Journal of Retailing and Consumer Services 52 (2020) 101918

Contents lists available at ScienceDirect

Journal of Retailing and Consumer Services

journal homepage: www.elsevier.com/locate/jretconser

Forecasting client retention — A machine-learning approach

Satu Elisa Schaeffer*, Sara Veronica Rodriguez Sanchez

School of Mechanical and Electrical Engineering, Universidad Autónoma de Nuevo León, Ciudad Universitaria, San Nicolás de los Garza, 66451, Nuevo León, Mexico

ARTICLE INFO

Keywords:
Client retention
Sales forecasting
Machine learning
Prepaid unitary services

ABSTRACT

In the age of big data, companies store practically all data on any client transaction. Making use of this data is commonly done with machine-learning techniques so as to turn it into information that can be used to drive business decisions. Our interest lies in using data on prepaid unitary services in a business-to-business setting to forecast client retention: whether a particular client is at risk of being lost *before* they cease being clients. The purpose of such a forecast is to provide the company with an opportunity to reach out to such clients as an effort to ensure their retention.

We work with monthly records of client transactions: each client is represented as a series of purchases and consumptions. We vary (1) the length of the time period used to make the forecast, (2) the length of a period of inactivity after which a client is assumed to be lost, and (3) how far in advance the forecast is made. Our experimental work finds that current machine-learning techniques able to adequately predict, well in advance, which clients will be lost. This knowledge permits a company to focus marketing efforts on such clients as early as three months in advance.

1. Introduction

The sales of *unitary prepaid services* has become a pervasive business model. It is present in cleaning services (such as prepaid car-wash tickets for a taxi company), telecommunications (prepaid minutes and megabytes on cell phones), and delivery services (including prepaid shipping labels for on-line retail). Huang et al. (2015) reports that prepaid clients have a significantly higher *churn rate* (on average 9.4 percent) than postpaid clients (on average 5.2 percent), meaning that a prepaid customer is almost twice as likely to be a lost client. Understanding the factors that influence whether a specific client is retained offers a competitive advantage (Ankenbruck, 2017).

We work with customer transaction data of unitary, prepaid business-to-business services, although the prepaid-service business model extends also to business-to-consumer markets. Our goal is to determine *beforehand* which clients will be lost, although we do not, in this work, determine the *cause* of the loss — whether it is due to the client switching to a competitor, a discontinued need for the service, or some other reason. This process, best described as *forecasting* client retention, has not yet been widely studied (Ngai et al., 2009; Sabbeh, 2018; Soltani and Navimipour, 2016).

We propose and test a methodology for determining one or more months beforehand which clients will be retained and which will be lost. We vary (1) the length of the time period of client transaction data used to make the classification, (2) the length of the period of inactivity used to assume that a client has been lost, and (3) the forecast horizon of how many months in advance the classification into lost and retained clients is made. We test the methodology using multiple with machine-learning techniques that existing literature has found well-suited for this type of tasks. Our experimental findings indicate that several methods produce useful classifications as far as three months in advance.

The remainder of the paper is organized as follows: in Section 2 we first discuss related literature, after which we explain detail the calculations done to prepare the input data for the classifiers. We then present the experimental setup and the results we obtain in Section 3. We discuss the implications of the findings in Section 4 and conclude the present work in Section 5.

2. Theory

The concept of *Customer Relationship Management* (CRM) emerged in early twentieth century. CRM helps an organization to gain knowledge that allows it to understand client behavior and relationships better, and hence to improve client acquisition, retention, and profitability (Sabbeh, 2018; Soltani and Navimipour, 2016). Although there is no single, universally accepted definition (Kracklauer et al., 2004; Ngai et al., 2009; Parvatiyar and Sheth, 2001; Swift, 2000), the one given by

* Corresponding author.
E-mail address: elisa.schaeffer@uanl.edu.mx (S.E. Schaeffer).

https://doi.org/10.1016/j.jretconser.2019.101918
Received 5 March 2019; Received in revised form 9 July 2019; Accepted 12 August 2019
Available online 06 September 2019
0969-6989/

Table 1
A comparison of related work in 1998–2018.

Publication	Year	Main technique(s)	Utilized data set(s)
Karp	1998	logistic regression	*no data analysis is reported*
Larivière and Van den Poel	2005	random forest, regression forest	financial services
Coussement and Van den Poel	2008	support vector machine, logistic regression, random forest	newspaper subscriptions
Tsai and Lu	2009	neural networks, self-organizing maps	telecom
Coussement et al.	2010	generalized additive models	newspaper subscriptions
Risselada et al.	2010	logistic regression, classification trees, bagging	internet service provider, health insurance
De Bock and Van den Poel	2011	ensemble techniques	bank, telecom, retail
Ballings and Van den Poel	2012	logistic regression, classification trees, bagging	newspaper subscriptions
Chen et al.	2012	support vector machine	retail, online transactions, telecom
De Bock and Van den Poel	2012	generalized additive models	banking
Verbeke et al.	2012	*numerous*	telecom
Miguéis et al.	2013	logistic regression, multivariate adaptive regression splines	retail
Ali and Arıtürk	2014	logistic regression, decision trees	private banking
Jahromi et al.	2014	decision trees, logistic regression	business-to-business
Ankenbruck	2017	logistic regression, decision trees, neural networks	home-delivery of dairy-product subscriptions
De Caigny et al.	2018	logistic regression, decision trees	financial services, retail, newspaper subscriptions, telecom, energy
Martínez et al.	2018	logistic regression and others	business-to-business

Ling and Yen (2016) is popular: CRM comprises a set of processes and enabling systems supporting a business strategy to build long term, profitable relationships with specific clients.

CRM is a four-stage process with respect to clients: *identification*, *attraction*, *development*, and *retention*. Customer retention refers to actions taken by an organization to ensure client loyalty and avoid client migration, including tasks such as one-to-one marketing, loyalty programs, and complaints management. Retaining existing clients is usually easier and less expensive than gaining new clients (Derby, 2018).

We refer the reader to the review of Sabbeh (2018) for more details on CRM. Also Soltani and Navimipour (2016) present a comprehensive study and survey on CRM mechanisms for 2009–2015, focused on five categories: E-CRM, knowledge management, data mining, data quality, and social CRM.

We start with a review of related work in Section 2.1. Then, in Section 2.2 we describe the data used in the study in terms of quantity, composition, and statistical properties, after which we describe in Section 2.3 our mechanism for determining which of the clients are considered lost, based on the aforementioned statistical findings. We conclude the theoretical discussion by describing the characterization done to the client time series to use as input to the machine-learning algorithms in Section 2.4.

2.1. Related work

Data Mining (DM) refers to collecting, storing, and analyzing data in order to apply statistics and computational methods to derive useful information from said data. In economics, it is often called *business intelligence*. Naturally, DM techniques are widely applied in CRM. Ngai et al. (2009) provide a literature review on the use of DM for CRM in 2000–2006, reporting that active research is largely focused on client retention and that the main elements of client retention include one-to-one marketing, loyalty programs, and complaints management.

Application of *Machine Learning* (ML) on CRM is an active research area. ML offers a variety of computational methods for DM, especially in the area of automated classification, that is, the task of dividing a data set into two or more subgroups. For details on ML, we refer the reader to the textbook of Marsland (2014). As we do not develop any ML approaches in this work, but instead only employ existing methods as provided by commonly-used libraries — with the hope of thus encouraging companies to adopt the suggested practice of forecasting client retention — we do not enter into a detailed discussion of the workings of the ML methods themselves, for brevity and clarity.

For readers unfamiliar with the machine-learning methods applied in this work, they all share the following general principle:

1. Some data of interest is presented, accompanied with pre-established *class labels* — in our case, the data consists of client-transaction time series and each client is labeled as either “lost” or “retained”.
2. The data is *characterized* in some feature space, representing each data entity as a vector of characteristics.
3. A *classifier* is then *trained* by presenting it some of the characteristic vectors. As the classifier produces a suggested class label as output, feedback is given: the internal configuration is adjusted if the suggested label did not match the pre-established class label; this is known as *supervised learning*.
4. The remaining characteristic vectors are used as input but without feedback to *test* the classifier. The resulting labels are then compared with the pre-established ones to establish the performance of the trained classifier.

The internal workings of the classifier for how to determine which label to suggest to a given input and how to adjust in case of errors depend on the machine-learning technique used.

In this section, we discuss the state of the art of applying ML in forecasting client retention. Our interest lies specifically in classifying prepaid client transaction data into clients that will be retained and clients that will be lost *ahead of time*, that is, one or more months before the loss takes place. As such literature is very scarce, we also consider efforts to forecast client retention for consumer settings, subscription-based sales, and regular retail. Table 1 provides a summary of related work over the past two decades, classifying published works that attend the task of forecasting client retention in terms of the machine-learning techniques they employ.

Historically, related work initiates with the use of *logistic regression* (Kleinbaum and Klein, 2010) by Karp (1998) in 1998 and the technique is still in use two decades later (Martínez et al., 2018). The use of *classification trees* (Breiman et al., 1984), in particular *Random Forest* (Breiman, 2001), as well as the first use of a Support Vector Machine (Cortes and Vapnik, 1995) for subscription-based services (Coussement and Van den Poel, 2008; Larivière and Van den Poel, 2005) appear as alternative approaches during the first decade. The second decade is also mostly concentrated on determining subscription renewals such as newspaper and insurance (Ankenbruck, 2017; Ballings and Van den Poel, 2012; Coussement et al., 2010; De Caigny et al., 2018; Risselada et al., 2010) and contractual services such as telecommunications and banking (Ali and Arıtürk, 2014; Chen et al., 2012; De Bock and Van den Poel, 2011; Tsai and Lu, 2009; Verbeke et al., 2012), while a few works focus on consumer retail (Miguéis et al., 2013). The business-to-business aspect is attended to by only Jahromi et al. (2014) and Martínez et al. (2018), but without the prepaid aspect that is essential in our

present work.

In terms of our present contribution, the meaningful insights of existing literature come in terms of what aspects of the data influence the outcome and which methods to employ. As in all application of machine learning, also in the context of forecasting client retention, the performance of any given machine-learning technique depends heavily on the characteristics of the data set (Risselada et al., 2010). Also, given a set of data, the selection of which subset of that data to use also affects the accuracy of the resulting classification (Amin et al., 2019). Sabbeh (2018) compare the performance of several machine-learning techniques for forecasting client retention using data from a telecommunications company. Their results indicate that Random Forest (Breiman, 2001) and ADA Boost (Freund and Schapire, 1997) have the best performance, whereas Support Vector Machine (Cortes and Vapnik, 1995) also achieves a good performance. The length of the time window (i.e., the amount of data used in making the forecast) was found to be logarithmically related to classification performance by Ballings and Van den Poel (Ballings and Van den Poel, 2012). Ankenbruck (2017) finds that detecting retained clients is easier than forecasting churn; in our case study, this is also the case. No existing work known to us carries out client retention forecasts for prepaid business-to-business transactions, for which we have no direct point of comparison for our results.

2.2. Client-transaction time series

We work with client-transaction records of a Mexican company that sells parcel-delivery as a prepaid service. The data fragment starts in January 2014 and ends in April 2017. We extracted the number of prepaid service sales to each client per month as well as the number of service consumptions of each client per month. We only analyze the clients for which both kinds of transactions are registered (the data provided by the company includes many clients that only have consumptions but no registered purchases, implying that those services were purchased through some alternative channel).

We combined the sales counts and the consumptions into a time series describing the client-side inventory of the purchased services: whenever the client purchases p units, the inventory goes up by p, and whenever the client consumes c units, the inventory goes down by c — we locate two data points per month, assuming that the purchase of new units occurs at the beginning of each month and that the consumption of existing units happens mid-month as a simplification. In future work, we are interested in analyzing individual transaction data where the exact time of purchase and/or consumption is known instead of the monthly binning present in the data we presently possess. See Table 2 for a summary of the related notation.

Note that as we have no knowledge of the level of inventory that the client had at the beginning of January 2014, we start at zero. If the client had existing stock in inventory at that point, the time series may later take on negative values; for example, if the client purchased only three units in January 2017 but then went on to consume seven units in that month, the inventory level is at minus four. Once the time series of a client has been extracted, we take the minimum inventory level and, in case it is negative, sum it to each value so as to obtain a non-negative time series for each client. We have transaction time series for $n = 1968$ clients, 1139 of which needed the adjustment for non-negativity, indicating the presence of a stock purchased before 2014.

Table 2
Notation used in the formulation of the problem.

Units purchased by client per time unit	p
Units consumed by client per time unit	c
Data points per month in each time series	2
Total number of clients in data set	n
Maximum length of a time series	k
Length of a useable time-series segment	k^*
Inactivity threshold as a number of data points	ℓ
Length of the an included time-series segment	λ
Length of the forecast horizon	δ
Length of the time-series segment used as input	ξ

Each time series contains at most $k = 81$ data points: the initial stock estimate (zero if the series required no adjustment due to apparent initial inventory) and two data points for each of the 40 months in the recorded period — clients that were not yet active at the start of the examined time interval have less data, as do those that stopped being clients during the time interval. From each time series of $\leq k$ observations, we eliminate any possible initial and final inactivity to obtain a time-series segment of $k' \leq k$ data points. Any clients that have less than three months (six data points) of total activity are excluded from any further analysis. Similarly, if extracting the time-series fragment up until the first inactivity of at least ℓ consecutive data points results in less than six remaining data points, the client is excluded from further analysis; we denote the length of this segment by k^*.

A total of 34 clients have no inactivity whereas the longest inactivity present in the data set corresponds to 62 data points, which is 31 months. Fig. 1 shows examples of time series for three different clients; note that some have clearer periodicity than others, the lengths of the periods differ, and the magnitude of the level of inventory also varies. The first example has no clear periodicity and the first period of inactivity is at the end of the series, whereas the second example includes a short intermediate inactivity and a longer final one; the third example is clearly periodic and contains no periods of inactivity.

During the processing, we also extracted for each client the *longest* period of inactivity for both purchases and consumptions (i.e., the longest amount of consecutive months with no transaction of that type); we consider as a period of inactivity a sequence of months *before* and *after* which there had been transactions — the initial inactivity of a not-yet client and the final inactivity of an already-lost client are *not* included in this analysis, cropping the initial and final months that have no registered activity.

The histograms of the 1525 purchase inactivities and the 882 consumption inactivities as well as that of the 1934 combined periods of inactivity (that is, the client had no transactions of either kind in the course of at least one month) are shown in Fig. 2. In terms of the purchases, 473 clients had no periods of inactivity, whereas in terms of uses 1099 clients had no periods of inactivity — a total of 34 had no inactivity in the combined transaction time series. It is very common for a client to be inactive for a month — 917 occurrences, making up 47 percent of the periods of inactivity in the combined series) — and then to resume purchases and consumptions, but inactivities of eight months or more are not so common (299 cases, 15 percent). In our experiments, we work with inactivities from three to seven months, of which there are 474 cases in the combined series (possibly repeating more than once for a given client), accounting for 25 percent of the periods of inactivity from which the client did in fact return. We excluded the two-month inactivities of which there were 244 (13 percent) as the one-month inactivities were so common so as to limit the “false alarms” in which the marketing department is lead to believe that a client might be lost when in fact they are very likely to reactivate without any intervention.

The preprocessing of the data is done in Python 3.6.2, the statistical treatment in R 3.4.4, and the visualization in Gnuplot 5.0, using Gnu bash 3.2.57 for utilities such as grep, sed, and awk for additional data processing. All of the computational experiments were executed on an iMac with a 4 GHz Intel Core i7 processor running macOS Mojave (10.14.3) with 32 GB 1867 MHz DDR3 main memory.

2.3. Classifying clients as retained or lost

Our goal is to determine which of the clients will sustain transaction activity and which ones will not — in order to classify the clients in this way into those *retained* and those *lost*, we first need to determine a threshold length of combined inactivity ℓ. Fig. 3 shows the proportion

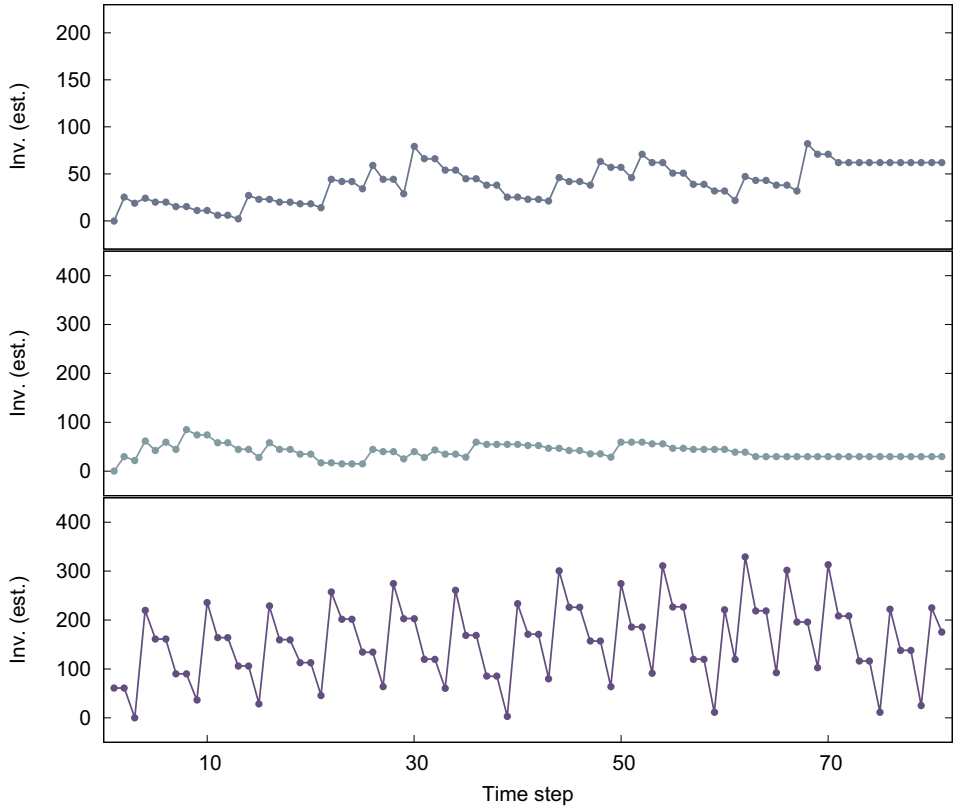

Fig. 1. Examples of the extracted time series of the (estimated) level of inventory for five clients over the k= 81 recorded time steps.

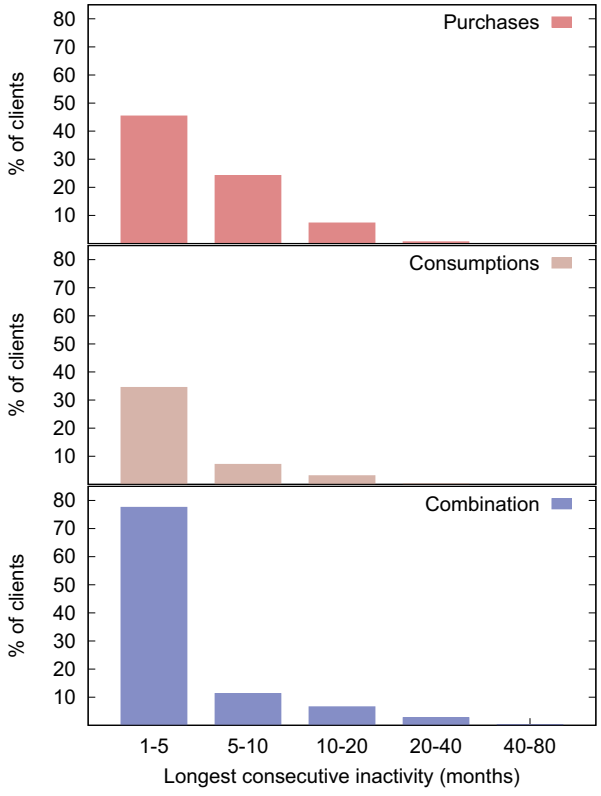

Fig. 2. Histograms of client inactivity.

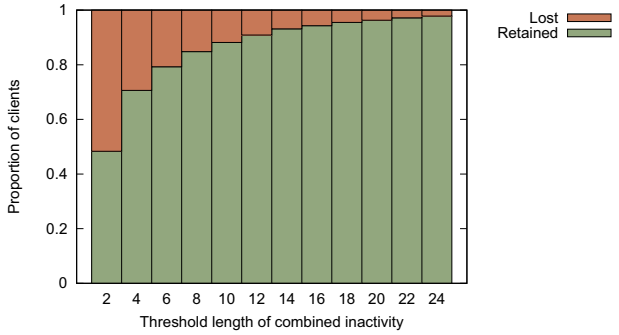

Fig. 3. Proportion of lost versus retained clients.

of clients classified in each group for the different values of ℓ from one month to one year in (recall that each month corresponds to two data points in the extracted time series: one for the purchases and another for the consumption). For the remainder of the present work, we will analyze $\ell \in \{6,8,10,12,14\}$ corresponding to periods from three months ($\ell = 6$) to seven months ($\ell = 14$). As mentioned before, one-month inactivities (two data points) are very common and hence would be a poor choice for a threshold in determining which clients to consider as lost. The decrease in the proportion of lost clients levels off rather quickly after three months (six data points), differences between seven and eight months (14 and 16 data points) already being minimal — hence our decision to experiment with $\ell \in [6,14]$.

We seek to train a supervised machine-learning algorithm to distinguish between lost and retained clients; for the clients classified as lost, we use time-series fragments δ data points *prior* to the *first* period of inactivity of length at least ℓ so that the input for the algorithm is unaware of the period of inactivity that will follow. The motivation behind this is enabling the service provider to forecast which clients are in danger of being lost *before* the client actually becomes inactive. We vary $\delta \in [2,4,6]$ (one and two months) and use time-series segments of length $\lambda \in [6,8,10,\ldots,20,22,24]$ as input to the machine-learning algorithm. We also require that $\lambda - \delta \geq 6$ so as not to analyze any client-

Table 3
Number of qualifying time-series segments for retained (R) and lost (L) clients for each parameter combination.

ℓ	λ	δ	R	L	ℓ	λ	δ	R	L	ℓ	λ	δ	R	L	ℓ	λ	δ	R	L	ℓ	λ	δ	R	L
6	8	2	1455	347	8	8	2	1558	258	10	8	2	1617	199	12	8	2	1657	151	14	8	2	1689	113
6	10	2	1425	324	8	10	2	1534	241	10	10	2	1595	187	12	10	2	1638	140	14	10	2	1670	104
6	10	4	1425	324	8	10	4	1523	241	10	10	4	1579	187	12	10	4	1625	140	14	10	4	1648	104
6	12	2	1391	297	8	12	2	1498	218	10	12	2	1562	172	12	12	2	1612	131	14	12	2	1655	97
6	12	4	1391	297	8	12	4	1497	218	10	12	4	1561	172	12	12	4	1609	131	14	12	4	1642	97
6	12	6	1390	297	8	12	6	1483	218	10	12	6	1545	172	12	12	6	1589	131	14	12	6	1637	97
6	14	2	1337	271	8	14	2	1442	196	10	14	2	1507	157	12	14	2	1559	118	14	14	2	1601	86
6	14	4	1337	271	8	14	4	1442	196	10	14	4	1506	157	12	14	4	1556	118	14	14	4	1596	86
6	14	6	1337	271	8	14	6	1442	196	10	14	6	1504	157	12	14	6	1549	118	14	14	6	1584	86
6	16	2	1215	226	8	16	2	1315	158	10	16	2	1380	125	12	16	2	1431	92	14	16	2	1475	67
6	16	4	1215	226	8	16	4	1315	158	10	16	4	1380	125	12	16	4	1431	92	14	16	4	1473	67
6	16	6	1215	226	8	16	6	1315	158	10	16	6	1380	125	12	16	6	1427	92	14	16	6	1463	67
6	18	2	1191	198	8	18	2	1290	141	10	18	2	1355	114	12	18	2	1406	83	14	18	2	1451	63
6	18	4	1191	198	8	18	4	1290	141	10	18	4	1355	114	12	18	4	1406	83	14	18	4	1451	63
6	18	6	1191	198	8	18	6	1290	141	10	18	6	1355	114	12	18	6	1406	83	14	18	6	1447	63
6	20	2	1165	185	8	20	2	1264	132	10	20	2	1328	109	12	20	2	1379	80	14	20	2	1424	62
6	20	4	1165	185	8	20	4	1264	132	10	20	4	1328	109	12	20	4	1379	80	14	20	4	1424	62
6	20	6	1165	185	8	20	6	1264	132	10	20	6	1328	109	12	20	6	1379	80	14	20	6	1424	62
6	22	2	1132	178	8	22	2	1230	129	10	22	2	1294	104	12	22	2	1345	77	14	22	2	1390	60
6	22	4	1132	178	8	22	4	1230	129	10	22	4	1294	104	12	22	4	1345	77	14	22	4	1390	60
6	22	6	1132	178	8	22	6	1230	129	10	22	6	1294	104	12	22	6	1345	77	14	22	6	1390	60
6	24	2	1103	165	8	24	2	1200	123	10	24	2	1262	98	12	24	2	1313	72	14	24	2	1358	56
6	24	4	1103	165	8	24	4	1200	123	10	24	4	1262	98	12	24	4	1313	72	14	24	4	1358	56
6	24	6	1103	165	8	24	6	1200	123	10	24	6	1262	98	12	24	6	1313	72	14	24	6	1358	56

data spanning less than three months of activity, for which we also exclude higher values of δ in order to maintain the qualifying subsets large enough to train and test classifiers. Table 3 shows the amount of available client data for each case. Note that the number of lost clients varies from as few as 56 to as high as 346, whereas the majority of the clients are retained; there is an order of magnitude of difference in the cardinalities of the two classes.

2.4. Time-series characterization

For the lost clients, we first extract the λ months immediately before the first inactivity initiates and then remove the forecasting horizon of length δ, whereas for the retained clients we select a sub-sequence of length $\lambda - \delta$ at random (we use the *same* pseudo-random selection for all of the parameter combinations so as to be able to compare their performance, as different selections would result in different models when with the same setup). We exclude cases where the sequence maintains a constant value, meaning that there were no transactions of any kind.

We then use 70% of each data set to *train* a classifier and the remaining 30% to evaluate the performance of the resulting classifier. The time-series fragments of length $\xi = \lambda - \delta$ are then characterized as follows to produce the input to the classifier (we do *not* input the raw data to the classifier):

Trend and level We extract the trend by fitting a linear regression model to the segment and obtaining the *slope* and the *intercept*.

Magnitude We use the average inventory level as a characterization of the magnitude of the level of inventory of the client.

Auto-correlations We compute the auto-correlations with all delays from two to eight and use the delay that gave the maximum and the value of that maximum as characteristics.

Fourier coefficients We also include the real and imaginary parts of the first three coefficients of a fast Fourier transform carried out on the time-series segment.

3. Results

We use the R library caret to train classification models. For each parameter combination, we first determine the level of performance of a Support Vector Machine (SVM) (Cortes and Vapnik, 1995) classifier trained to distinguish between lost and retained clients; we use SVM for the initial experiments as it has shown promising performance in the literature (Sabbeh, 2018) and is fast to train. We use a linear and a radial SVM on the training samples and carry out a prediction on the test samples used the trained classifier, as shown in Algorithm 1 — the R code itself is available upon request, as well as anonymized versions of the data with client identities and exact dates removed. As there are many more retained clients than lost clients, we select a subset of them uniformly at random to include in each data set, matching the number of lost clients available for that parameter combination. We also executed preliminary experiments with a polynomial SVM, but the run time was prohibitive for most of the parameter combinations and there was little or no improvement in the accuracy for those combinations that did not result in excessive run times.

Algorithm 1
Training and testing a classifier with balanced data.

input: client data, values of ℓ, λ, and δ
output: a trained classifier for forecasting client retention
1:**procedure**
2: $\mathscr{A} \leftarrow$ client data that contain $\geq \lambda$ consecutive *active* data points ▷set of all clients that have sufficient data
3: $\mathscr{L} \leftarrow$ clients in $\mathscr{A}$ *with* inactivity $\geq \ell$ *after* the active period ▷clients that are assumed to be lost
4: $h \leftarrow |\mathscr{L}|$ ▷number of lost clients for the values of λ and ℓ
5: $\mathscr{R} \leftarrow$ clients in $\mathscr{A}$ with *no* inactivity $\geq \ell$ ▷clients that are assumed to be retained
6: $\mathscr{B} \leftarrow$ a **sample** of size h from $\mathscr{R}$ ▷as most clients are retained, using all of them would bias the training
7: $\mathscr{D} \leftarrow \mathscr{L} \cup \mathscr{B}$ ▷a balanced data set with equal number of lost and retained clients
8: $\mathscr{D}' \leftarrow$ eliminate from each series in $\mathscr{D}$ the last δ data points ▷apply the forecasting horizon
9: $\mathscr{I} \leftarrow$ characteristics of each series in $\mathscr{D}'$ ▷as detailed in Section 2.4
10 attach to $\mathscr{I}$ the *class labels* ▷include for each client which originate from $\mathscr{L}$ and which from $\mathscr{B}$
11 $\mathscr{I}$ ▷center and scale the characteristics
12 $\mathscr{T}_{\text{train}}, \mathscr{T}_{\text{test}} \leftarrow$ randomly divide $\mathscr{I}$ into disjoint subsets ▷we use 70% for training and 30% for testing
13 $\mathscr{C} \leftarrow$ the selected type of **classifier** ▷for the balanced experiments, we use SVM variants

(*continued on next page*)

Algorithm 1 (*continued*)

14: **train** $\mathscr{C}$ using $\mathscr{T}_{\text{train}}$ ▷supervised learning: modify $\mathscr{C}$ if the output does not match the class label
15: **test** $\mathscr{C}$ using $\mathscr{T}_{\text{test}}$ ▷compute the performance measures
16: end procedure

Each trained model is evaluated in terms of the number of true positives T_p (a client was correctly classified as retained), false positives F_p (a client was incorrectly classified as retained), true negatives T_n (a client was correctly classified as lost), and false negatives F_n (a client was incorrectly classified as lost). The total number of positive cases (retained clients) is $C_p = T_p + F_n$ and the total number of negative cases (lost clients) is $C_n = T_n + F_p$. We write $T_t = T_p + T_n + F_p + F_n = C_p + C_n$ as each sample falls into exactly one of the four categories. From this information, we compute

sensitivity $S_e = T_p/C_p$,

specificity $S_p = T_n/C_n$,

accuracy $A = (T_p + T_n)/T_t$,and

$$\text{lift} = \frac{T_p/(T_p + F_p)}{C_p/T_t}$$

as performance measures of the classifier. For each combination, five replicas of the training executed for each of the two model types: linear and radial SVM. For brevity, we only include the results for those replicas that achieved an accuracy of 0.75 or higher in at least one model type and highlight in green the performance measures of those models that we consider well-performing: sensitivity, specificity, and accuracy all at 0.80 or above. Note that all of the models in Table 4 have p-values smaller than 0.005. Even the smallest data set corresponding to $\ell = 14$, $\lambda = 24$, $\delta = 6$ with only 56 lost clients achieved a reasonable accuracy. The only value of ℓ with no well-performing classifier is $\ell = 10$, and even there a radial SVM yields a model with all the performance measures being above 0.79 on the second replica corresponding to $\lambda = 24$ and $\delta = 2$; all values of λ and δ have at least one well-performing classifier.

The most stable case in terms of performance is that with four-month inactivity ($\ell = 8$), using eleven-month time series as input ($\lambda = 22$), and a three-month forecast horizon ($\delta = 6$) where four out of five replicas were above 0.75 accuracy with both model types and the fourth replica has a radial model with S_e, S_p, and A all above 0.80. The results indicate that all of the parameter combinations are viable start points for forecasting client retention even with a data set as small as ours. It is rather common for machine-learning methods to perform the better the more data is available for training, which we also expect to be the case here.

We then trained a second set of models for this specific combination of parameters ℓ, λ, δ, in order to explore a wider range of configurations for the SVM (setting tunelength to 50) and with a different control calculation (optimism bootstrap), using also the polynomial SVM as the combinations are much fewer, making the elevated run time acceptable, ten replicas per model type. The runtime of the linear model was on average 0.8 s with a standard deviation of 0.15, whereas the radial model took on average 1.9 s (with a standard deviation of 0.05), and the polynomial model required 13.7 s on average (standard deviation 0.14); the distributions of the runtimes are shown in Fig. 4.

The accuracies of the ten replicas per each model are shown in Fig. 5; as the accuracy is very similar in the three cases, the medians being nearly identical, the usage of time-consuming options of radial or polynomial variants is unjustified — the fastest option of a linear SVM achieves essentially the same accuracy in much less time. We therefore find no particular necessity for anything but the simplest option, the linear SVM, for this particular data set.

We then analyzed the misclassifications made by the ten replicas of the linear SVM among the 38 clients that were used as testing data (out of the 129 clients of each kind — lost or retained — in that subset), the other 70% having been used for training; each replica used a different pseudo-random assignment of the training and the testing data. The complete data set contains 1230 retained clients with data available to this combination of $\ell = 8$, $\lambda = 22$, and $\delta = 6$, but as there were only 129 lost clients, the data for training was balanced to classes of equal size, as discussed earlier.

A total of 13 clients were mislabeled as lost once (one percent of the true retained clients), 10 clients were mislabeled as lost twice, 12 three times, 7 clients four times, and only one client five times. Fig. 6a shows the complete time series for the client that was misclassified as lost five times out of ten.

In the opposite scenario, classifying clients as retained when they were in fact lost, there were 43 clients mislabeled by one out of then linear SVM models (one third of the actual lost clients), 56 clients (43%) twice, 7 clients three times, 7 clients four times, and one single client six times. Fig. 6b shows the time series for the client that was mislabeled as retained although the preprocessing labeled it as lost. It is clear that the SVM is indeed mistaken and this client is not lost, although the steep drop at the very end of the time series may indicate that further data could well reveal the loss of this particular client at a future time.

Now, in all of the experiments thus far, the imbalance in the classes — there being few lost clients and an overwhelming majority of retained clients — the test data was also drawn from the balanced data set with half lost and half retained clients. We now report the accuracies while training on a balanced data set but testing on *all* of the remaining data, which includes many more retained than lost clients. In this case, instead of reporting the accuracy per se in which the majority class outweighs the minority one, we report the *balanced accuracy* A_b which is the average of the individual class accuracies, sensitivity and the specificity:

$$A_b = \frac{S_e + S_p}{2}, \tag{1}$$

Table 5 reports the values of A_b, S_p (the success rate at classifying clients correctly as lost), and S_e (the success rate at classifying clients correctly as retained) for the data set of $\ell = 8$, $\lambda = 22$, and $\delta = 6$ where classifiers were trained on balanced data but tested with all of the remaining client data executing 30 replicas of testing resulting classifier in order to determine whether the results are stable.

- 30 replicas of a linear SVM using
```
c = trainControl(method = "boot", number = 30)
p = c("center", "scale")
train(…, method = "svmLinear",
  trControl = c, preprocess = p, tunelength = 10)
```
- *k-nearest neighbor* (KNN) (Cover and Hart, 1967) models varying $k \in [2,8]$ (hence 7×10 replicas) with
```
knn3(…, k = k),
```
- Random Forest (RF) models (30 replicas) (Breiman, 2001) with
```
train(…, method = "ranger"),
```
- ADA Boost (ADA; 5 replicas due to elevated runtime) (Freund and Schapire, 1997) with
```
train(…, method = "ada").
```

We include the KNN models as a baseline as they are computationally very simple — anything that performs worse than KNN in terms of accuracy is simply not worth the effort. Random Forest and ADA Boost are included for having performed well in existing literature for similar tasks (Sabbeh, 2018).

Classifying clients correctly as retained is considered the easier of the two tasks, so the measure of major interest is the peak specificity, in which Random Forest is the best-performing option, with SVM in second place. In terms of quality measures (A_b, S_p, S_e), KNN is always

Table 4
For $\ell \in [6,8]$, performance measures of replicas (r) with either the linear or the radial model has $A \geq 0.75$: sensitivity (S_e), specificity (S_p), accuracy (A), lift (L), and the p-value for the result differing from noise. We highlight the A, S_e, and S_e in green when ≥ 0.80; the combinations of ℓ, λ, and δ that have an adequate classifier are highlighted in blue, n being the number of time series n in each data set.

Parameters					*Linear model*									*Radial model*								
					Outcomes				Performance					Outcomes				Performance				
ℓ	λ	δ	r	n	T_p	T_n	F_p	F_n	S_e	S_p	A	L	p	T_p	T_n	F_p	F_n	S_e	S_p	A	L	p
6	18	2	4	198	43	50	9	16	0.73	0.85	0.79	1.65	0.0000	49	50	9	10	0.83	0.85	0.84	1.69	0.0000
6	20	2	4	185	41	45	10	14	0.75	0.82	0.78	1.61	0.0000	39	45	10	16	0.71	0.82	0.76	1.59	0.0000
6	20	4	1	185	41	43	12	14	0.75	0.78	0.76	1.55	0.0000	48	40	15	7	0.87	0.73	0.80	1.52	0.0000
6	20	4	4	185	42	43	12	13	0.76	0.78	0.77	1.56	0.0000	50	38	17	5	0.91	0.69	0.80	1.49	0.0000
6	20	4	5	185	42	45	10	13	0.76	0.82	0.79	1.62	0.0000	42	42	13	13	0.76	0.76	0.76	1.53	0.0000
6	22	2	1	178	45	41	12	8	0.85	0.77	0.81	1.58	0.0000	45	38	15	8	0.85	0.72	0.78	1.50	0.0000
6	22	2	3	178	41	43	10	12	0.77	0.81	0.79	1.61	0.0000	44	40	13	9	0.83	0.75	0.79	1.54	0.0000
6	22	4	3	178	36	44	9	17	0.68	0.83	0.75	1.60	0.0000	42	41	12	11	0.79	0.77	0.78	1.56	0.0000
6	24	2	3	165	36	44	5	13	0.73	0.90	0.82	1.76	0.0000	41	33	16	8	0.84	0.67	0.76	1.44	0.0000
6	24	2	4	165	39	40	9	10	0.80	0.82	0.81	1.62	0.0000	45	36	13	4	0.92	0.73	0.83	1.55	0.0000
8	18	2	1	141	29	36	6	13	0.69	0.86	0.77	1.66	0.0000	31	33	9	11	0.74	0.79	0.76	1.55	0.0000
8	20	2	4	132	32	33	6	7	0.82	0.85	0.83	1.68	0.0000	29	32	7	10	0.74	0.82	0.78	1.61	0.0000
8	22	2	4	129	24	33	5	14	0.63	0.87	0.75	1.66	0.0000	27	32	6	11	0.71	0.84	0.78	1.64	0.0000
8	22	4	1	129	26	33	5	12	0.68	0.87	0.78	1.68	0.0000	31	31	7	7	0.82	0.82	0.82	1.63	0.0000
8	22	6	1	129	28	29	9	10	0.74	0.76	0.75	1.51	0.0000	32	28	10	6	0.84	0.74	0.79	1.52	0.0000
8	22	6	2	129	28	33	5	10	0.74	0.87	0.80	1.70	0.0000	32	29	9	6	0.84	0.76	0.80	1.56	0.0000
8	22	6	3	129	28	32	6	10	0.74	0.84	0.79	1.65	0.0000	28	33	5	10	0.74	0.87	0.80	1.70	0.0000
8	22	6	4	129	26	32	6	12	0.68	0.84	0.76	1.62	0.0000	31	31	7	7	0.82	0.82	0.82	1.63	0.0000
8	24	2	1	123	35	26	10	1	0.97	0.72	0.85	1.56	0.0000	36	22	14	0	1.00	0.61	0.81	1.44	0.0000
8	24	2	2	123	25	29	7	11	0.69	0.81	0.75	1.56	0.0000	28	26	10	8	0.78	0.72	0.75	1.47	0.0000
8	24	4	3	123	27	27	9	9	0.75	0.75	0.75	1.50	0.0000	28	26	10	8	0.78	0.72	0.75	1.47	0.0000
8	24	6	2	123	30	29	7	6	0.83	0.81	0.82	1.62	0.0000	31	28	8	5	0.86	0.78	0.82	1.59	0.0000
8	24	6	4	123	27	30	6	9	0.75	0.83	0.79	1.64	0.0000	27	29	7	9	0.75	0.81	0.78	1.59	0.0000
10	18	2	1	114	25	33	1	9	0.74	0.97	0.85	1.92	0.0000	28	30	4	6	0.82	0.88	0.85	1.75	0.0000
10	18	2	5	114	23	30	4	11	0.68	0.88	0.78	1.70	0.0000	26	29	5	8	0.76	0.85	0.81	1.68	0.0000
10	20	4	4	109	21	30	2	11	0.66	0.94	0.80	1.83	0.0000	20	29	3	12	0.62	0.91	0.77	1.74	0.0000
10	22	2	3	104	24	23	8	7	0.77	0.74	0.76	1.50	0.0000	24	24	7	7	0.77	0.77	0.77	1.55	0.0000
10	22	4	3	104	21	26	5	10	0.68	0.84	0.76	1.62	0.0000	22	25	6	9	0.71	0.81	0.76	1.57	0.0000
10	22	4	4	104	22	28	3	9	0.71	0.90	0.81	1.76	0.0000	24	24	7	7	0.77	0.77	0.77	1.55	0.0000
10	22	6	2	104	20	29	2	11	0.65	0.94	0.79	1.82	0.0000	24	24	7	7	0.77	0.77	0.77	1.55	0.0000
10	22	6	4	104	23	27	4	8	0.74	0.87	0.81	1.70	0.0000	23	26	5	8	0.74	0.84	0.79	1.64	0.0000
10	24	2	2	98	21	25	4	8	0.72	0.86	0.79	1.68	0.0000	23	25	4	6	0.79	0.86	0.83	1.70	0.0000
10	24	4	1	98	20	26	3	9	0.69	0.90	0.79	1.74	0.0000	24	22	7	5	0.83	0.76	0.79	1.55	0.0000
10	24	4	3	98	23	23	6	6	0.79	0.79	0.79	1.59	0.0000	23	23	6	6	0.79	0.79	0.79	1.59	0.0000
10	24	6	2	98	25	20	9	4	0.86	0.69	0.78	1.47	0.0000	25	19	10	4	0.86	0.66	0.76	1.43	0.0001
12	20	2	3	80	20	19	5	4	0.83	0.79	0.81	1.60	0.0000	21	19	5	3	0.88	0.79	0.83	1.62	0.0000
12	20	2	4	80	18	19	5	6	0.75	0.79	0.77	1.57	0.0001	20	18	6	4	0.83	0.75	0.79	1.54	0.0000
12	22	4	2	77	16	19	4	7	0.70	0.83	0.76	1.60	0.0003	20	17	6	3	0.87	0.74	0.80	1.54	0.0000
12	22	4	5	77	15	21	2	8	0.65	0.91	0.78	1.76	0.0001	19	17	6	4	0.83	0.74	0.78	1.52	0.0001
12	22	6	4	77	16	21	2	7	0.70	0.91	0.80	1.78	0.0000	19	19	4	4	0.83	0.83	0.83	1.65	0.0000
12	24	2	1	72	18	18	3	3	0.86	0.86	0.86	1.71	0.0000	18	14	7	3	0.86	0.67	0.76	1.44	0.0005
12	24	4	3	72	16	19	2	5	0.76	0.90	0.83	1.78	0.0000	17	17	4	4	0.81	0.81	0.81	1.62	0.0000
12	24	6	3	72	15	17	4	6	0.71	0.81	0.76	1.58	0.0005	14	19	2	7	0.67	0.90	0.79	1.75	0.0001
14	18	2	2	63	14	16	2	4	0.78	0.89	0.83	1.75	0.0000	15	13	5	3	0.83	0.72	0.78	1.50	0.0006
14	20	2	1	62	13	16	2	5	0.72	0.89	0.81	1.73	0.0002	16	15	3	2	0.89	0.83	0.86	1.68	0.0000
14	20	2	5	62	13	17	1	5	0.72	0.94	0.83	1.86	0.0000	16	13	5	2	0.89	0.72	0.81	1.52	0.0002
14	20	4	2	62	13	15	3	5	0.72	0.83	0.78	1.62	0.0006	16	11	7	2	0.89	0.61	0.75	1.39	0.0020
14	20	4	5	62	13	14	4	5	0.72	0.78	0.75	1.53	0.0020	13	14	4	5	0.72	0.78	0.75	1.53	0.0020
14	22	2	1	60	13	17	1	5	0.72	0.94	0.83	1.86	0.0000	13	16	2	5	0.72	0.89	0.81	1.73	0.0002
14	22	4	4	60	13	17	1	5	0.72	0.94	0.83	1.86	0.0000	18	9	9	0	1.00	0.50	0.75	1.33	0.0020
14	22	6	1	60	13	17	1	5	0.72	0.94	0.83	1.86	0.0000	15	13	5	3	0.83	0.72	0.78	1.50	0.0006
14	24	2	3	56	12	13	3	4	0.75	0.81	0.78	1.60	0.0011	13	12	4	3	0.81	0.75	0.78	1.53	0.0011
14	24	6	2	56	13	12	4	3	0.81	0.75	0.78	1.53	0.0011	14	11	5	2	0.88	0.69	0.78	1.47	0.0011
14	24	6	5	56	13	13	3	3	0.81	0.81	0.81	1.62	0.0003	11	13	3	5	0.69	0.81	0.75	1.57	0.0035

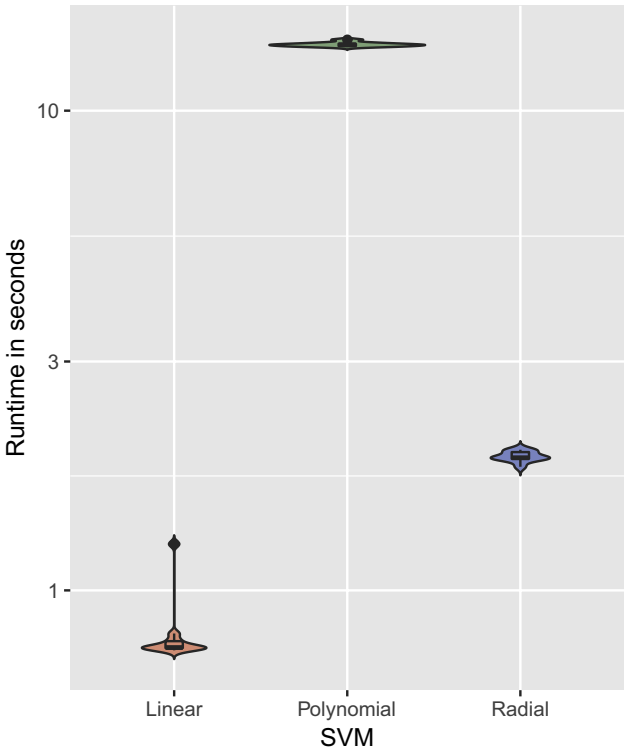

Fig. 4. Runtimes in seconds over ten replicas for the three types of SVM using the subset with $\ell = 8$, $\lambda = 22$, and $\delta = 6$. Note the logarithmic scale on the vertical axis.

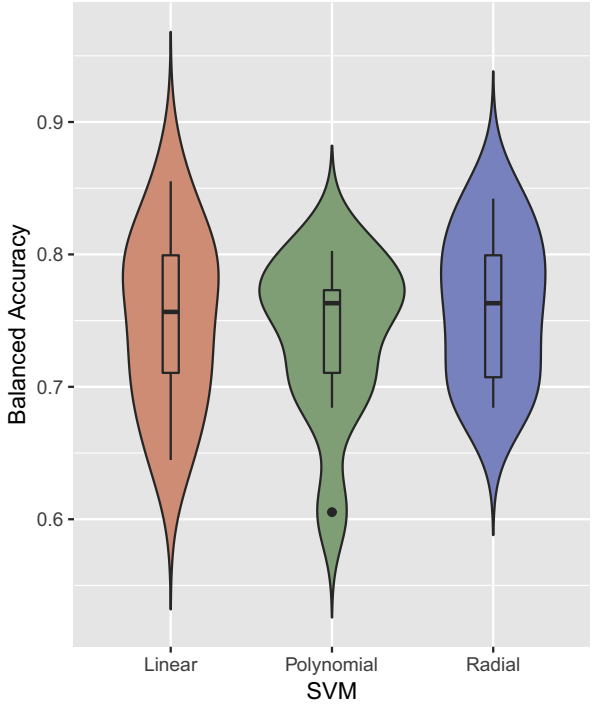

Fig. 5. Accuracies over ten replicas for the three types of SVM using the subset with $\ell = 8$, $\lambda = 22$, and $\delta = 6$.

the worst option, albeit undeniably the fastest, as was expected. Random Forest shows the highest values for balanced accuracy and specificity with a reasonable runtime, whereas ADA Boost has the best sensitivity but takes a minute to train even with these small data sets whereas Random Forest only takes a few seconds and SVM requires approximately 1 s. As the maximum quality measures achieved by SVM are above the minimum values of ADA Boost for both balanced accuracy and specificity, the extra cost in the runtime cannot be justified. We expect the sensitivity of all models to improve if more data is added; it is important to keep in mind that only 129 lost clients exist in total in our data set for the combination of ℓ, λ, δ used in this comparative study.

4. Discussion

For scenarios where client retention needs to be forecast infrequently for small or medium data sets (similar to the one used in this work), our findings indicate that Random Forest is the best option, whereas, for near real-time forecasting for large data sets, SVM may prove adequate and much faster than Random Forest — this latter option makes sense when the data arrives and is analyzed on a minute-by-minute bases, whereas hourly, daily, weekly, or monthly processing allows for more time-consuming models to be trained.

The trained classifiers only work with a characterization of a fragment corresponding to $\lambda - \delta$ data points previous to the *first* inactivity of length $\geq \ell$ (the longest inactivity of the data set is drawn in gray), for which a period with stagnant inventory results in labeling a client as lost, although classifiers appear to often be able to forecast that such clients were not in fact sufficiently similar to lost clients when the client in question indeed reactivates towards the end of the time series. An alternative to the present approach would be working with the *last* period of inactivity instead of the first, especially in scenarios where many clients present a volatile on-off purchasing behavior with frequent extended periods of absence.

Being able to single out clients that may potentially soon be lost with a three-month horizon is of practical commercial interest, as the marketing department has ample opportunity to reach out to the clients with retention-oriented promotions. It is also important to note that in the practice, reaching out to a client that was mistakenly thought to be in danger to be lost is not as serious of an error as not reaching out to one that actually would be lost, so the false negatives for the client-loss category are the main cause of concern. Only having to contact the clients that were labeled to be at risk of loss by the ML algorithm greatly reduces the number of clients that need to be targeted by the marketing department, saving the company time and money.

5. Conclusions

We work with data from prepurchase and usage of unitary services in a business-to-business setting with the goal of identifying potential future client loss with a machine-learning (ML) approach. We train classifiers with time series that estimate the client-side inventory and classify the clients as lost or retained in terms of periods of inactivity in both the usage and the prepurchase of the unitary service provided by the company. We explore different lengths of the time series used as input to the ML algorithm, different lengths for the period of inactivity that label a client as lost, and different forecast horizons, that is, how much in advance the client loss is being detected. We find that a linear SVM performs acceptably for a wide range of the three parameters: the time-series length, the length of the inactivity period, and the forecast horizon. The best performance for predicting lost clients is achieved by Random Forest with a specificity up to 92 percent.

In future work, we hope to train models with larger data sets, enrich the features extracted from the time series describing the client transactions, incorporate other types of client-relation data into the ML inputs, as well as to explore how well our proposed approach extends to business-to-consumer sales such as prepaid cellular services. We are also interested in semi-supervised models and settings in which the model is partially or completely retrained as new data arrives.

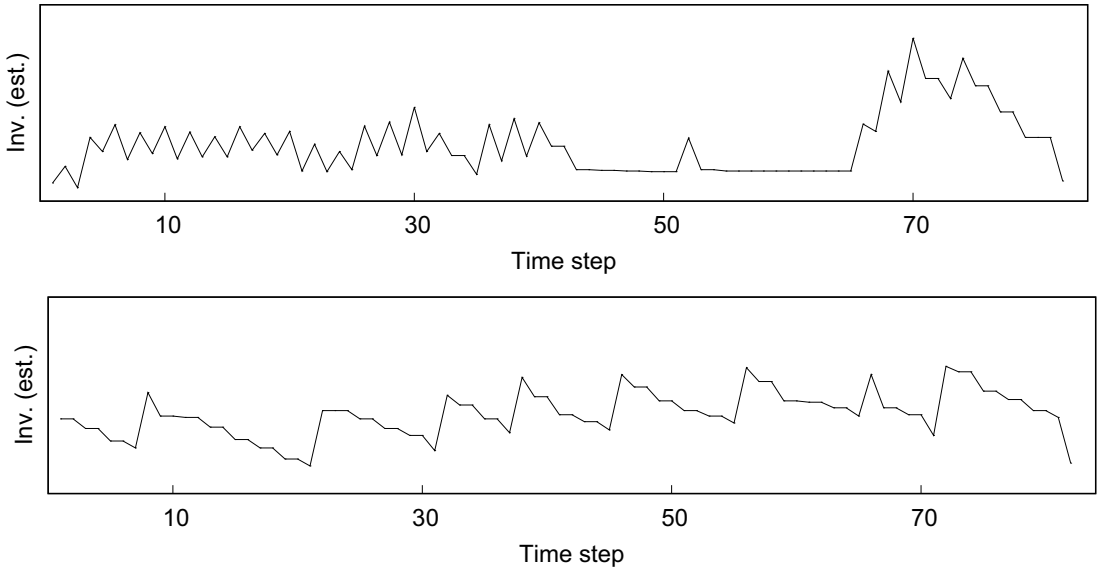

Fig. 6. Time series of clients mislabeled by the classifier with respect to how they were prelabeled by the cut-off rule: a client classified as *retained* but prelabeled as *lost* on the top, and a client classified as *lost* but prelabeled as *retained* on the bottom.

Table 5
A comparison of SVM, KNN, RF, and ADA for the data set with $\ell = 8$, $\lambda = 22$, and $\delta = 6$. The best value of each measure is indicated in boldface and the worst is underlined.

Measure		SVM	KNN	RF	ADA
Balanced accuracy A_b	min	0.680	0.282	0.725	0.739
	max	0.799	0.444	**0.824**	0.796
Specificity S_p	min	0.711	0.158	0.710	0.684
	max	0.842	0.395	**0.921**	0.816
Sensitivity S_e	min	0.642	0.359	0.721	0.716
	max	0.669	0.619	0.805	**0.816**
Runtime (in seconds)	min	0.742	**0.001**	3.632	59.502
	max	1.184	0.002	4.057	60.541

References

Ali, Ö.G., Arıtürk, U., 2014. Dynamic churn prediction framework with more effective use of rare event data: the case of private banking. Expert Syst. Appl. 41 (17), 7889–7903.

Amin, A., Al-Obeidat, F., Shah, B., Adnan, A., Loo, J., Anwar, S., Jan. 2019. Customer churn prediction in telecommunication industry using data certainty. J. Bus. Res. 94, 290–301.

Ankenbruck, S., 2017. Churn the data around — a machine learning approach to understanding why customers leave. In: Conference of the SouthEast SAS Users Group, pp. 191.

Ballings, M., Van den Poel, D., 2012. Customer event history for churn prediction: how long is long enough? Expert Syst. Appl. 39 (18), 13517–13522.

Breiman, L., Oct. 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A., 1984. Classification and Regression Trees, first ed. Chapman and Hall/CRC.

Chen, Z.-Y., Fan, Z.-P., Sun, M., 2012. A hierarchical multiple kernel support vector machine for customer churn prediction using longitudinal behavioral data. Eur. J. Oper. Res. 223 (2), 461–472.

Cortes, C., Vapnik, V., Sep. 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297.

Coussement, K., Benoit, D.F., Van den Poel, D., 2010. Improved marketing decision making in a customer churn prediction context using generalized additive models. Expert Syst. Appl. 37 (3), 2132–2143.

Coussement, K., Van den Poel, D., 2008. Churn prediction in subscription services: an application of support vector machines while comparing two parameter-selection techniques. Expert Syst. Appl. 34 (1), 313–327.

Cover, T., Hart, P.E., Jan. 1967. Nearest neighbor pattern classification. IEEE Trans. Inf. Theory 13 (1), 21–27.

De Bock, K.W., Van den Poel, D., 2011. An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction. Expert Syst. Appl. 38 (10), 12293–12301.

De Bock, K.W., Van den Poel, D., 2012. Reconciling performance and interpretability in customer churn prediction using ensemble learning based on generalized additive models. Expert Syst. Appl. 39 (8), 6816–6826.

De Caigny, A., Coussement, K., De Bock, K.W., 2018. A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. Eur. J. Oper. Res. 269 (2), 760–772.

Derby, N., 2018. Reducing customer attrition with machine learning for financial institutions. In: Proceedings of the SAS Global Forum, pp. 1796.

Freund, Y., Schapire, R.E., Aug. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. 55 (1), 119–139.

Huang, Y., Zhu, F., Yuan, M., Deng, K., Li, Y., Ni, B., Dai, W., Yang, Q., Zeng, J., 2015. Telco churn prediction with big data. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. ACM, New York, NY, USA, pp. 607–618.

Jahromi, A.T., Stakhovych, S., Ewing, M., 2014. Managing b2b customer churn, retention and profitability. Ind. Mark. Manag. 43 (7), 1258–1268.

Karp, A.H., 1998. Using logistic regression to predict customer retention. In: Proceedings of the Eleventh Northeast SAS Users Group Conference, pp. 095.

Kleinbaum, D.G., Klein, M., 2010. Logistic Regression — A Self-Learning Text, third ed. Statistics for Biology and Health. Springer.

Kracklauer, A.H., Mills, D.Q., Seifert, D., 2004. Customer management as the origin of collaborative customer relationship management. In: Kracklauer, A.H., Mills, D.Q., Seifert, D. (Eds.), Collaborative Customer Relationship Management: Taking CRM to the Next Level. Springer, Berlin, Heidelberg, Germany, pp. 3–6.

Larivière, B., Van den Poel, D., 2005. Predicting customer retention and profitability by using random forests and regression forests techniques. Expert Syst. Appl. 29 (2), 472–484.

Ling, R., Yen, D.C., 2016. Customer relationship management: an analysis framework and implementation strategies. J. Comput. Inf. Syst. 41 (3), 82–97.

Marsland, S., 2014. Machine Learning: an Algorithmic Perspective, second ed. CRC Press, Boca Raton, Florida, USA.

Martínez, A., Schmuck, C., Pereverzyev Jr., S., Pirker, C., Haltmeier, M., 2018. A machine learning framework for customer purchase prediction in the non-contractual setting. Eur. J. Oper. Res.

Miguéis, V.L., Camanho, A., e Cunha, J.F., 2013. Customer attrition in retailing: an application of multivariate adaptive regression splines. Expert Syst. Appl. 40 (16), 6225–6232.

Ngai, E.W.T., Xiu, L., Chau, D.C.K., 2009. Application of data mining techniques in customer relationship management: a literature review and classification. Expert Syst. Appl. 36 (2), 2592–2602.

Parvatiyar, A., Sheth, J.N., 2001. Customer relationship management: emerging practice, process, and discipline. J. Econ. Soc. Res. 3 (2), 1–34.

Risselada, H., Verhoef, P.C., Bijmolt, T.H.A., 2010. Staying power of churn prediction models. J. Interact. Mark. 24 (3), 198–208.

Sabbeh, S.F., 2018. Machine-learning techniques for customer retention: a comparative study. Int. J. Adv. Comput. Sci. Appl. 9 (2), 273–281.

Soltani, Z., Navimipour, N.J., Aug. 2016. Customer relationship management mechanisms: a systematic review of the state of the art literature and recommendations for future research. Comput. Hum. Behav. 61, 667–688.

Swift, R.S., 2000. Accelerating Customer Relationships: Using Crm and Relationship Technologies. Prentice Hall, Upper Saddle River, NJ, USA.

Tsai, C.-F., Lu, Y.-H., 2009. Customer churn prediction by hybrid neural networks. Expert Syst. Appl. 36 (10), 12547–12553.

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., Baesens, B., 2012. New insights into churn prediction in the telecommunication sector: a profit driven data mining approach. Eur. J. Oper. Res. 218 (1), 211–229.