the correlation between customers' variables.

We developed a model that is robust enough to (1) accommodate any distribution to the interpurchase times variable, as well as to the rewards; (2) consider that customers can have phases in which they are active (the supposition in this case is that the interpurchase times distribution completely describes such variable), and phases when they aren't; and, (3) consider that the customers purchase behavior (and, therefore, their CLV) is correlated.

As discussed in section 3.1, we consider that CLV_c might be correlated with CLV_d , $c, d \in C$, $c \neq d$. We don't know the closed form of the CLV_c distribution, so we have to resort on some other way to assess this correlation structure. Considering that CLV_c is mainly dependent on the interpurchase time and on the rewards, then the correlation between those two variables could give us a good approximation to $\rho_{c,d}$. So, we compute the correlation between interpurchases and rewards for each customer and between customers: (1) $\rho_{X_{ic}, R_{ic}} > 0, \forall c \in C$, (2) $\rho_{X_{ic}, X_{id}} > 0, \forall c, d \in C, \forall c \neq d$, (3) $\rho_{R_{ic}, R_{id}} > 0, \forall c, d \in C, \forall c \neq d$, and (4) $\rho_{X_{ic}, R_{id}} > 0, \forall c, d \in CI, \forall c \neq d$, and use the Cholesky decomposition to generate series of correlated random numbers. The Cholesky decomposition of a square matrix \mathbf{B} of order n consists in finding a lower triangular matrix \mathbf{T} of the order n , such that $\mathbf{B}=\mathbf{T}\mathbf{T}'$. The \mathbf{T} matrix is called the Cholesky factor. One important consideration is that the \mathbf{B} matrix must be positive definite or positive semi-definite.

In order to generate a sequence of correlated random numbers, we must follow this procedure: (1) estimate the correlation structure (a correlation matrix) we want to replicate, which we will call \mathbf{P} , (2) compute the Cholesky factor \mathbf{T} of the \mathbf{P} matrix, (3) generate series of uncorrelated random numbers, (4) normalize these series, (5) multiply these sequences by \mathbf{T}' , and (4) reverse the normalization. We used Cholesky decomposition because it is very straightforward and efficient algorithm. In fact, any generalized square root of \mathbf{P} would produce the same effect.

Every correlation matrix is at least positive semi-definite. In most applications, the empirical sequences of variables X_i and R_i , for two customers i and j , $i \neq j$, don't have the same length, that is, their numbers of purchases are different. To compute the correlation between that variables for those customers, some data from the longest sequence must be discarded. This poses a major problem in the estimation process, as we might lose a lot of valuable data.

We propose a procedure to circumvent such a problem based on bootstrapping. The pseudo-code of this procedure is presented in Algorithm 1. The summation of two positive semi-definite matrices, as depicted in Step 8 of this Algorithm, is also a positive semi-definite matrix. Therefore, it is guaranteed that the output of Algorithm 1 is also a positive semi-definite matrix.

Using Algorithm 1 and the Cholesky decomposition, the simulated CLV_c distributions will be based on randomly generated and correlated series, whose correlation

replicates the original structure found in data. This result is most important when the objective is to generate a customer's portfolio, because the customer-base risk (based on covariance) is readily available in the CLV_c distributions.

Algorithm 1 Bootstrapping procedure to generate a correlation matrix

Input: \mathbf{V} vector of n variables with distinct lengths.

Output: **AvgCor** correlation matrix.

Step 1: Compute the length of the largest vector in \mathbf{V} and assign it to m .

Step 2: Declare a m by n , $Rmat$ matrix.

Step 3: Estimate the CDF for each vector in \mathbf{V} .

Step 4: Populate $Rmat$ by column with each vector of \mathbf{V} combined with a zero vector of length equal to the difference between m and the length of the corresponding vector of \mathbf{V} .

Step 5: For each column of $Rmat$, generate a random sequence of length equal to the difference between m and the length of the corresponding vector of \mathbf{V} , from the corresponding CDF, and assign this sequence to the zeros part of the corresponding column in $Rmat$.

Step 6: Compute the correlation matrix of $Rmat$ and assign to $Cmat$.

Step 7: Assign $Cmat$ to $AvgCor$.

Step 8: Repeat Steps 5 and 6, compute the weighted average between $Cmat$ and $AvgCor$, assign it to $AvgCor$ and stop when reaches convergence of $AvgCor$.

Ross (1996) shows that the renewal function $m_c(t)$, eq. 3, can be calculated by means of the n^{th} convolution of the interpurchase times distribution function with itself, which corresponds to the sum of n random variables. Renewal functions of simple distributions, such as the exponential (which leads to the Poisson process), are available. More complex distributions don't have a known renewal function, and taking into account the complexity of the model, an analytical solution to eq. 13 could not be presented at the moment. We have to resort on a numerical approximation to find solutions to the proposed equation.

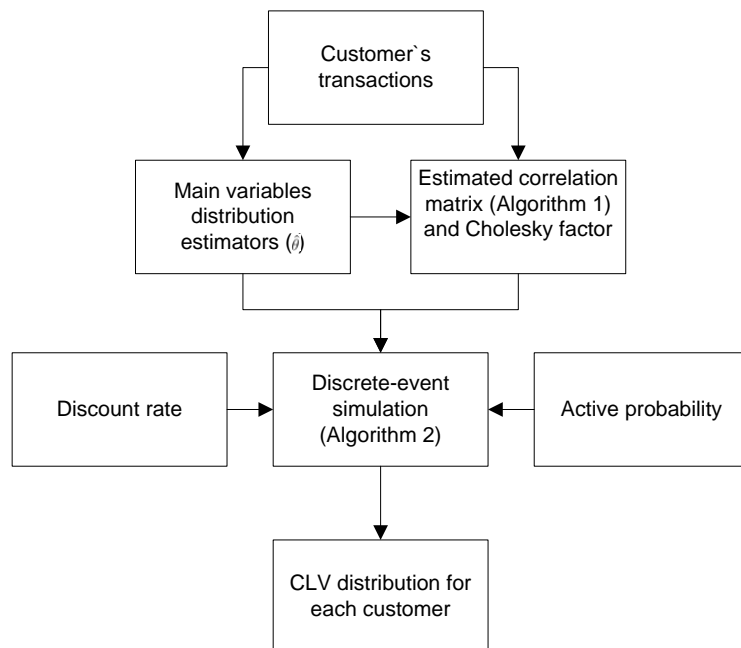
Now we present the simulation method proposed to solve eq. 13, according to the general procedure depicted in Figure 4. As was discussed in section 2.1, the relationship between customer and companies can be classified as non-contractual or contractual. In the non-contractual setting, companies don't observe when the customer has stopped purchasing with them, because there is no formal contractual instrument guiding the relationship. If we only have transactional data from customers, then interpurchase times and ticket values are the variables that can be used to identify her or his purchasing pattern. Therefore, for the non-contractual setting, variables X_{ic} and R_{ic} are the main variables described in Figure 4.

In the contractual case, companies know when the customer has finished the relationship with them. But more than that, there's a pattern in time with respect to contract-related expenditures. Normally, customers pay for the service monthly (credit cards bills, for example). Because of that, we argue that a pattern in the customer's usage behavior could emerge. In order to identify and benefit from such pattern, for the contractual setting, variables Q_{cp} , W_{cp} , d_{ic} , and y_{ic} are the main variables (see Figure 4).

The CLV_c is estimated by a finite sum of eq. 13. The discrete-event simulation method used in this framework is presented in Algorithm 2. This algorithm follows the notation presented bellow, in addition to the one presented in section 3.1.

- $\hat{\theta}$: Estimated vector of the main variables distribution parameters.
- $Loop_crit$: Convergence criteria for each simulation run.
- $Loop_crit$: Convergence criteria for the mean CLV_c , $\forall c \in C$.
- CLV_c : Vector of simulated CLV_c , $c \in C$, with length equal to the number of simulation runs.
- n_sim : Number of simulation runs.
- RND : Vector of randomly generated numbers.

Figure 4 – Proposed solution method for the model



Source: the author.

Algorithm 2 Discrete-event simulation method to estimate CLV_c

INPUT: $\hat{\theta}$, r , T , $Loop_crit$, Sim_crit .

OUTPUT: CLV_c

- 1: Initialize variables: CLV_c , $InnerConv_c$, n_sim
- 2: **repeat**
- 3: Initialize variables: RND , CLV_c , $CLVa_c$, $InnerConv_c$
- 4: **repeat**
- 5: RND = Random numbers for the main variables according to $\hat{\theta}$, $\forall c \in C$
- 6: Cholesky decomposition of RND with Cholesky factor T
- 7: Update S_{ic}
- 8: Compute $Pa_c(t)$, according to $\hat{\theta}$
- 9: Compute $Ra_{ic}(t)$
- 10: $InnerConv_c = \left(\frac{Ra_{ic}(t)}{CLV_c} < Loop_crit \right)$
- 11: Update CLV_c
- 12: **until** $Conv_c == \mathbf{true}$, $\forall c \in C$
- 13: $OuterConv_c = \left(\frac{mean(CLV_c)(n_sim-1)+CLV_c}{n_sim} < Sim_crit \right)$
- 14: Update CLV_c
- 15: **until** $OuterConv_c == \mathbf{true}$, $\forall c \in C$
- 16: **return** CLV_c

4 EMPIRICAL ANALYSIS

In this chapter, we present the empirical tests performed with the model. We run a set of tests with two data sets, representing two different relationships between company and customer, that is, non-contractual and contractual.

All tests were run using R programming language (R Core Team, 2018), except for the bootstrapping procedure described in Algorithm 1, which was coded in C++. The code used for the non-contractual setting is presented in Appendix A, and for the contractual setting, in Appendix B. The following sections shows the results.

4.1 NON-CONTRACTUAL SETTING

For the non-contractual setting, we used a set of transactions from a medium sized groceries store located in the southern region of Brazil. This transactions set comprised 72,664 tickets between 02/Jan/2014 and 31/Oct/2015, and 561 randomly chosen, loyal clients (i.e. clients that purchased using a credit scheme offered by the groceries store). Table 4 shows descriptive statistics of ticket values and interpurchase times.

Table 4 – Transactions descriptive statistics from a groceries store

	Ticket value (R\$)	Interpurchase times (days)
Mean	183.04	3.65
Standard deviation	1,276.00	3.47
Skewness	17.23	2.74
Kurtosis	365.51	12.11
Minimum	0.07	1
Maximum	48,244.09	43

Font: research data

We must emphasize the fact that the selected clients are considered to be loyal. So, it is quite possible that the purchases frequency and total expenditures are larger than in other situations. Also, we expect that the variables distributions are more stable, because the customers possibly are using a high fraction of their budgets on the particular groceries store, resulting in a smaller variability. Therefore, the results shown in this section cannot be generalized to other situations, although the model still is generalizable to different settings, companies, and industries.

Following the proposed solution for the model (Figure 4), firstly we defined the distributions for variables X_{ic} and R_{ic} . One of the main characteristics of our model is that it is robust enough to accommodate different distribution families for the variables. For this reason, it can be easily adapted to different kinds of companies e customers, working

on a non-contractual basis. All that is needed is the definition of the distributions that best describe the behavior of those variables.

For the application in this dissertation, we used the empirical distribution for both variables. After that, we followed Algorithm 1 to generate a correlation matrix between those variables, for all customers, and computed the Cholesky factor.

We evaluated the running time of the correlation matrix estimation, varying the number of customers (100, 200 and 400) and convergence criteria (0.1, 0.01 and 0.001) for this estimation. The running times for convergence are shown in Table 5.

Table 5 – Running time for the correlations bootstrapping procedure with groceries data

Number of customers	Convergence criteria	Running time
100	.1	.36
200	.1	.89
400	.1	3.58
100	.01	1.50
200	.01	3.50
400	.01	15.86
100	.001	12.80
200	.001	35.61
400	.001	134.13

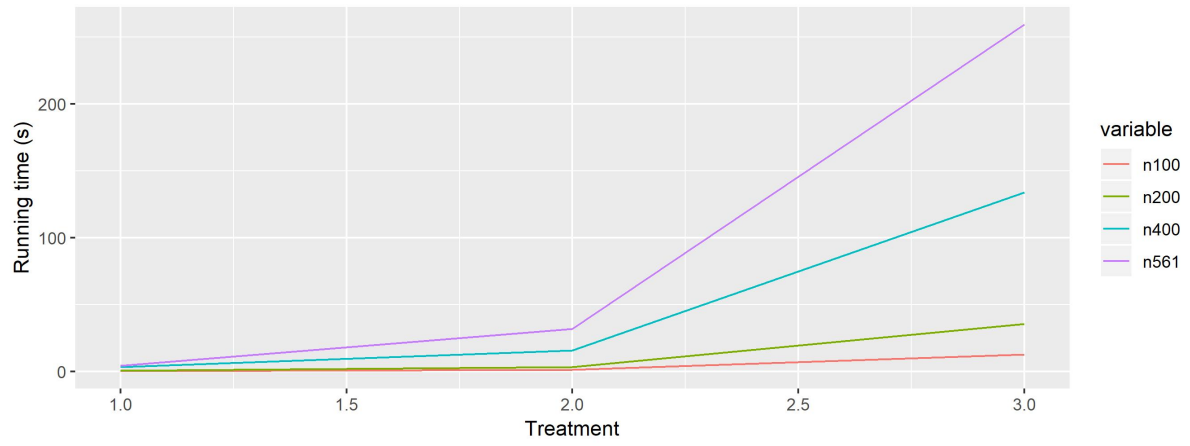
Source: the author.

The line graph in Figure 5 shows the growth in running time for convergence of Algorithm 1, based in the Table 5 parameters. The x-axis values “Treatment” in this graph are calculated as $-\log_{10}(\text{criteria})$. As we can see in Table 5 and Figure 5, the computation time needed for convergence grows very quickly. Since there are two variables for each customer, the matrix size is $(2n)^2$ (n : number of customers). Given the quick growth in computation time with respect to both number of customers and convergence criteria, the decision on which convergence criteria to choose lies in the trade off between the precision needed for the correlations and the computation time available.

The Pearson correlation coefficient between interpurchases and rewards, for each customer and between customers, varied from -0.1142 to 0.3972 (excluding correlations $\rho_{X_{ic}, X_{ic}}, \forall c \in C$ and $\rho_{R_{ic}, R_{ic}}, \forall c \in C$). In order to illustrate the distribution of correlations between those variables, a correlations matrix heat map is presented in Figure 6. The top right corner and the bottom left corner shows the lowest and highest correlations, respectively. As we can see, those variables are not strongly correlated. This could be due to the nature of the particular business (groceries store), to the specific data set we used, or even it is a general characteristic of the non-contractual setting. For that matter, we suggest further investigations, since the correlations structure between customers directly affect the construction of a customers portfolio.

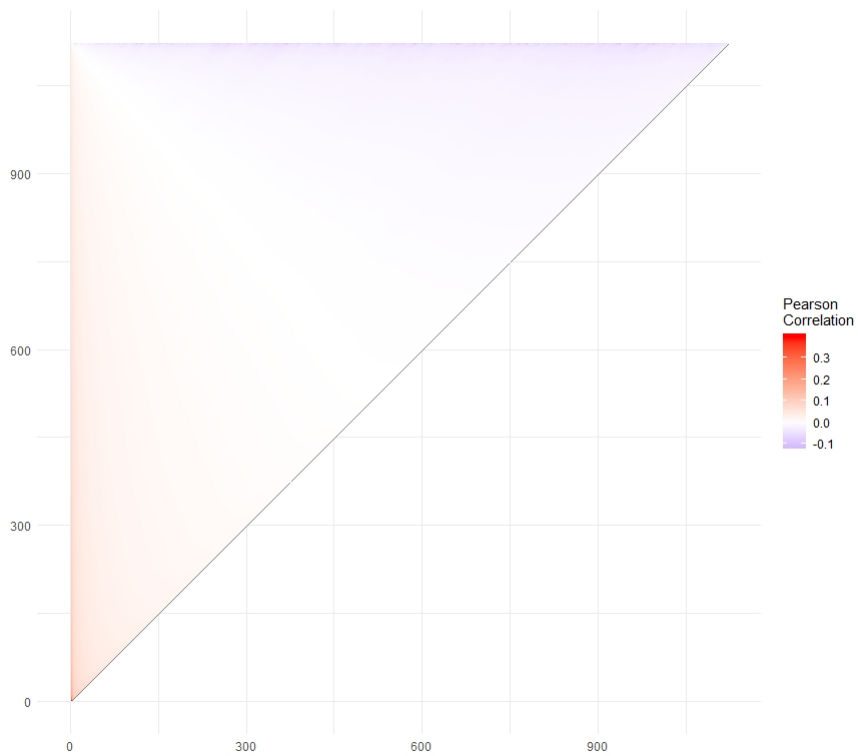
With this procedure, we replicate the variable’s correlation structure in the sim-

Figure 5 – Convergence run time for the bootstrapping procedure to generate the correlation matrix



Source: the author.

Figure 6 – Correlations matrix heat map for the groceries data set



Source: the author.

ulations. Thereby, we expect that each customer’s simulated CLV distribution will be correlated to each other customer’s CLV distributions, accordingly to the natural correlation structure found in the transactions data. The main managerial implication for this is that a customer’s portfolio based on these distributions will have a more realistic risk assessment.

Following the procedure to compute customers values, we proceeded to simulate eq. 13 using Algorithm 2. The annual discount rate was defined to 15%. Since we don’t have the actual data for churn rates and the customers ages (to compute their actual dying probability), we randomly attributed dates of birth to the customers and used the mortality tables published by IBGE (IBGE, 2015) to compute their dying probability. Variable $Pa_c(t)$ is, therefore, the complement of the dying probability for a given customer.

Table 6 shows the running times and number of runs needed for the simulations convergence.

Table 6 – Running time and number of runs for the convergence of simulations with non-contractual data

Number of customers	Active probability	Correlation structure	Convergence criteria	Run time (s)	Number of simulation runs
100	False	True	0.001	99.7	332
100	False	True	0.01	13.7	48
100	False	True	0.1	2.0	6
100	True	False	0.001	252.4	365
100	True	False	0.01	144.0	233
100	True	False	0.1	46.0	68
200	False	True	0.001	318.2	405
200	False	True	0.01	48.6	71
200	False	True	0.1	3.7	5
200	True	False	0.001	3688.0	1895
200	True	False	0.01	452.9	370
200	True	False	0.1	72.9	54
400	False	True	0.001	1417.8	511
400	False	True	0.01	397.5	170
400	False	True	0.1	29.9	11
400	True	False	0.001	3505.4	1397
400	True	False	0.01	829.0	326
400	True	False	0.1	334.4	115

Source: the author.

Beyond the correlation structure that is considered in the simulated CLVs, we also take into account the active probability for each customer. With this feature, we can incorporate both the churn behavior and the customer’s dying probabilities. Given the fact that we don’t necessarily need to stipulate an arbitrary, finite, simulation horizon, the relationship end is incorporated in a more realistic fashion. The main managerial implication is that the manager don’t need to decide a CLV estimation horizon for all customers. A compound active probability (considering both churn and dying probabilities) can be estimated based on actual data, which in turn, will handle the estimation horizon endogenously.

We selected ten customers simulated with the whole data set until convergence in 0.001, considering the correlation structure and don't considering the active probability. Their mean CLV and standard deviation (sd) from 482 independent runs are shown in Table 7.

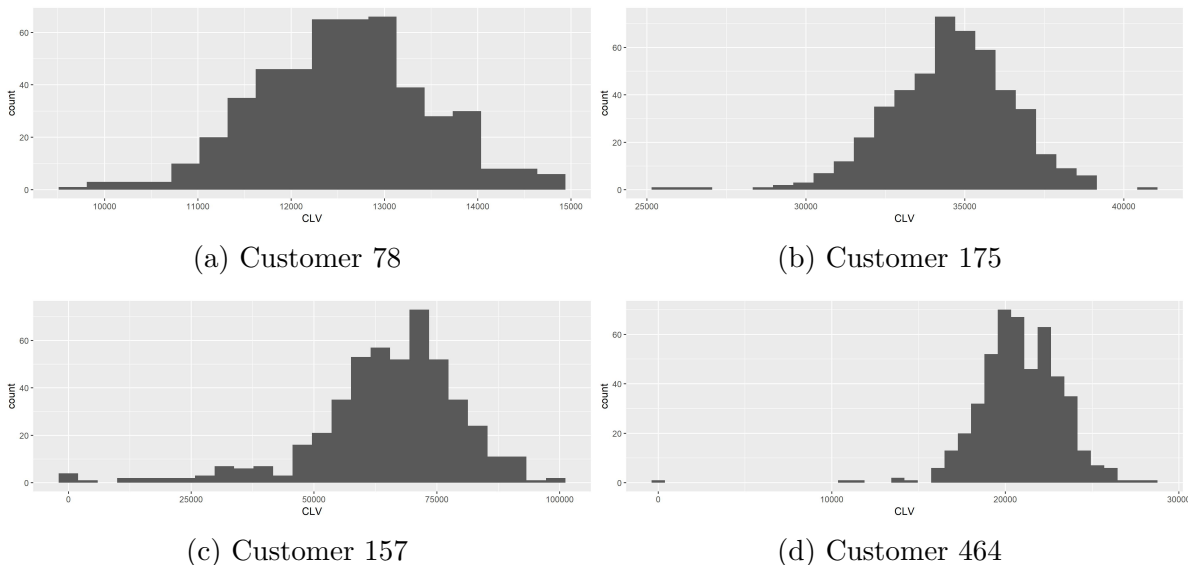
Table 7 – Mean CLV and standard deviation for a few customers of the non-contractual data set

Customer	Mean CLV	Sd
78	12554.6	908.3
175	34532.4	1960.2
157	65022.4	15406.4
464	20841.9	2479.6
12	44701.8	3231.5
120	88994.5	5724.6
268	9179.9	835.7
383	44015.1	5374.4
548	14792.6	2048.6
439	17428.1	921.3

Source: the author.

The distribution of simulated CLVs (482 independent runs) from the first four customers of Table 7 were plotted in histograms, as depicted in Figure 7.

Figure 7 – CLV distribution histograms for a few customers



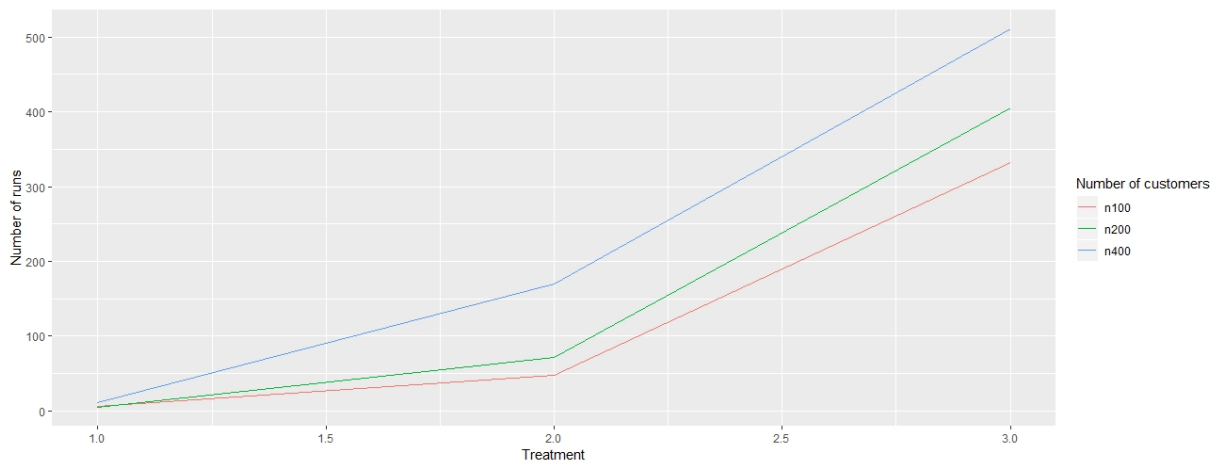
Source: the author

As we can see in the four graphs of Figure 7, these histograms have a shape similar to the Normal distribution. We ran a Shapiro-Wilk normality test with the CLV distribution for all 561 simulated customers. Of the 561 customers, 269 adjusted ($\alpha = .05$) to the Normal distribution (47.95%), which is a low proportion.

We tested for normality, but the simulated CLVs don't necessarily need to follow this family. In fact, there is no need to choose a theoretical family of distributions at all, because the simulated distribution contains enough information: the moments (expected value, variance, and so on) and the risk assessment (covariance between customers). So, one of the main contributions of our model is to present a CLV distribution that summarizes what we consider the most important features of customer's behavior: (1) interpurchase and ticket value distributions individually, (2) correlation between all customers, and (3) individual active probabilities.

Figure 8 depicts the number of runs needed for simulation convergence, according to the selected treatments, for the tested levels of numbers of customers. As we can see, the lines representing the number of customers are almost parallel, with a small difference, compared to the difference when we move to a more strict convergence criteria. It suggests that the number of customers don't affect the number of simulation runs as much as the convergence criteria.

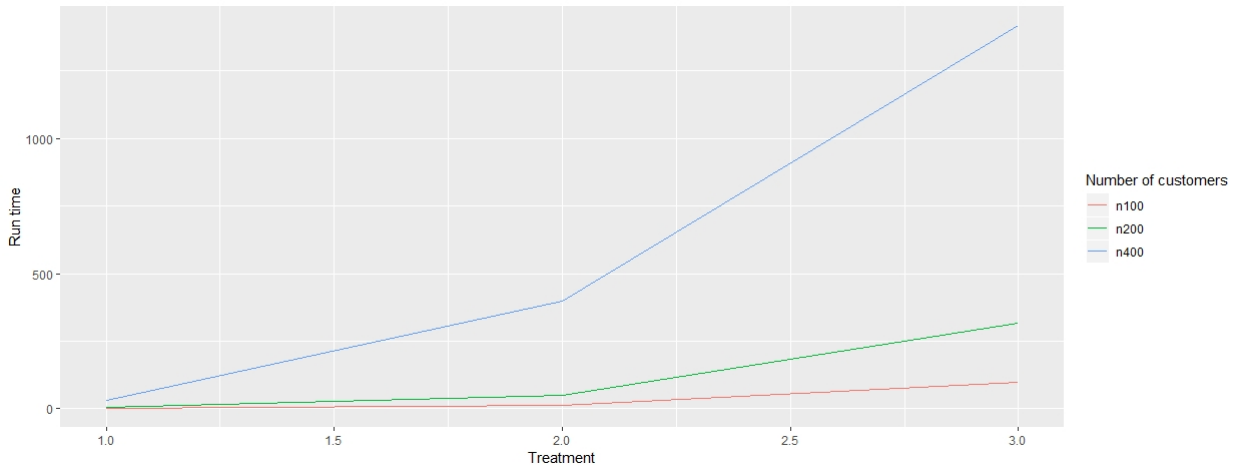
Figure 8 – Number of runs needed for convergence with different treatments, for the non-contractual setting



Source: the author.

Figure 9 shows the run time needed for the simulations convergence, according to each different convergence criteria (treatment), for the tested levels of numbers of customers. Even though the number of customers do not affect the number of runs as much as the convergence criteria, it does affect the runtime. The number of runs necessary for convergence of a particular c customer has little effect from the number of customers being simulated, but the data structures size increases significantly when we simulate more customers. So, the larger difference between the simulations with small number of customers (100 and 200) and the larger (400), in terms of run time, should be due to the growth in data structures.

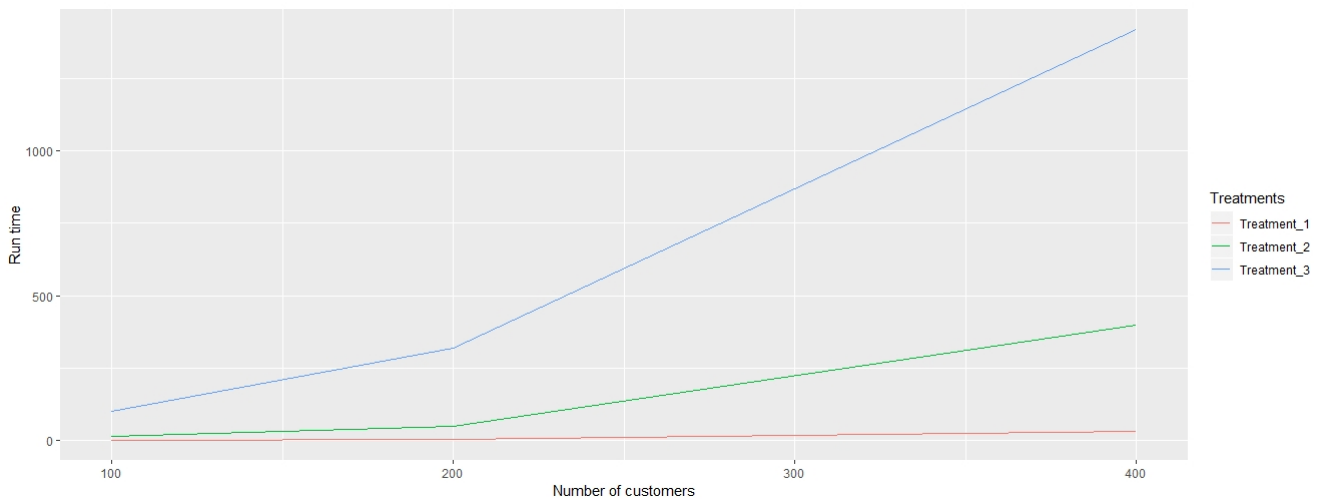
Figure 9 – Convergence run time for different treatments, for the non-contractual setting



Source: the author.

Figure 10 presents the run time needed for simulation convergence with respect to the number of customers, for the three different treatments. Again, we can see a quick growth in run time needed for convergence when increasing both number of customers and convergence criteria for the simulation.

Figure 10 – Convergence run time for different numbers of customers, for the non-contractual setting



Source: the author.

We devise a few guidelines to choose the convergence criterion. If the decision maker needs a very precise estimate of the mean, then a maximum width of the confidence interval (CI) could be a good criterion, which would guide the necessary number of runs to reduce the mean CI width to a desired value. Also, if the available time for simulation

is low, then a maximum run time could be chosen. We would not suggest to diminish our criterion (the difference between mean CLV_c of two consecutive simulation runs) to a more strict precision, because the mean CLV_c is rather high.

4.2 CONTRACTUAL SETTING

For the contractual setting, we used a set of transactions from a credit card company. This transactions set comprised 18,675 operations between 04/Jun/2012 and 06/Oct/2015, and 79 clients. Table 8 shows descriptive statistics of ticket values and interpurchase times.

Table 8 – Transactions descriptive statistics from a credit card data set

	Ticket value (R\$)	Interpurchase times (days)
Mean	78.94	3.04
Standard deviation	100.63	9.39
Skewness	3.71	12.37
Kurtosis	18.97	260.00
Minimum	0.29	0
Maximum	999.00	335

Font: authors archives

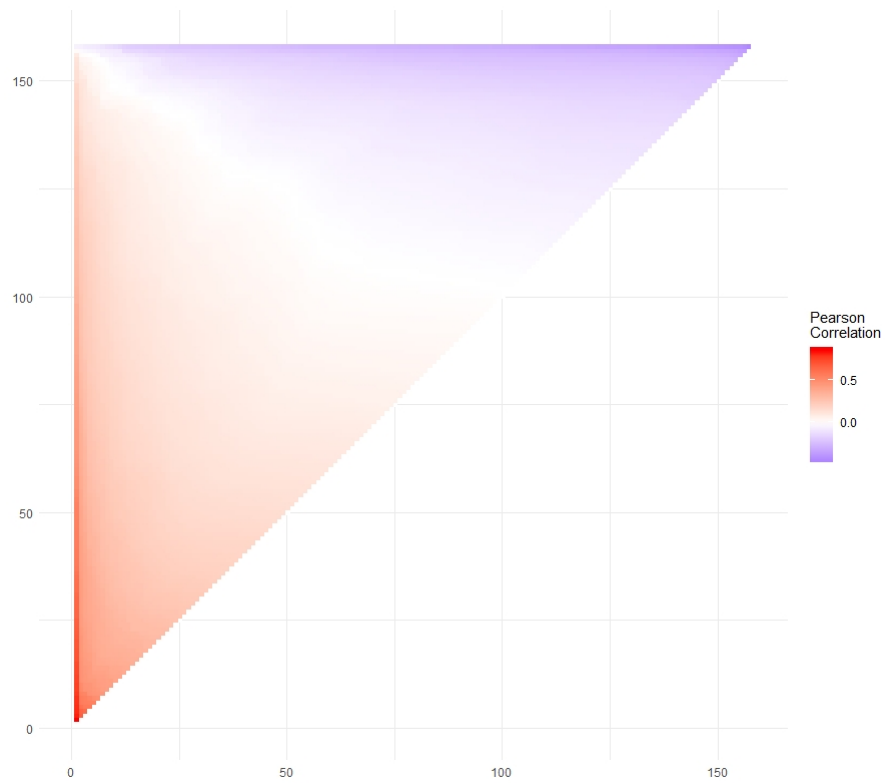
For the contractual case, the main variables chosen for the proposed solution (Figure 4) are: Q_{cp} , W_{cp} , d_{ic} , and y_{ic} . We tested the adjust of those variables to several distribution families. For variable Q_{cp} , we used the two-sided truncated Normal distribution; for variable W_{cp} , we used a one sided truncated (lower tail) Normal distribution; variable d_{ic} was adjusted to a beta distribution; and variable y_{ic} was adjusted to a log-normal distribution.

As was discussed in section 3.2, the contractual setting generally has a specific pattern regarding contract-related expenditures. In some cases, such pattern repeats monthly, which generally coincides with the customer’s salary. This rationale justifies the use of period-related variables, as the ones chosen.

Variable d_{ic} , and its CDF $F_{d_{ic}}$, contain the information regarding purchases distribution along each period. For most services, customers may split their usages among different service providers (e.g. customers may have more than one credit card). One important managerial impact of our model is that the distribution $F_{d_{ic}}$ may indicate whether the customer is splitting her or his usages between different providers. Since this variable has a support between 0 and 1, its CDF can naturally be adjusted to a beta distribution. If this distribution is not even (it’s not uniform), then probably the customer has more than one provider for the particular service.

We ran Algorithm 1 for the data set, to get the Pearson correlation matrix between variables Q_{cp} and W_{cp} . The coefficients varied from -0.4457 to 0.8638 (excluding correlations $\rho_{Q_{cp}, Q_{cp}}$ and $\rho_{W_{cp}, W_{cp}}$, $\forall c \in C$). We draw a heat map with this correlation matrix, which is depicted in Figure 11.

Figure 11 – Correlations matrix heat map for the credit card data set



Source: the author.

We run the model with different levels of simulation convergence criteria (0.1, 0.01, 0.001), considering the active probability, computing the correlation structure, and number of customers (20, 40 and 79). Table 9 shows the runtime and number of simulation runs necessary for convergence.

Table 9 – Runtime and number of runs for the contractual setting simulations

Number of customers	Active probability	Correlation structure	Convergence criteria	Run time (s)	Number of simulation runs
20	False	True	0.001	154.0	345
20	False	True	0.01	13.6	26
20	False	True	0.1	2.6	5
20	True	False	0.001	115.1	204
20	True	False	0.01	19.3	30
20	True	False	0.1	2.7	4
40	False	True	0.001	211.1	328
40	False	True	0.01	31.4	46
40	False	True	0.1	4.9	7
40	True	False	0.001	342.5	410
40	True	False	0.01	35.3	39
40	True	False	0.1	3.6	4
79	False	True	0.001	612.6	535
79	False	True	0.01	59.5	55
79	False	True	0.1	10.4	8
79	True	False	0.001	735.9	501
79	True	False	0.01	71.1	49
79	True	False	0.1	18.5	11

Font: authors archives

We selected ten customers simulated with the whole data set until convergence in 0.001, considering the correlation structure and don't considering the active probability. Their mean CLV and standard deviation from 535 independent runs are shown in Table 10.

Table 10 – Mean CLV and standard deviation for a few customers of the contractual data set

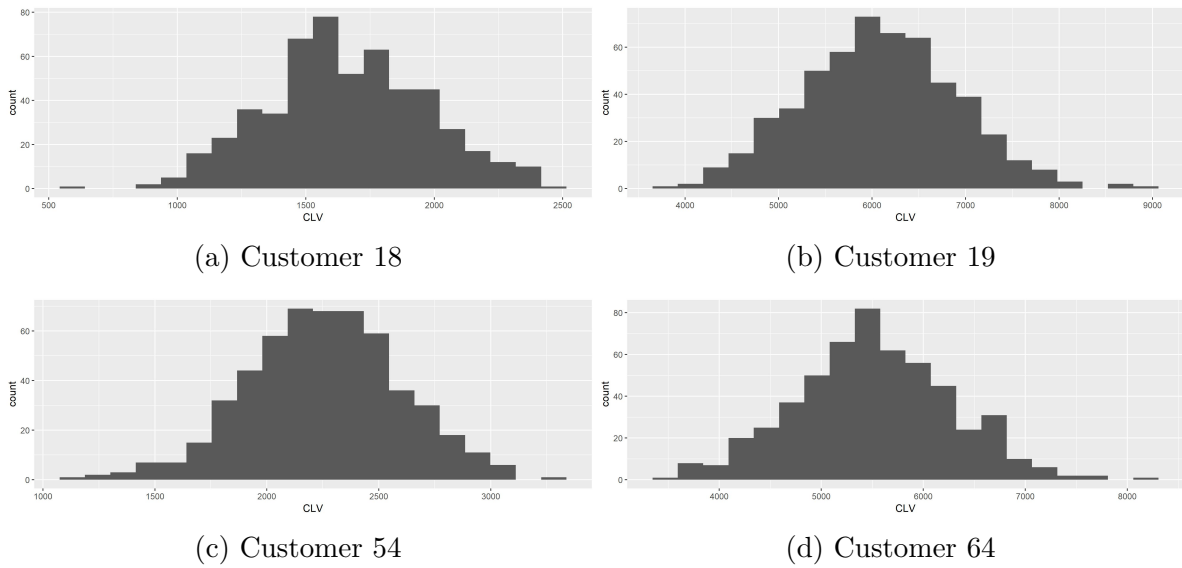
Customer	Mean CLV	Sd
75	2254.1	343.2
58	6087.9	829.6
25	1655.4	308.3
28	5508.9	763.9
42	13044.7	1591.6
32	13843	1176.7
14	23242.2	3408
51	5262.3	677.1
64	4194.4	595.4
72	21300.6	2387.4

Source: the author.

The distribution of simulated CLVs (535 independent runs) from four customers were plotted in histograms, as depicted in Figure 12.

As we did in the non-contractual case, we tested these series against the hypothesis of normality. Shapiro-Wilk test was used with the CLV distribution of all 79 customers of the data set. Of those, 70 adjusted ($\alpha = .05$) to the normal distribution (88.6%).

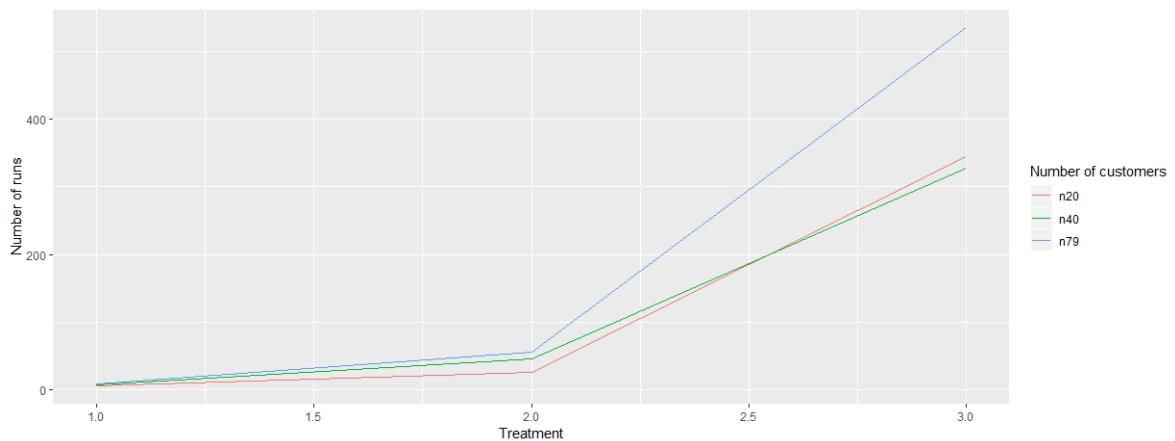
Figure 12 – CLV distribution histograms for a few customers



Source: the author

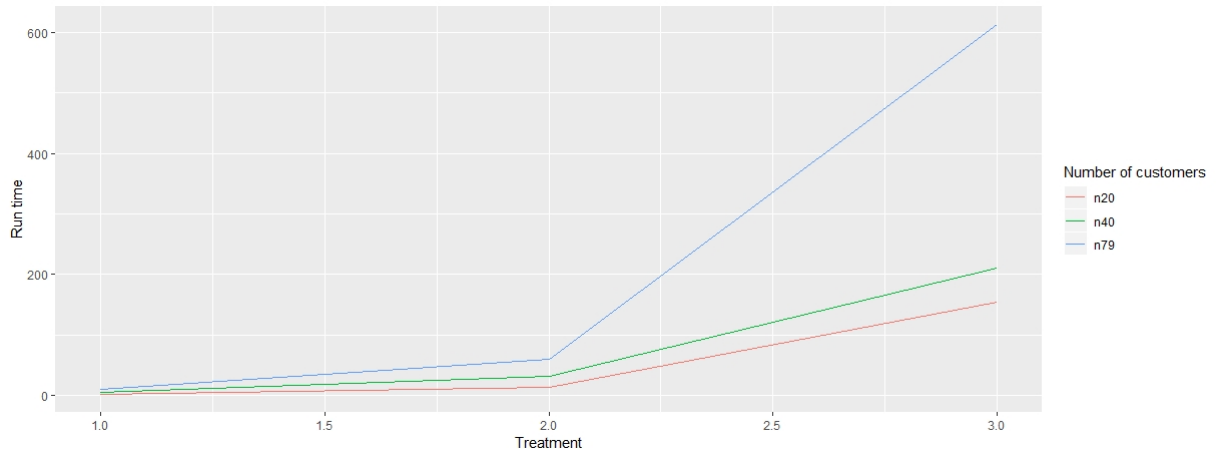
Figure 13 depicts the number of runs needed for convergence, with different levels of convergence criterion, and Figure 14 shows the convergence run time for the different treatments chosen. As it happened in the non-contractual case, here we can see that the number of runs necessary for convergence is less affected by the convergence criterion than the run time, probably due to the fact that a single customer will take a very similar number of runs to converge when the data set increases in number of customers. The quick growth in run time for the 79-customers case, as compared to 20 and 40 customers cases, could be due to the growth in the data structures and amount of randomly generated numbers necessary for the simulator.

Figure 13 – Number of runs needed for convergence with different treatments, for the contractual setting



Source: the author.

Figure 14 – Convergence run time for different treatments, for the contractual setting

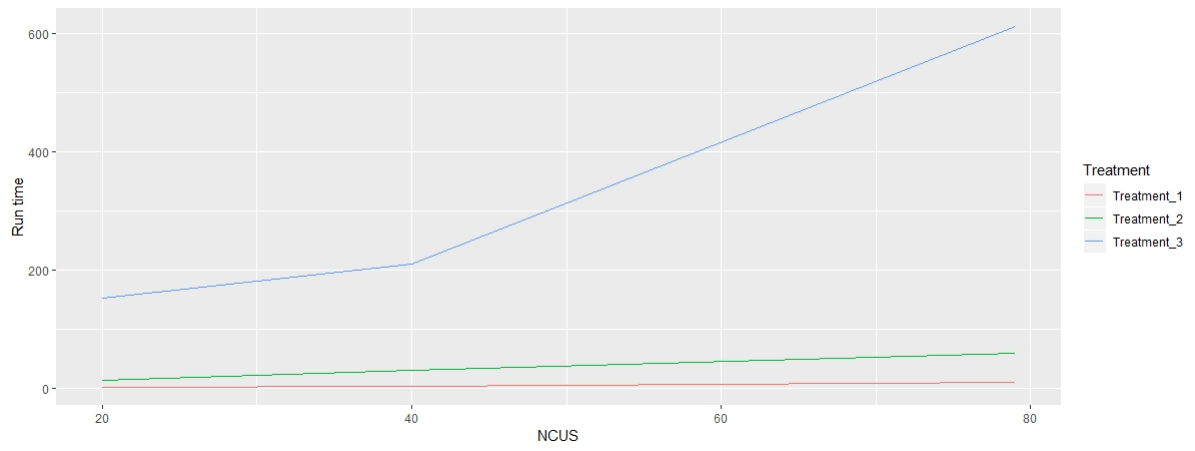


Source: the author.

Figure 15 shows the run time needed for the simulations convergence when changing the number of customers, for the different levels of convergence criteria. When the treatment is held fixed and the number of customers increases by an order of almost 4 (3.95, from 20 to 79), then the run time increases by an order of $\frac{735.9-154.0}{154.0} = 3.77$ in treatment 3 (convergence 0.001), by an order of $\frac{71.1-13.6}{13.6} = 4.22$ in treatment 2 (convergence 0.01), and by an order of $\frac{18.5-2.6}{2.6} = 6.11$ in treatment 1 (convergence 0.1). This suggests that for a given convergence criteria, an increase in the amount of simulated customers will cause linear growth in run time, or even less than that.

This result demonstrates that even a programming language not intended for strict performance, like R, but easier to program than faster languages, like C++, can be used in some real world implementations. Fader, Hardie and Lee (2005) pointed out the implementation easiness of their model, as the use of marketing models in actual practice is becoming less of an exception, and more of a rule.

Figure 15 – Convergence run time for different numbers of customers, for the contractual setting



Source: the author.

5 CONCLUSIONS

In this dissertation, we developed and implemented in two real cases a stochastic model to estimate the discounted lifetime value of a customer. The problem addressed consists in identifying a model that allows an adequate representation of the relationship between customers and companies, so that one can obtain good predictions of future cash flows that clients can provide to companies. The goal of this dissertation is, therefore, to contribute to the body of research on CLV by developing a model that can estimate CLV in two different configurations of customer-company settings. The selected model is the renewal reward process, a generalization of the Poisson process, which allows the use of any probability distribution family to the interpurchase times variable.

The relationship between companies and customers is complex. The extant literature has developed some methods to classify CLV-related problems, such as non-contractual and contractual, as to the nature of the relationship, discrete and continuous, as to the opportunities of transactions, and so forth. Bechwati and Eshghi (2005) points out that models must take into account the patterns of cash flow: “it is not one size fits all”. Besides that, dropout rates must be considered, possible correlation between customers, and the possibility that, after leaving the company, the former client returns.

Our first specific objective leads us to identify the following relevant variables: interpurchase times, rewards received from each customer, number of purchases made by a customer in a given period, sum of rewards received from a customer in a given period, normalized time of the i^{th} purchase, and normalized rewards. Those variables were operationalized according to the relationship nature between company and customer. Our second specific objective leads us to attribute distribution functions families which best fit the aggregate of customers. The third objective, successfully implemented, made the problem even more complex, which is one of the reasons we had to resort on discrete-event simulation method to solve it. Our fourth specific objective leads us to the renewal reward process. Finally, regarding the fifth specific objective, we tested the model with two real world sets of transactions, from two different settings.

We selected the aforementioned variables because they are readily available in a company, or they can be easily computed from transactional data. Specifically in the case of the variables used for the contractual setting, the choice is based on the patterns that we suppose can be found in this particular setting. In this case, the contracted service payments frequency may coincide with the customer’s salary frequency, possibly giving rise to a pattern that repeats in these periods.

There are several CLV-correlated problems that can be found in the literature

and that directly or indirectly affects customer profitability. Amongst them, we mention: the estimation of dropout rates, a probability that a given customer is close to end his relationship with the company, the link between customer equity (summation of CLVs over all clients) and firm value, etc.

Our model is characterized by: allowing the variables to be adjusted to any distribution family, considering an infinite prediction horizon, considering a possible correlation structure between customers, and taking into account the specificities of the contractual setting, by identifying a pattern in the purchases incidence. In short, it is flexible enough to take into account all those characteristics when estimating CLV, or easily disconsider some of them, when we find out that they're not relevant to a specific application.

We point out the main managerial implications of our model: (1) the correlation structure present in the transactional data is replicated in the simulated distributions, which leads to a more realistic customers portfolio risk assessment; (2) the manager don't need to stipulate a simulation horizon, because the model internally handles the CLV distribution convergence; (3) the manager can decide which CDFs will be adjusted to the variables, which makes the model more robust and customizable to the application specificities; (4) the consideration of a customizable active distribution makes the model adaptable to the specific application churn behavior. More than that, if the manager finds out that any of those customizations are not needed, it can easily be removed from the model.

The model was implemented and solved using discrete-event simulation. It was run using two real world data sets: one provided by a groceries store owner, and another by a credit card company.

One of the limitations is that only one case was use to test each setting (contractual and non-contractual). We also didn't have access to dropout rates from the customers, so we had to use artificially generated data to test a part of our model. Future studies could explore our model in different industries, other than groceries and credit card. Due to a relatively short time horizon in the data sets, it wasn't possible to study the influence of seasonality, which is likely to be present in both cases.

BIBLIOGRAPHY

- ALLENBY, Greg M.; LEONE, Robert P.; JEN, Lichung. A dynamic model of purchase timing with application to direct marketing. *Journal of the American Statistical Association*, v. 94, n. 446, p. 365–374, 1999.
- BECHWATI, Nada Nasr; ESHGHI, Abdolreza. Customer lifetime value analysis: Challenges and words of caution. *Marketing Management Journal*, v. 15, n. 2, p. 87–97, 2005.
- BELL, David; DEIGHTON, John; REINARTZ, Werner J.; RUST, Roland T.; SWARTZ, Gordon. Seven barriers to customer equity management. *Journal of Service Research*, v. 5, n. 1, p. 77–85, 2002.
- BERGER, Paul D.; NASR, Nada I. Customer lifetime value: Marketing models and applications. *Journal of Interactive Marketing*, v. 12, n. 1, p. 17 – 30, 1998.
- BOLTON, Ruth N. A dynamic model of the duration of the customer’s relationship with a continuous service provider: The role of satisfaction. *Marketing Science*, v. 17, n. 1, p. 45–65, 1998.
- DWYER, F. Robert. Customer lifetime valuation to support marketing decision making. *Journal of Interactive Marketing*, v. 11, n. 4, p. 6 – 13, 1997.
- EHRENBERG, A. S. C. The pattern of consumer purchases. *Applied Statistics*, v. 8, n. 1, p. 26–41, 1959.
- FADER, Peter S.; HARDIE, Bruce G. S. How to project customer retention. *Journal of Interactive Marketing*, v. 21, n. 1, p. 76–90, 2007.
- _____. Probability models for customer-base analysis. *Journal of Interactive Marketing*, v. 23, n. 1, p. 61 – 69, 2009.
- FADER, Peter S.; HARDIE, Bruce G. S.; LEE, Ka Lok. ‘Counting your Customers’ the easy way: An alternative to the pareto/nbd model. *Marketing Science*, v. 24, n. 2, p. 275–284, 2005.
- FADER, Peter S.; HARDIE, Bruce G. S.; SHANG, Jen. Customer-base analysis in a discrete-time noncontractual setting. *Marketing Science*, v. 29, n. 6, p. 1086–1108, 2010.
- GUPTA, Sunil; HANSSENS, Dominique; HARDIE, Bruce; KAHN, William; KUMAR, V.; LIN, Nathaniel; RAVISHANKER, Nalini; SRIRAM, S. Modeling customer lifetime value. *Journal of Service Research*, v. 9, n. 2, p. 139–155, 2006.
- HILLIER, Frederick S.; LIEBERMAN, Gerald J. *Introduction to Operations Research*. New York: McGraw-Hill Education, 2015.
- IBGE. *Tábuas Completas de Mortalidade*. Instituto Brasileiro de Geografia e Estatística, 2015. Disponível em: <<https://www.ibge.gov.br/estatisticas-novoportal/sociais/populacao/9126-tabuas-completas-de-mortalidade.html?=&t=o-que-e>>. Acesso em: 10 october 2018.

KUMAR, V.; SHAH, Denish. Expanding the role of marketing: From customer equity to market capitalization. *Journal of Marketing*, v. 73, n. 6, p. 119–136, 2009.

LAW, Averill M. *Simulation Modeling and Analysis*. 5. ed. [S.l.]: McGraw-Hill Education, 2015. ISBN 0073401323.

LAW, Averill M.; KELTON, W. David. *Simulation Modeling and Analysis*. 3. ed. New York: McGraw-Hill Higher Education, 2000. ISBN 0070592926.

NESLIN, Scott A.; TAYLOR, Gail Ayala; GRANTHAM, Kimberly D.; MCNEIL, Kimberly R. Overcoming the ‘recency trap’ in customer relationship management. *Journal of the Academy of Marketing Science*, v. 41, n. 3, p. 320–337, 2013.

PARK, Chang Hee; PARK, Young-Hoon; SCHWEIDEL, David A. A multi-category customer base analysis. *International Journal of Research in Marketing*, v. 31, n. 3, p. 266 – 279, 2014. ISSN 0167-8116.

PERKINS-MUNN, Tiffany; AKSOY, Lerzan; KEININGHAM, Timothy L.; ESTRIN, Demitry. Actual purchase as a proxy for share of wallet. *Journal of Service Research*, v. 7, n. 3, p. 245–256, 2005.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2018. Disponível em: <<https://www.R-project.org/>>.

REINARTZ, Werner J.; KUMAR, V. On the profitability of long-life customers in a noncontractual setting: An empirical investigation and implications for marketing. *Journal of Marketing*, v. 64, n. 4, p. 17–35, 2000.

ROMERO, Jaime; LANS, Ralf van der; WIERENGA, Berend. A partially hidden Markov model of customer dynamics for CLV measurement. *Journal of Interactive Marketing*, v. 27, n. 3, p. 185 – 208, 2013. ISSN 1094-9968.

ROSS, Sheldon. *Stochastic Processes*. 2. ed. Canada: John Wiley & Sons, 1996. ISBN 9780471120629.

SCHMITTLEIN, David C.; MORRISON, Donald G.; COLOMBO, Richard. Counting your customers: Who-are they and what will they do next? *Management Science*, v. 33, n. 1, p. 1–24, 1987.

SCHMITTLEIN, David C.; PETERSON, Robert A. Customer base analysis: An industrial purchase process application. *Marketing Science*, v. 13, n. 1, p. 41–67, 1994.

SERFOZO, Richard. *Basics of Applied Stochastic Processes*. Berlin: Springer-Verlag, 2009. Disponível em: <<http://link.springer.com/book/10.1007%2F978-3-540-89332-5>>. Acesso em: 23 junho 2015.

SUNDER, Sarang; KUMAR, V.; ZHAO, Yi. Measuring the lifetime value of a customer in the consumer packaged goods industry. *Journal of Marketing Research*, v. 53, n. 6, p. 901–921, 2016.

WIRTZ, Jochen; XIAO, Ping; CHIANG, Jeongwen; MALHOTRA, Naresh. Contrasting the drivers of switching intent and switching behavior in contractual service settings. *Journal of Retailing*, v. 90, n. 4, p. 463 – 480, 2014.

WU, Couchen; CHEN, Hsiu-Li. Counting your customers: Compounding customer's in-store decisions, interpurchase time and repurchasing behavior. *European Journal of Operational Research*, v. 127, n. 1, p. 109 – 119, 2000.

APPENDIX A – FUNCTIONS USED TO SIMULATE THE CLV MODEL WITH THE NON-CONTRACTUAL SETTING

Main program.

```

1 #####
2 #   Non-contractual Setting
3 #       Main Program
4 #
5 #####
6
7
8 ##### HEADER #####
9 # Initialize environment
10 source("initialization.R")
11
12 # Load functions
13 Load_Functions("General", "CLV", "Optim", "Tests", "Graph")
14
15
16 ##### SOLVE THE NON-CONTRACTUAL MODEL BY SIMULATION #####
17
18 # MODEL PARAMETERS
19 param <- list(
20     # Name of the data set
21     data_set      = (____),
22     # Minimum number of observations per customer
23     min_obs      = (____),
24     # Remove outliers from the data set?
25     remove_outliers = (____),
26     # Amount of subjects to estimate CLV
27     n_subjects   = (____),
28     # Annual Discount rate
29     annual_disc_rate = (____),
30     # Estimate the best distribution to the data? Else: empirical
31     best_fit      = (____),
32     # Use the variables correlation structure in the simulation?
33     vars_correl   = (____),
34     # Convergence criteria for the correlation matrix bootstrapping
35     corr_converge = (____),
36     # Consider the survival probability of each customer
37     survival_prob = (____),
38     # Minimum relative precision for the simulation convergence test
39     sim_converge  = (____)
40 )
41
42
43 # CLV SIMULATIONS
44 #####
45 CLV_results <- compute_CLV()

```

Initialization script.

```

1 #####
2 # Workspace initialization
3 #
4 #####
5
6 # Remove variables
7 rm(list=ls ())
8
9 # Clean memory
10 gc(reset = T)
11
12 # Set default options
13 Sys.setlocale("LC_ALL", "English")
14
15 # Load packages
16 source("Load_Packages.R")
17
18 # Load functions
19 source("Load_Functions.R")
20
21 # Parallelization
22 nCores <- detectCores(logical = TRUE) - 1
23
24 # Clean console
25 cat("\f")

```

Packages loading script.

```

1 # General statistics
2 library(MASS)
3 library(VGAM)
4 library(EnvStats)
5 library(psych)
6
7 # Extreme value theory
8 library(evd)
9
10 # Goodness-of-fit tests
11 library(ADGofTest)
12
13 # Data transformation and similar packages
14 library(plyr)
15 library(reshape2)
16
17 # Graphics
18 library(ggplot2)
19
20 # C++ integration
21 library(Rcpp)
22
23 # Parallelization
24 library(parallel)
25
26 # Mathematical programming
27 library(quadprog)
28 library(lpSolve)

```

```

29
30 # Others
31 library(tictoc)
32 library(beepr)
33 library(microbenchmark)
34 library(lubridate)
35 library(eeptools)

```

Definition of function Load_Functions().

```

1 Load_Functions <- function (...) {
2
3   prefix <- c(...)
4   func_list <- vector()
5   for(p in prefix){
6     func_list <- c(func_list ,
7     dir(path = paste0("./functions"),
8     pattern = paste0(p),
9     full.names = TRUE)
10    )
11   }
12   for(fun in func_list){
13     if(grepl(pattern = ".R", x = fun)){
14       source(fun)
15     } else if(grepl(pattern = ".cpp", x = fun)){
16       Rcpp::sourceCpp(fun)
17     }
18   }
19   cat("\f")
20 }

```

Definition of function compute_CLV().

```

1 compute_CLV <- function () {
2
3   # 0 Preamble
4   result <- list()
5   chol_factor <- NULL
6   tCor <- NULL
7
8   daily_disc_rate <- -log(1 - param$annual_disc_rate) / 365
9
10
11  # 1 Load data, remove zeroes
12  data <- read_data(data_set = param$data_set, remove_outliers = param$remove_
13    outliers ,
14  min_obs = param$min_obs)
15  first_day <- min(data$Date)
16  last_day <- max(data$Date)
17  if(param$survival_prob){
18    births <- read.csv(file = "../supermercado_01/birth.csv")
19    births$birth <- as.Date(as.character(births$birth), format = "%Y-%m-%d")
20    mortality <- read.csv(file = "../mortality_tables/mortality_table_2015.csv")
21  }
22
23  # 2 Prepare data for the simulations

```

```

24 ids      <- unique(data$Cod_cli)
25 # include <- as.numeric(unlist(read.csv("../supermercado_01/sampled_ids.csv")))
    [1:param$n_subjects]
26 include  <- sample(ids, size = param$n_subjects)
27 rows_incl <- unlist(lapply(include, function(x) which(data$Cod_cli == x)))
28 split_data <- split(x = data[rows_incl, c(3,5)], f = data$Cod_cli[rows_incl])
29
30
31 # 3 Estimate the distributions parameters, means and sd
32 tic("Parameter_estimation")
33 # var_dists <- split(x = data[rows_incl, c(3,5)], f = data$Cod_cli[rows_incl])
34 cl       <- makeCluster(nCores)
35 clusterExport(cl, list("fit_dist", "best_fit", "fit_empirical", "param",
36 "gen_rnd", "rempirical"))
37 var_dists <- parLapply(cl, split_data, function(x){
38 library(ADGofTest)
39 library(evd)
40 RW <- fit_dist(series = x[,1], best_dist = param$best_fit)
41 IP <- fit_dist(series = x[,2], best_dist = param$best_fit)
42 return(list("RW" = RW, "IP" = IP))
43 })
44 stopCluster(cl)
45 var_mean <- as.vector(vapply(split_data, function(x){
46 RW_mean <- mean(x[,1])
47 IP_mean <- mean(x[,2])
48 return(c(RW_mean, IP_mean))
49 }, FUN.VALUE = c(.1,.1)))
50 var_sd <- as.vector(vapply(split_data, function(x){
51 RW_sd <- sd(x[,1])
52 IP_sd <- sd(x[,2])
53 return(c(RW_sd, IP_sd))
54 }, FUN.VALUE = c(.1,.1)))
55 tPar <- toc()
56
57
58 # 4 Estimate the correlation and decompose the matrix, when needed
59
60 if(param$vars_correl){
61 tic("Correlation")
62 cor_mat <- corr_fun(subj_vars = split_data,
63 dists = var_dists,
64 corr_converge = param$corr_converge)
65 chol_factor <- chol(cor_mat)
66 tCor <- toc()
67 }
68 if(param$cor_boots_only) return(c(param$n_subjects, param$corr_converge, as.
    numeric(tBot$toc-tBot$tic)))
69
70
71 # 5 Separate the customer's dates of birth for the simulation
72 if(param$survival_prob){
73 births <- lapply(as.integer(names(var_dists)), function(x){
74 births$birth[which(births$Cod == x)]
75 })
76 }
77
78

```

```

79 # 6 Run the simulations for the dataset
80 tic("Simulation")
81 simulated_CLV <- CLV_simulation(var_dists = var_dists,
82 births = births,
83 mortality = mortality,
84 chol_factor = chol_factor,
85 disc_rate = daily_disc_rate,
86 means = var_mean,
87 sds = var_sd,
88 ini_date = last_day+1)
89 tSim <- toc()
90
91 result <- list("Customers" = as.numeric(names(var_dists)),
92 "CLV_vectors" = simulated_CLV$CLV,
93 "Convergence_time" = simulated_CLV$Convergence_time,
94 "n_simulations" = simulated_CLV$n_runs,
95 "Running_time" = c("Parameter_estimation"=tPar$toc-tPar$tic,
96 "Correlation"=tCor$toc-tCor$tic,
97 "Simulation"=tSim$toc-tSim$tic))
98 return(result)
99 }

```

Definition of function fit_empirical().

```

1 fit_empirical <- function(series){
2
3 len <- length(series)
4 values <- sort(unique(series))
5 dist <- data.frame("values" = values)
6
7 dist$cdf <- vapply(values, function(x) sum(series <= x) / len, FUN.VALUE = .1)
8 dist <- as.matrix(dist)
9
10 return(dist)
11 }

```

Definition of function corr_fun().

```

1 corr_fun <- function(subj_vars, dists, corr_converge){
2
3 # Prepare data
4 n_subj <- length(subj_vars)
5 n_vars <- ncol(subj_vars[[1]])
6 matrix_dim <- n_subj * n_vars
7 l <- vector(mode = "list", length = matrix_dim)
8 for(i in 1:n_subj){
9 l[(i * n_vars - n_vars + 1):(i * n_vars)] <- as.list(subj_vars[[i]])
10 }
11
12 ## Compute the correlation matrix
13
14 # Populate initial values for ini_mat
15 max_len <- max(vapply(l, length, FUN.VALUE = 1))
16 ini_mat <- vapply(l, function(x){
17 c(x, rep(NA, max_len - length(x)))
18 }, FUN.VALUE = rep(.1, max_len))
19

```



```

20 # Estimate the empirical distributions
21 dists <- lapply(1, fit_empirical)
22
23 # Compute the bootstrapped correlation matrix
24 corr_matrix <- avg_cor(ini_mat = ini_mat, dists = dists, converge = corr_converge
25 )
26 return(corr_matrix)
27 }

```

Definition of functions `rempirical()` and `avg_cor()`.

```

1 #include <Rcpp.h>
2 #include <cmath>
3 #include <chrono>
4
5 using namespace Rcpp;
6
7 // [[Rcpp::plugins(cpp11)]]
8
9 int supremum(NumericVector A, double T){
10 int L = 0;
11 int R = A.size();
12
13 while(L < R){
14 int m = floor((L + R) / 2);
15
16 if(A[m] < T)
17 L = m + 1;
18 else
19 R = m;
20 }
21 return L;
22 };
23
24 // [[Rcpp::export]]
25 NumericVector rempirical(int n, NumericMatrix par){
26 // n: number of random numbers to be generated
27 // par: two-column empirical distribution matrix
28
29 int idx;
30 NumericVector u(n);
31 NumericVector Vr(n);
32
33 // Generate uniformly distribute random numbers
34 u = runif(n);
35
36 // Search for the supremum on the par_matrix
37 for(int i=0; i<n; i++){
38 idx = supremum(par.column(1), u[i]);
39 Vr[i] = par.column(0)[idx];
40 }
41 return Vr;
42 }
43
44 NumericMatrix corC(NumericMatrix x) {
45

```

```

46 Environment stats("package:stats");
47 Function corr=stats["cor"];
48 NumericMatrix out=corr(x);
49
50 return out;
51 }
52
53 NumericMatrix SumMat(NumericMatrix x, NumericMatrix y){
54
55 NumericMatrix out(x.nrow(), x.ncol());
56
57 for(int i=0; i<x.nrow(); i++)
58 for(int j=0; j<x.ncol(); j++)
59 out(i,j) = x(i,j) + y(i,j);
60 return out;
61 }
62
63 NumericMatrix DifMat(NumericMatrix x, NumericMatrix y){
64
65 NumericMatrix out(x.nrow(), x.ncol());
66
67 for(int i=0; i<x.nrow(); i++)
68 for(int j=0; j<x.ncol(); j++)
69 out(i,j) = x(i,j) - y(i,j);
70 return out;
71 }
72
73 NumericMatrix DivMat(NumericMatrix x, NumericMatrix y){
74
75 NumericMatrix out(x.nrow(), x.ncol());
76
77 for(int i=0; i<x.nrow(); i++)
78 for(int j=0; j<x.ncol(); j++)
79 out(i,j) = x(i,j) / y(i,j);
80 return out;
81 }
82
83 NumericMatrix AbsMat(NumericMatrix x){
84
85 NumericMatrix out(x.nrow(), x.ncol());
86
87 for(int i=0; i<x.nrow(); i++){
88 for(int j=0; j<x.ncol(); j++){
89 out(i,j) = fabs(x(i,j));
90 }
91 }
92 return out;
93 }
94
95 double AcumMat(NumericMatrix x){
96
97 int nCol = x.ncol();
98 double out = 0;
99 for(int i=0; i<nCol; i++)
100 out += sum(x.column(i));
101 return out;
102 }

```

```

103
104 bool nextStepTest(NumericMatrix x, double maxCrit){
105 // Returns true if at least one correlation is not lesser than maxCrit
106
107 int nCol = x.ncol();
108 int nRow = x.nrow();
109 bool out = true;
110 for(int i=0; i<nRow; i++)
111 for(int j=0; j<nCol; j++)
112 out = out & (x(i,j) < maxCrit);
113 return !out;
114 }
115
116 // [[Rcpp::export]]
117 NumericMatrix avg_cor(NumericMatrix ini_mat, List dists, double converge){
118 /* ini_mat: initial values of the matrix, used to estimate the empirical CDF
119 the missing part will be bootstrapped
120 dists: list of empirical CDF matrices
121 converge: the minimum convergence decimal for the correlations */
122
123 int Ncol = ini_mat.ncol();
124 int Nrow = ini_mat.nrow();
125 IntegerVector len_rand = rep(0,Ncol);
126 NumericMatrix r_mat(clone(ini_mat)); // bootstrapped matrix
127 NumericMatrix r_cor(Ncol, Ncol); // Randomly generated correlation matrix
128 NumericMatrix diff(Ncol, Ncol); // Matrix w differences between new and avg
    corr
129 NumericMatrix out(Ncol, Ncol); // Average correlations matrix
130 NumericMatrix auxOut(Ncol, Ncol);
131 bool next = true;
132
133 // 0 Compute the length of the random part of each variable
134 for(int j=0; j<Ncol; j++)
135 for(int i=0; i<Nrow; i++)
136 if(r_mat.column(j)[i] == NA)
137 len_rand[j]++;
138
139 int N = 1;
140
141 // 1 Populate the random part of out
142 for(int j=0; j<Ncol; j++)
143 for(int i=0; i<len_rand[j]; i++)
144 r_mat((Nrow - len_rand[j] + i), j) = as<double>(rempirical(1, dists[j]));
145
146 // 2 Compute the correlation matrix
147 out = corC(r_mat);
148
149 while(next){
150 N++;
151
152 // 3 Repopulate the random part of out
153 for(int j=0; j<Ncol; j++)
154 for(int i=0; i<len_rand[j]; i++)
155 r_mat((Nrow - len_rand[j] + i), j) = as<double>(rempirical(1, dists[j]));
156
157 // 4 Recompute the correlation matrix
158 r_cor = corC(r_mat);

```

```

159
160 // 5 Compute the new average correlation matrix
161 auxOut = out;
162 out = SumMat((out * (N-1)), r_cor) / N;
163
164 // 6 Compute the stopping criteria
165 diff = AbsMat(DifMat(out, auxOut));
166 next = nextStepTest(diff, converge);
167 }
168 return out;
169 }

```

Definition of function `CLV_simulation()`.

```

1 CLV_simulation <- function(var_dists, births, mortality, means, sds,
2 chol_factor = NULL, disc_rate, ini_date){
3
4 n_ini <- 500 # initial amount of random numbers for each customer
5 seq_min <- rep(TRUE, 10) # Minimum sequence length of converged differences
6 n_cus <- length(var_dists)
7 RW_idx <- seq(1, n_cus*2-1, by = 2)
8 IP_idx <- seq(2, n_cus*2, by = 2)
9 CLV <- matrix(nrow = 0, ncol = n_cus)
10 mean_CLV <- rep(0, length = n_cus)
11 converge_mat <- matrix(nrow = 0, ncol = n_cus)
12 n_path <- 1
13 # sim_time <- 0
14
15
16 ## LOOP FOR THE SERIES OF SIMULATIONS
17 repeat{
18 # tic("path_time")
19 # print(paste("path", n_path))
20
21 # Initialize the simulation variables
22 rnd_vars <- matrix(nrow = 0, ncol = 2*n_cus)
23 IP_cum <- matrix(nrow = 0, ncol = n_cus)
24 RW_disc <- matrix(nrow = 0, ncol = n_cus)
25 RW_cum_disc <- matrix(nrow = 0, ncol = n_cus)
26 active <- matrix(data = TRUE, nrow = n_ini, ncol = n_cus)
27 converge_row <- rep(0, n_cus)
28 n_run <- 1
29
30 # Randomly generates dates at which customers will die, if necessary
31 if(param$survival_prob) die_dates <- die_date_fun(dates_births = births, ini_date
32 , mortality)
33
34 ## LOOP FOR A SINGLE PATH
35 repeat{
36 # print(paste("run", n_run))
37
38 # Computes the current run simulation rows
39 run_rows <- (n_run * n_ini - n_ini + 1):(n_run * n_ini)
40
41 # 0 Generate series of uncorrelated random variables
42 rnd_list <- lapply(var_dists, function(x){

```

```

43 RW_dist <- x$RW$dist
44 RW_par <- x$RW$pars
45 IP_dist <- x$IP$dist
46 IP_par <- x$IP$pars
47
48 IP_rnd <- gen_rnd(n = n_ini, dist = IP_dist, par = IP_par, var = "IP")
49 RW_rnd <- gen_rnd(n = n_ini, dist = RW_dist, par = RW_par, var = "RW")
50 out <- matrix(data = c(RW_rnd, IP_rnd), ncol = 2, byrow = F)
51
52 return(out)
53 })
54 rnd_vars <- rbind(rnd_vars, do.call(cbind, rnd_list))
55
56 # 1 Multiply by Cholesky, if needed
57 if(param$vars_correl){
58 # Standardize the series
59 rnd_vars[run_rows,] <- sweep(rnd_vars[run_rows,], 2, means) %*% diag(1/sds)
60
61 # Multiply the series by the Cholesky factor
62 rnd_vars[run_rows,] <- rnd_vars[run_rows,] %*% chol_factor
63
64 # Reverse the standardization
65 rnd_vars[run_rows,] <- sweep((rnd_vars[run_rows,] %*% diag(sds)), 2, means, "+")
66 }
67
68 # 2 Update IP_cum
69 IP_cum <- apply(rnd_vars[, IP_idx], 2, cumsum)
70
71 # 3 Computes the vector of dates and the vector of mortality, if needed
72 if(param$survival_prob){
73 for(i in 1:n_cus){
74 test_dates <- ini_date + IP_cum[run_rows, i]
75 active[, i] <- test_dates < die_dates[[i]]
76 }
77 }
78
79 # 3 Update RW_cum_disc
80 RW_disc <- rbind(RW_disc,
81 rnd_vars[run_rows, RW_idx] *
82 exp(-disc_rate * IP_cum[run_rows,]) *
83 active)
84 RW_cum_disc <- apply(RW_disc, 2, cumsum)
85
86 # 3 Compute convergence only for non converged customers
87 not_conv <- which(converge_row == 0)
88 converge_row[not_conv] <- apply(RW_cum_disc[run_rows, not_conv, drop=F], 2,
89 function(x){
90 differences <- diff(x) / x[1:(length(x)-1)]
91 differences[is.nan(differences)] <- 0
92 test <- differences < .001
93 true_ini <- which(test[1:(length(test)-length(seq_min)+1)])
94 converge <- sapply(true_ini, function(ini)
95 all(test[ini:(length(seq_min)+ini-1)] == seq_min))
96 if(any(converge)){
97 converge <- true_ini[converge][1]
98 row <- converge + n_ini * (n_run - 1)
99 return(row)

```

```

99 } else{
100 return(0)
101 }
102 })
103
104 # 4 Test for convergence of all customers
105 if(any(converge_row == 0)){
106 n_run <- n_run + 1
107 } else{
108 converge_mat <- rbind(converge_mat, diag(IP_cum[converge_row,]))
109 break
110 }
111 } # End of single path
112
113 # 5 Save the simulated CLVs
114 CLV <- rbind(CLV,
115 diag(sapply(converge_row, function(x) RW_cum_disc[x,])))
116
117 if(n_path >= 2){
118 new_mean <- (mean_CLV * (n_path - 1) + CLV[n_path,]) / n_path
119 conv_aux <- abs(mean_CLV - new_mean) / mean_CLV < param$sim_converge
120 conv_test <- all(conv_aux)
121
122 end_path <- toc(quiet = TRUE)
123 # sim_time <- sim_time + end_path$toc - end_path$tic
124 print(paste(n_path, ":", sum(conv_aux), "/", n_cus, sep = ""))
125 # if(conv_test | (sim_time > 3600)) break
126 if(conv_test) break
127 } else{
128 mean_CLV <- as.vector(CLV)
129 }
130 n_path <- n_path + 1
131 } # End of simulations
132
133 result <- list("CLV" = CLV,
134 "Convergence_time" = converge_mat,
135 "n_runs" = n_path)
136 return(result)
137 }

```

Definition of function die_date_fun().

```

1 die_date_fun <- function(dates_births, ini_date, mortality){
2 out <- lapply(dates_births, function(x){
3 age <- floor(age_calc(x, enddate = ini_date, units = "years"))
4 row <- which(mortality[,1] == age) + 1
5 ra <- runif(1)
6 for(i in row:nrow(mortality)){
7 if(ra <= mortality[i,2]) break
8 }
9 die_age <- mortality[i,1]
10 rm <- runif(1)
11 die_date <- (x + years(die_age)) %m+% months(ceiling(rm * 12))
12 return(die_date)
13 })
14 return(out)
15 }

```

Definition of function `gen_rnd()`.

```
1 gen_rnd <- function(n, dist, par, var){
2
3   if(dist == "gev"){
4     out <- rgev(n, par[1], par[2], par[3])
5   } else if(dist == "weibull"){
6     out <- rweibull(n, par[1], par[2])
7   } else if(dist == "gamma"){
8     out <- rgamma(n, par[1], par[2])
9   } else if(dist == "empirical"){
10    out <- rempirical(n, par = par)
11  }
12
13  if(var == "IP"){
14    # Always generates >= 1 values for IP
15    out[which(out <= 0)] <- 1
16  }
17
18  return(unnname(out))
19 }
```

APPENDIX B – FUNCTIONS USED TO SIMULATE THE CLV MODEL WITH THE CONTRACTUAL SETTING

Main program.

```

1
2 #####
3 #   Contractual Setting
4 #       Main Program
5 #
6 #####
7
8
9 ##### HEADER #####
10 # Initialize environment
11 source("initialization.R")
12
13 # Load functions
14 Load_Functions("Data", "Vars", "Distributions", "Simulation", "Test",
15               "Graph")
16
17 ##### SOLVE THE CONTRACTUAL MODEL BY SIMULATION #####
18
19 # MODEL PARAMETERS
20
21 param <- list(
22 # Name of the data set
23 data_set      = (_____),
24 # Minimum number of observations per customer
25 min_obs      = (_____),
26 # Remove outliers from the data set?
27 remove_outliers = (_____),
28 # Fraction of subjects to estimate CLV
29 n_subjects    = (_____),
30 # Annual Discount rate
31 anual_disc_rate = (_____),
32 # Estimate the best distribution to the data? Else: empirical

```



```

33 best_fit      = (____),
34 # Use the variables correlation structure in the simulation?
35 vars_correl   = (____),
36 # Consider the survival probability of each customer
37 survival_prob = (____),
38 # Minimum relative precision for the simulation convergence test
39 sim_converge  = (____)
40 )
41
42 # ESTIMATE CLVs
43 CLV_results <- compute_CLV()

```

Packages loading script.

```

1 # General statistics
2 library(MASS)
3 library(rriskDistributions)
4 library(msm)
5 library(EnvStats)
6
7 # Data transformation and similar packages
8 library(reshape2)
9
10 # Graphics
11 library(ggplot2)
12
13 # C++ integration
14 library(Rcpp)
15
16 # Parallelization
17 library(parallel)
18
19 # Others
20 library(tictoc)
21 library(beepr)
22 library(microbenchmark)
23 library(lubridate)
24 library(eeptools)

```

Definition of function compute_CLV().

```

1 compute_CLV <- function() {
2
3 # 0 Preamble

```

```

4  daily_disc_rate <- -log(1 - param$annual_disc_rate) / 365
5
6
7  # 1 Load data
8  data <- read_data(data_set = param$data_set, remove_outliers = param$
  remove_outliers,
9  min_obs = param$min_obs)
10 ids <- unique(data$Cliente)
11 include_ids <- sample(x = ids, size = param$n_subjects)
12 include_rows <- unlist(lapply(include_ids, function(x) which(data$
  Cliente == x)))
13 data <- data[include_rows,]
14 ids <- unique(data$Cliente)
15 # ini_date <- as.Date(paste("1-",month(max(data$Data.da.Transacao)),
  "-",year(max(data$Data.da.Transacao)), sep = ""), format = "%d-%m
  -%Y")
16 ini_date <- dmy(paste("1-",month(max(data$Data.da.Transacao))+1, "-",
  year(max(data$Data.da.Transacao)), sep = ""))
17
18 if(param$survival_prob){
19 births <- read.csv(file = "../cc_movimentos/birth.csv",
  stringsAsFactors = FALSE)
20 births$birth <- ymd(births$birth)
21 mortality <- read.csv(file = "../mortality_tables/mortality_table_
  2015.csv")
22 }
23
24 # 2 Compute the model's variables
25 Q_w_month <- q_w_month_fun(data, ids)
26 d_purch <- d_purch_fun(data, ids)
27 d_rewards <- R_fun(data, ids, Q_w_month)
28
29
30 # 3 Estimate the distribution parameters
31 Q_par <- t(vapply(ids, function(x){
32 series <- Q_w_month$monthly_purchases[which(Q_w_month$id == x)]
33 # series <- series[!series == 0]
34 q <- msm::qtnorm(p = c(0.025, 0.5, 0.75, 0.975), mean = mean(
  series), sd = sd(series), lower = 0, upper = 1)
35 suppressWarnings(get.tnorm.par(q = q, show.output = FALSE))
36 # get.tnorm.par(q = q, show.output = FALSE)
37 }, FUN.VALUE = rep(.1,4))

```

```

38 W_par <- t(vapply(ids, function(x){
39   series <- Q_w_month$sum_purch[which(Q_w_month$id == x)]
40   # series <- series[!series == 0]
41   q      <- msm::qtnorm(p = c(0.025, 0.5, 0.75, 0.975), mean = mean(
      series), sd = sd(series), lower = 0)
42   suppressWarnings(get.tnorm.par(q = q, show.output = FALSE))
43 }, FUN.VALUE = rep(.1,4))
44 d_purch_par <- t(vapply(d_purch, function(x){
45   suppressWarnings(ebeta(x, method = "mle")$parameters)
46 }, rep(.1,2)))
47 d_rewards_par <- t(vapply(d_rewards, function(x){
48   fitdistr(x, densfun = "log-normal")$estimate
49 }, rep(.1,2)))
50
51 # 4 Compute mean and sd for vars Q_m and W_m
52 # Q_mean_sd <- data.frame("mean"=E_tnorm(Q_par[,1],Q_par[,2],Q_par
      [,3],Q_par[,4]),
53 #                          "sd"=sd_tnorm(mean = Q_par[,1],sd = Q_par
      [,2],lw = Q_par[,3],up = Q_par[,4]))
54 # W_mean_sd <- data.frame("mean"=W_par[,1],
55 #                          "sd"=sqrt((W_par[,1]*pi^2)/3))
56
57
58 # 4 Estimate the correlation between Q_month and W_month
59 #   and decompose the matrix, when needed
60 if(param$vars_correl){
61   tic("Correlation")
62   cor_mat <- corr_fun(input_data = Q_w_month[, c(1,5,6)],
63   Q_par,
64   W_par, ids, corr_converge = .001)
65   chol_factor <- chol(cor_mat)
66   tCor <- toc()
67 }
68
69
70 # 5 Separate the customer's dates of birth for the simulation
71 if(param$survival_prob){
72   births <- lapply(ids, function(x){
73     births$birth[which(births$Cod == x)]
74   })
75 }
76

```

```

77
78 # 6 Run the simulations
79 tic("Simulation")
80 simulated_CLV <- CLV_simulation(Q_par, W_par, d_purchase_par, d_rewards_
      par,
81 births, mortality, chol_factor,
82 disc_rate = daily_disc_rate,
83 ini_date = ini_date)
84 tSim <- toc()
85
86 result <- list("Customers" = ids,
87 "CLV_vectors" = simulated_CLV$CLV,
88 "Convergence_time" = simulated_CLV$Convergence_time,
89 "n_simulations" = simulated_CLV$n_runs,
90 "Running_time" = tSim$toc - tSim$tic)
91 return(result)
92 }

```

Definition of function q_w_month_fun().

```

1 q_w_month_fun <- function(data, ids){
2
3 unique_month <- lapply(ids, function(x){
4 rows <- which(data$Cliente == x)
5 n <- length(rows)
6
7 first <- as.Date(
8 paste("1-", month(data$Data.da.Transacao[rows[1]]), "-", year(data$
      Data.da.Transacao[rows[1]]), sep = ""),
9 format = "%d-%m-%Y")
10 last <- as.Date(
11 paste("1-", month(data$Data.da.Transacao[rows[n]]), "-", year(data$
      Data.da.Transacao[rows[n]]), sep = ""),
12 format = "%d-%m-%Y")
13 seq_m <- seq(from = first, to = last, by = "month")
14 df <- data.frame("id" = x, "month" = seq_m)
15 return(df)
16 })
17 out <- do.call(rbind, unique_month)
18 out$month <- ymd(out$month)
19
20 cl <- makeCluster(nCores)
21 clusterExport(cl, list("data"), envir = environment())

```

```

22 vars_vec <- t(parApply(cl, out, 1, function(y){
23   library(lubridate)
24   cli <- as.numeric(y[[1]])
25   dt <- ymd(y[[2]])
26   purchases <- which(data$Cliente == cli &
27     month(data$Data.da.Transacao) == month(dt) &
28     year(data$Data.da.Transacao) == year(dt))
29   days_purch <- day(data$Data.da.Transacao[purchases])
30   n_purch <- length(days_purch)
31   diff_days_purch <- length(unique(days_purch))
32   amount <- sum(data$Valor.da.Transacao[purchases])
33   return(c(n_purch, diff_days_purch, amount))
34 })
35 stopCluster(cl)
36 out$n_purchases <- vars_vec[,1]
37 out$days_purch <- vars_vec[,2]
38 var <- out$days_purch / n_days(out$month)
39 out$monthly_purchases <- var
40 out$sum_purch <- vars_vec[,3]
41 return(out)
42 }

```

Definition of function d_purch_fun().

```

1 d_purch_fun <- function(data, ids){
2   d <- lapply(ids, function(x){
3     rows <- which(data$Cliente == x)
4     dates <- data$Data.da.Transacao[rows]
5     day(dates) / n_days(dates)
6   })
7   return(d)
8 }

```

Definition of function R_fun().

```

1 R_fun <- function(data, ids, W){
2   purch_vectors <- vector(mode = "list", length = nrow(W))
3   id_rows <- lapply(ids, function(x) which(W$id == x))
4   out <- vector(mode = "list", length = length(ids))
5   for(i in 1:nrow(W)){
6     rows <- which(data$Cliente == W$id[i] &
7       month(data$Data.da.Transacao) == month(W$month[i]) &
8       year(data$Data.da.Transacao) == year(W$month[i]))

```

```

9  purch_vectors[[i]] <- data$Valor.da.Transacao[rows] / sum(data$Valor.
    da.Transacao[rows])
10 }
11
12 for(i in 1:length(ids)){
13 out[[i]] <- unlist(purch_vectors[id_rows[[i]])
14 }
15 names(out) <- ids
16 return(out)
17 }

```

Definition of function CLV_simulation().

```

1  CLV_simulation <- function(Q_par, W_par, d_purchase_par, d_rewards_par,
2  births, mortality, chol_factor, disc_rate, ini_date){
3
4  n_ini <- 12 # initial amount of months for each customer
5  n_path <- 1
6  n_cus <- nrow(Q_par)
7  Q_W_par <- split(cbind(Q_par, W_par), seq(n_cus))
8  Q_cols <- seq(from = 1, by = 2, length.out = n_cus)
9  W_cols <- seq(from = 2, by = 2, length.out = n_cus)
10 CLV <- matrix(data = 0, nrow = 0, ncol = n_cus)
11 conv_t <- rep(0, n_cus)
12 conv_vec <- matrix(data = 0, nrow = 0, ncol = n_cus)
13 active <- lapply(1:n_cus, function(x) rep(T, n_ini))
14 sim_time <- 0
15 # Q_W_means <- as.double(unlist(split(cbind(Q_mean_sd[,1], W_mean_sd
    [,1]), seq(n_cus))))
16 # Q_W_sds <- as.double(unlist(split(cbind(Q_mean_sd[,2], W_mean_sd
    [,2]), seq(n_cus))))
17
18
19 ## LOOP FOR THE SERIES OF SIMULATIONS
20
21 repeat{
22 # print(paste("path", n_path))
23 tic("path_time")
24
25 sim_months <- seq(ini_date, by = "months", length.out = n_ini)
26 n_run <- 1
27 run_rows <- (1+n_run*n_ini-n_ini):(n_run*n_ini)
28 CLV_single <- rep(0, n_cus)

```

```

29 CLV_aux      <- rep(0, n_cus)
30 days2sum     <- 0
31 converge     <- rep(FALSE, n_cus)
32
33 # Randomly generates dates at which customers will die, if necessary
34 if(param$survival_prob) die_dates <- die_date_fun(dates_births =
      births, ini_date, mortality)
35
36
37 ## LOOP FOR A SINGLE PATH
38 repeat{
39 # print(paste("run", n_run))
40 n_days_months <- n_days(sim_months) # n_days for the loop months
      only
41
42 # 0 Generate variables Q_m and W_m
43 rnd_list <- lapply(QW_par, function(x){
44 Q <- rtnorm(n_ini, mean = x[[1]], sd = x[[2]], lower = x[[3]], upper
      = x[[4]])
45 W <- rtnorm(n_ini, x[[5]], x[[6]], x[[7]], x[[8]])
46 return(cbind(Q, W))
47 })
48 rnd_QW <- do.call(cbind, rnd_list)
49
50 # 1 Multiply by Cholesky, if needed
51 if(param$vars_correl){
52 # Standardize the series
53 QW_means <- colMeans(rnd_QW[run_rows,])
54 QW_sds <- apply(rnd_QW[run_rows,], 2, sd)
55 rnd_QW <- sweep(rnd_QW[run_rows,], 2, QW_means) %*% diag(1/QW_
      sds)
56
57 # Multiply the series by the Cholesky factor
58 rnd_QW[run_rows,] <- rnd_QW[run_rows,] %*% chol_factor
59
60 # Reverse the standardization
61 rnd_QW <- sweep((rnd_QW[run_rows,] %*% diag(QW_sds)), 2, QW_means
      , "+")
62 }
63
64 # 2 Compute actual number of purchases in each month (Q_month * days_
      in_month)

```

```

65 rnd_Q_W[run_rows, Q_cols] <- sweep(rnd_Q_W[run_rows, Q_cols], MARGIN
   = 1, n_days_months, "*")
66 rnd_Q_W[run_rows, Q_cols] <- round(rnd_Q_W[run_rows, Q_cols])
67
68 # 3 When Q_m is negative, Q_m <- 0 and W_m <- 0
69 #   When W_m is negative, W_m <- Q_m * .01
70 to_zero <- rnd_Q_W[run_rows, Q_cols] < 0
71 rnd_Q_W[run_rows, Q_cols][to_zero] <- 0
72 rnd_Q_W[run_rows, W_cols][to_zero] <- 0
73 to_incr <- rnd_Q_W[run_rows, W_cols] < 0
74 rnd_Q_W[run_rows, W_cols][to_incr] <- .01 * rnd_Q_W[run_rows, Q_cols
   ][to_incr]
75
76 # 4 Compute active probability, when needed
77 if(param$survival_prob){
78 seq_months <- seq(from = ini_date, to = ini_date %m+% months(n_ini-1)
   , by = "months")
79 active <- lapply(die_dates, function(x) seq_months < x)
80 }
81
82 # 5 Generate d_purch and reward
83 for(i in 1:n_cus){
84 # input <- cbind(rnd_Q_W[run_rows, c(Q_cols[i], W_cols[i])], n_days_
   months)
85 # purch_months <- apply(input, 1, function(x){
86 input <- cbind(rnd_Q_W[run_rows, c(Q_cols[i], W_cols[i])], n_days_
   months)
87 input[,1] <- input[,1] * active[[i]]
88 input <- split(input, seq(n_ini))
89 purch_months <- lapply(input, function(x){
90 d <- sort(rbeta(x[[1]], d_purch_par[i,1], d_purch_par[i,2]) * x[[3]])
91 r <- rlnorm(x[[1]], d_rewards_par[i,1], d_rewards_par[i,2])
92 r <- r / sum(r)
93 r <- r * x[[2]]
94 cbind(d, r)
95 })
96 if(n_run > 1){
97 days2sum <- days2sum +
98 sum(n_days(seq(sim_months[1]-months(n_ini), by = "months", length.out
   = n_ini)))
99 for(j in 1:n_ini){
100 purch_months[[j]][,1] <- purch_months[[j]][,1] + days2sum

```



```

101 }
102 purchases <- do.call(rbind, purch_months)
103 CLV_aux[i] <- CLV_single[i]
104 CLV_single[i] <- CLV_single[i] + sum(purchases[,2] * exp(-disc_rate *
    purch_months[,1]))
105 } else{
106 for(j in 2:n_ini){
107 purch_months[[j]][,1] <- purch_months[[j]][,1] + cumsum(n_days_months
    ) [j-1]
108 }
109 purchases <- do.call(rbind, purch_months)
110 CLV_single[i] <- sum(purchases[,2] * exp(-disc_rate * purch_months[,1]))
111 }
112 }
113
114 # 6 Compute convergence
115 converge <- abs(CLV_single - CLV_aux) / CLV_aux < .001
116 conv_aux <- conv_t == 0 & converge
117 conv_t[conv_aux] <- n_run * n_ini
118 if(any(!converge)){
119 n_run <- n_run + 1
120 sim_months <- seq(sim_months[n_ini]+months(1), by = "months", length.
    out = n_ini)
121 } else{
122 break
123 }
124 } # End of a single path
125
126 # 7 Save the simulated CLVs
127 CLV <- rbind(CLV, CLV_single)
128 conv_vec <- rbind(conv_vec, conv_t)
129
130 # 8 Compute paths convergence and stopping criteria
131 if(n_path > 1){
132 new_mean <- (mean_CLV * (n_path-1) + CLV[n_path,]) / n_path
133 sim_test <- abs(new_mean - mean_CLV) / mean_CLV < param$sim_converge
134 print(paste(n_path, ":", sum(sim_test), "/", n_cus, sep = ""))
135 end_path <- toc(quiet = TRUE)
136 sim_time <- sim_time + end_path$toc - end_path$tic
137 if(all(sim_test) | (sim_time > 3600)) break
138 } else{
139 mean_CLV <- CLV[1,]

```

```

140 }
141 n_path <- n_path + 1
142 } # End of simulation
143
144 result <- list("CLV" = CLV,
145 "Convergence_time" = conv_vec,
146 "n_runs" = n_path)
147 return(result)
148 }

```

Definition of function n_days().

```

1 n_days <- function(dates){
2 dt <- ymd(dates)
3 d31 <- c(1, 3, 5, 7, 8, 10, 12)
4 d30 <- c(4, 6, 9, 11)
5
6 m <- month(dates)
7 y <- year(dates)
8
9 in31 <- m %in% d31
10 in30 <- m %in% d30
11 m2 <- m == 2
12 leapyear <- ((y %% 4 == 0) & (y %% 100 != 0)) | (y %% 400 == 0)
13
14 return(in31 * 31 + in30 * 30 + (m2 * leapyear) * 29 + (m2 * !leapyear
15 ) * 28)

```