Predicting next week redemption in a digital frequency loyalty program

F.T.H. van den Boogaart[1], T. Hantke[2] and M. van Otterlo[1]

[1] Tilburg University, Tilburg, the Netherlands
frederiquevdb@gmail.com, m.vanotterlo@uvt.nl
[2] IceMobile, Amsterdam, the Netherlands
thomas.hantke@icemobile.com

Abstract. Frequency loyalty programs (FLPs) are a vehicle for grocery retailers to increase their pool of loyal consumers. In an FLP, consumers collect stamps until they reach the required number of stamps and trade their full stamp card for reward items. Nowadays, FLPs can be organized digitally, which means that consumers collect, save and redeem stamps in an application on their smartphone. Our goal is to predict whether households which participate in a digital FLP will redeem in a forthcoming week. Redemption refers to the moment a household exchanges their stamps for a reward item.

Our dataset consists of transactional and mobile application activity data which originates from a recent FLP. Several supervised binary classification algorithms are applied to this data. Results show that the random forest algorithm outperforms the other classifiers in terms of AUC score.

Additionally, the most important features and the optimal time window of the input features are explored. Findings from feature importance analyses uncover that variables related to the progression of the consumers' stamp collection and attributes concerned with the number of push messages contribute most to the prediction performance. With regards to optimizing the time window of the input features, we conclude that adding feature values from prior weeks does not significantly influence prediction performance.

Keywords: Loyalty program, Redemption, Predictive Profiling, Customer Relationship Management, Supervised Learning, Marketing.

1 Introduction

Nowadays, consumers spend a substantial amount of their time and income on grocery shopping. As a consequence, grocery retailers continuously focus on enlarging customer loyalty and preventing churn. Churn refers to the moment a customer is leaving a grocery retailer for a competitor (Weiss, 2005). A commonly implemented instrument to achieve loyalty is a loyalty program. The goal of a loyalty program is to stimulate consumer behavior through incentives (Lim & Lee, 2015).

Various types of loyalty programs can be identified in practice and literature. In this study, we focus on short term frequency reward programs in the grocery domain. In a

frequency reward program (FLP), consumers are required to collect stamps through repeated purchases. These stamps can then be traded to redeem products or receive discounts on items (Danaher, Sajtos, & Danaher, 2017; Rossi, 2017). Short term means that the program has a well-defined beginning and ending point. For example, Dutch grocery retailer Albert Heijn recently ran a short term FLP, during which consumers received one stamp for each 10 euros spent in the store. Handing in a full stamp card of twenty stamps gave discounts on high-quality glassware.

In our modern age, retailers can organize FLPs digitally, instead of using traditional paper stamps (Kim, Wang, & Malthouse, 2015). In a digital FLP, consumers obtain, collect and redeem stamps digitally with the use of a mobile application. By introducing a mobile application, retailers are handed the possibility to send push notifications to the consumer. Such notifications are shown as alerts on the home screen of a mobile device, hence the user of the device does not have to explicitly run the application to receive and read the push message.

The goal of this study is to construct a model which predicts whether a household that participates in a digital FLP will trade their full stamp card and redeem an item in a forthcoming week. Data used for model building is transactional and application data collected during a recent digital short term FLP. Transactional data refers to data concerned with purchasing behavior, such as the number of store visits and the amount of money spent. Application data includes, among others, consumers' stamp balance and the number of push notifications sent to them.

At the moment, loyalty program sales and stock projections typically are based on market information and on data of previous or similar loyalty programs. Data from earlier weeks of the running program is not included. Re-projections are solely made once during a program. A model which predicts when consumers are likely to redeem and which features predict redemption provides the opportunity to exploit the moment of redemption in an FLP. For instance, stock levels and replenishments can be planned more accurately and efficiently. Alternatively, special offers can be provided to consumers the week before they will most probably redeem as an attempt to maximize spend levels.

We distinguish three sub tasks. The first sub task focuses on finding the most suitable classification algorithm for our model. To that end, several algorithms are applied to the data and tested on their prediction performance. The second task aims to determine the importance of the input features. A proper understanding of the role of the input attributes may help to comprehend the impact these have on the prediction performance of the model. Lastly, we examine whether varying the length of the time window of the input features affects the performance of our model. This is necessary as the data for this study is collected during the whole program period, which is twenty-one weeks in total. Hence, a time window for the features must be selected.

In the following sections, we will first discuss related work on loyalty programs and prediction models using loyalty program data. Subsequently, a brief explanation of our dataset and the machine learning methods is given. After this, the results and the accompanying discussion are presented. In the last paragraph, a general conclusion and recommendations are provided.

2 Related work

A loyalty program can be defined as "*a concentrated effort by retailers with the aim to enhance basket size, visit frequency and to attract more consumers*" (Allaway, Berkowitz, & D'Souza, 2003, pp. 39-40). Research shows that grocery retailers which have successfully implemented loyalty programs in the past, benefit from increased transactional and emotional consumer loyalty (Ashley, Noble, Donthu, & Lemon, 2011; Meyer-Waarden, 2007). We exclusively focus on transactional loyalty, which deals with purchasing behavior and is measured by spend, frequency and basket size (Breugelmans et al., 2015; Dorotic, Verhoef, Fok, & Bijmolt, 2014). Spend refers to the amount of money spent by a consumer within a certain period of time (Fernandez-Villaverde & Krueger, 2007), while frequency indicates the number of visits to physical store of a retailer. Basket size is concerned with the number of items bought during one visit or over a period of time.

2.1 Loyalty program data

Machine learning methods are a rising global trend within the field of consumer relationship management (CRM) (Ngai, Xiu, & Chau, 2009). However, only a subset of academic articles within the field of CRM and loyalty programs apply machine learning techniques. De Cnudde and Martens (2015) used data from a loyalty card program organized by the public sector in Antwerp, Belgium. They employed machine learning to predict whether a citizen will actively use their loyalty card and whether they will stop using the card within a short period of time. Results show a naïve Bayes classifier performing best for predicting participation and possible future inaction.

Ma et al. (2013) used transactional data from a Chinese public transport loyalty card to model travel patterns of individuals. Travel patterns were modelled by identifying card holders' travelling sequences from the dataset. Afterwards, k-means clustering and rough set theory are combined for classification of travel pattern regularity. Outcomes reveal that the accuracy and efficiency of the rough set algorithm exceeds those of the k-nearest neighbor, a neural network with three hidden layers, decision tree and the naïve Bayes classifier.

In a study by Reutterer et al. (2017), transactional data from households which participate in a loyalty program at a grocery retailer is used to introduce a novel method to extract product recommendations for various consumer segments. K-means clustering and association rule mining are applied to identify segments and derive item sets. The gained item sets then serve as the input in an optimization process to yield profit-maximizing product recommendations for the segments. Findings show that segment-specific promotions lead to a higher increase in profit compared to undifferentiated promotions, assuming the differentiated marketing efforts are effective.

2.2 Frequency reward programs

Frequency reward programs (FLPs) refers to short-term reward programs offered by a grocery retailer with the aim to reward consumers based on their total spend during the running time of the program. The moment at which a consumer trades their points for a reward item is called *redemption.* Dorotic et al. (2014) investigated loyalty program participants' shopping behavior before and after the moment of redemption in an FLP. Results reveal that buying behavior is influenced before and after the moment of redemption, but only when consumers made the decision to redeem before visiting the store. Namely, purchasing frequency and spend amount show a significant increase between the decision moment and actual redemption. This is in line with earlier findings of Kivetz, Urminsky and Zheng (2006) and Taylor and Neslin (2005), which state that consumer spend levels and frequency increase as they reach the point of earning their reward. It must be noted though that the research by Dorotic et al. (2014) is concerned with a long-term rather than a short-term program.

2.3 Digital FLP

Consumers nowadays can use a mobile application to collect, store and redeem stamps of an FLP. In prior research on consumers' spending levels when using a brand's mobile application. Kim et al. (2015) used data from a Canadian *alliance* loyalty program, which means that the program runs at multiple retailer simultaneously. Results indicate that continuous use of the app increases future spending levels. Additional findings show that when a consumer discontinues their use of the app, spending levels tend to decrease again. It is noteworthy to mention that the study by Kim et al. (2015) involves a long term loyalty program, rather than a short term program.

Digital FLPs contain a personal aspect, as it allows grocery retailer to send personalized messages to consumers' mobile devices. One commonly used method is *push messaging*. During an FLP, push notifications may be sent to digital participants when they reach crucial points in their stamp saving process. Wijnen (2017) explored the effect of push messages on loyalty in FLPs in the grocery retail domain. Outcomes of that study indicate that push messages influence transactional loyalty in an FLP. Transactional loyalty is realized when a consumer continues to spend their money and do their grocery shopping at a particular store. Yet, in order to achieve positive loyalty effects, consumers must have a positive attitude towards personal marketing (Tsang, Ho, & Liang, 2004). Furthermore, findings by Andrews, Goehring, Hui, Pancras and Thornswood (2016) uncover that sending push notifications or any other pop-up alerts to mobile devices encourages spontaneous and impulsive purchases. This is due to the fact that personalized promotions are more relevant for consumers as they target their specific needs and interests.

3 Data and experimental setup

Data used for our study originates from an FLP which ran for twenty-one weeks. The last two weeks were clean up weeks, meaning that stamps could still be traded but no

longer be collected. The three types of data used are *transactional*, *application activity* and *push messaging* data. These three types of data are obtained in three separate datasets which later are merged.

It is crucial to mention that 1) data is recorded on household level, and 2) solely data from loyalty card holders is used. Data on household level is used because of the mechanics of the loyalty card at the retailer of interest. Consumers assign themselves to a household, which typically consists of two to ten consumers who live together, or are a family. All household members' activities are tracked and recorded under their *collective* household number.

Solely data of loyalty card holders is used as consumers must log in using their loyalty card number before using the app. Consequently, digital data is only available of households which own a loyalty card. A second reason is that transactions done by a specific loyalty card number can be traced back to a household. Of households who do not own a loyalty card, transactions cannot be connected to one specific household.

3.1 Pre-processing

This section elaborates on the design of the initial retrieved dataset and on how this data is pre-processed in order to obtain our prediction dataset.

Transactional data. In the transaction dataset, each row represents one distinct transaction. Entries of consumers who have not participated in the program digitally or do not own a loyalty card are discarded.

Since our overall goal is to predict whether a digitally participating household will redeem in the upcoming week, all data must be aggregated on week-level. Therefore, the transaction data is aggregated in such a way that each row now represents one week of a household. Examples of features in the transactional data are number of store visits per week (*Frequency*), average spend level per visit, total spend level, average number of articles bought per visit (*Total quantity*) and total number of articles bought during that week

Application activity data. Application activity refers to the process of collecting stamps digitally. This dataset is designed in such a way that it displays the app data per household number per week.

As data was already extracted on week level, no further aggregation is required. Features in the application activity dataset are, among others, stamp balance at the end of a week, number of collected stamps during that week and number of redeemed stamps during the week.

Push notification data. In the original dataset, each instance represents one push message sent to a particular household. Exclusively push messages with status *sent* are kept, since messages which failed to be sent were never received by app users.

Identical to the transaction dataset, push messaging data is aggregated on week level. Thus, each row states the number of push messages which were sent to a household in

that specific week. Moreover, the cumulative number of push messages sent to a household up to that specific week is added as a column. It is noteworthy to mention that the content of the push messages is not being studied.

Merging and finalizing the dataset. After aggregation, the three datasets are ready to be merged into one bigger data frame. The application activity dataset is joined with the transactional and push datasets on household number and week number. The app activity dataset is leading, since this table is necessary to be able to create labels from the data. Namely, the app activity contains information on when households have redeemed and this information is needed to determine whether a consumer redeems in the upcoming week.

Labels are created by generating a binary feature named *Redeems_Nextweek*. This feature states whether the household will redeem in the upcoming week. This target feature is created using existing features which state the number of redeemed stamps and items in a particular week. After the label is created, these existing features are removed from the dataset.

The merged dataset contains some outliers. Outliers in the dataset were found by plotting distributions of the variables. To illustrate, for one household the total money spent at the retailer in one week equals $ 46 069.45 and for another household the number of visits in a week counts 57 times. These values typically are a result of the fact that cashiers may scan their own loyalty card when consumers forgot to bring theirs. Because outliers may significantly harm the machine learning process, households which contain outliers are omitted from the dataset.

If a household registered in the digital saving app when the program has already started, the program weeks corresponding to that household prior to registration are filled with zeros. Of those households which registered after the start of the program, data on how many stamps they collected or redeemed before registration is lacking.

Lastly, of each household the row corresponding to the last program week is omitted. That is, our task is to predict next week redemption, and after the last program week no further redemptions will take place.

3.2 The prediction problem

The final dataset contains the features listed in Table 1. As can be seen from the table, there are 18 input features accompanied by 1 target feature. The total number of rows in the final dataset equals 156 567. Our main goal is to predict whether a household will redeem in a forthcoming week. Thus, a model must be constructed which predicts whether *Redeems_Nextweek* holds value *0* or *1* for a particular week of a household *given* the input features.

Table 1. Features in the final dataset. The target feature is printed in bold.

Feature name	Description
DigitalCollectedStamps	Number of collected stamps (per week)
DigitalCollectedStampsCml	Number of collected stamps (cumulative)

DigitalStampsBalance	Stamp balance at the end of the week
DigitalStampsBalanceMutation	Difference between the stamp balance at the beginning and the end of the week
DigitalRewardsCml	Number of redeemed items (cumulative)
DigitalRedeemedStampsCml	Number of redeemed stamps (cumulative)
Push	Number of push messages sent to a household (per week)
Push_Cumulative	Number of push messages sent to a household (cumulative)
Qty_total	Total number of articles bought (per week)
Qty_mean	Mean number of articles bought (per week)
Qty_min	Min number of articles bought (per week)
Qty_max	Max number of articles bought (per week)
Spend_total	Total amount of money spent (per week)
Spend_mean	Mean amount of money spent (per week)
Spend_min	Min amount of money spent (per week)
Spend_max	Max amount of money spent (per week)
Frequency	Number of visits to the store (per week)
Redeem_Potential	Number of full stamp cards (cumulative)
Redeems_Nextweek	**Whether a household will redeem an item in the next week**

3.3 Models and evaluation

Our first task is to identify the best performing algorithm for the prediction model. Since the dataset is labeled (i.e., redeems next week 1 or 0) and the outcome variable is categorical and binary, we are concerned with a supervised binary prediction task. As it is well-known that various machine learning methods have their strengths and weaknesses (Caruana & Niculescu-Mizil, 2006), we apply several different models to find the one that is best for our dataset. The classifiers we will apply are Gaussian naïve Bayes, support vector machine (SVM), decision tree, random tree forest, logistic regression and a multilayer perceptron (MLP).

Referring to the algorithms, for each of the classifier the parameters are trained and optimized by the algorithm. Besides, there are hyperparameters which are chosen by the authors based on initial explorations of the dataset to set them appropriately. Table 2 lists the used algorithms the accompanying hyperparameter settings.

Table 2. Applied algorithms and accompanying hyperparameter settings.

Algorithm	Hyperparameter settings

Gaussian naïve Bayes	*NA*
Support vector machine	Kernel: Linear, Radio based function (RBF), Sigmoid, Polynomial *Maximum number of iterations of polynomial kernel is set at 10 million, due to time-wise computational restrictions*
Decision tree	Impurity: Entropy, Gini Maximum tree depth: 3, 4, 5, 10
Random forest	Impurity: Entropy, Gini Maximum tree depth: 3, 5, 6, 7, 10 Maximum number of trees: 10
Logistic regression	Solver: Liblinear, Broyden-Fletcher-Goldfarb-Shanno (LBFGS), Newton-conjugate gradient (Newton-CG), Stochastic gradient descent (SGD) Alpha *(when solver is SGD)*: 0.10, 0.20, 0.50
Multilayer perceptron	Hidden layer size: 10, 50, 100, 200, 300, 400, 500 Activation function: Identity, Logistic, Tanh, Relu Solver: LBFGS, SGD, Adam Alpha: 0.0001, 0.001, 0.01, 0.1, 0.2, 0.3, 0.5 Learning rate *(when solver is SGD)*: Constant, Invscaling, Adaptive

In order to ensure generalization of the classification model and to avoid overfitting, we partition the data into a training and a test set. To avoid a selection bias and ensure the representativeness of the cases in the training and validation set in the whole dataset, 10-fold cross validation is applied.

Our dataset suffers from a severe class imbalance as only 5.29% of the instances has class label *1*, while 94.71% is labelled with *0*. We are interested in the minority class, as our goal is to predict next week redemption. To deal with class imbalance, we apply the method of under-sampling. Under-sampling is only applied to the training set, since the distribution of class labels in the test set must stay as close to reality as possible. With respect to cross validation, for each fold under-sampling is solely applied to the training set. Another means to overcome the class imbalance problem is over-sampling. However, we use under-sampling due to the fact that with over-sampling computation time increases as the number of instances increases, whereas no additional information in the dataset is obtained. The number of instances in the training set after applying under-sampling equals 13 222.

Prediction performance of the algorithms will be measured using area under the curve (AUC) score, which measures to what extent the classifier establishes a perfect discrimination between classes. AUC scores range from 0.5 to 1, with 1 for a perfect discriminating classifier and 0.50 for a random classifier. A distinct reason to use AUC score is based on findings of Huang and Lin (2005). They found that the coefficient of the profit curve is steeper for AUC than for less advanced metrics, such as accuracy. As loyalty program providers typically aim at increasing profits, the focus should thus lie on improving AUC score.

To compare the results of the algorithms two *baseline models* are created. The first baseline model is based on the moment a household has a full stamp card. More specifically, the week after they completed a stamp card, a moment of redemption is predicted. The second baseline is based on insights from a global loyalty company. Consumers do not redeem an item directly after they completed a stamp card, but wait a number of weeks. The difference between the moment a household has a full stamp card and the subsequent moment of redemption is used as baseline. To establish this baseline, the difference in weeks is calculated for each household. The mean difference in weeks is 7.0928, hence households on average redeem in the eighth week, counting from the week they complete a stamp card. This will be the prediction of our second baseline model.

3.4 Feature importance

Our second task is to determine feature importance. To do so, first, a metric which is not related to a certain algorithm is used to gain an overall understanding and to provide a basis for establishing the contributions of features. The metric used is the *point-biserial correlation coefficient* (r_{pb}), since the target feature in our dataset is binary and the input features are numeric. The r_{pb} measures the degree of relatedness between one input feature and the target in isolation. Values of the r_{pb} range from -1 to 1, where -1 indicates a perfect negative correlation and 1 represents a perfect positive correlation (Kornbrot, 2005). The higher the absolute value, the higher the correlation between features.

In addition, feature importance as identified by the individual classification algorithms is used. Attention should be paid to the fact that not all algorithms allow for determining feature importance directly. Only algorithms which do allow for direct extraction of feature importance are incorporated, due to time-wise computational limitations of this research. In particular, naïve Bayes is not included here as it assumes independence and equal importance of features (Lee, Gutierrez, & Dou, 2011). Next to that, because of the many connections and multiple layers of the neural network, feature importance cannot be deduced for the MLP either without using additional frameworks.

The algorithms used to explore feature importance are SVM, decision tree, random forest and logistic regression. The decision tree and random forest use Gini importance to determine feature importance. For SVM and logistic regression feature importance is represented by the attribute weights. The higher the absolute value of a weight, the more important the feature is in making a prediction. It should be emphasized that values of Gini importance and attribute weights cannot directly be compared, because Gini importance measures absolute importance while the weights of logistic regression and SVM provide a relative importance. Therefore, we will use the ranking of feature importance by the algorithms rather than the given values.

Lastly, using the feature importance as assigned by the algorithms, the top 5 of each algorithm is selected. Then, for each algorithm a model is created including solely this top 5. Comparing the performance of this small model to the full model with all features provides insights in whether this top 5 is sufficient for the model to predict. Once these small models are generated, feature ablation is applied to each of the models. We follow

the ablation process of leaving out one feature at the time to observe the change in performance. The feature whose removal leads to the highest decrease in AUC score shows to be the most important.

3.5 Impact of time window

Our third task is to examine whether predictions can become more accurate when based on features of previous weeks, e.g. a longer time window. So far, each row in the dataset contains data from one week of a household. To investigate the impact of the time window, lag features are added to each instance. A difficulty with creating lag features is to determine the number of prior time steps to be included. This often is domain specific. Findings from earlier research show that shopping behavior alters as soon as consumers reach the point of redemption (Dorotic et al., 2014; Kivetz et al., 2006; Taylor & Neslin, 2005). Therefore, a smaller time window of a few weeks seems most appropriate.

To decide on the exact number of weeks, the distribution of when households redeem their first item is plotted in Fig. 1. It shows that during the first two weeks of the program almost no redemptions took place. From week 3 on the number of first redemptions starts growing. Based on this and on the fact that household data of one week before the program started is available, we decided to create lag features from 1 until 3 prior weeks (T-1 till T-3).

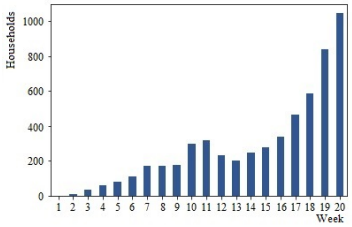

Fig. 1**.** Week during which a household redeems their first item(s).

Moreover, in existing literature it is concluded that shopping behavior changes when a consumer reaches their moment of redemption (Dorotic et al., 2014; Kivetz et al., 2006; Taylor & Neslin, 2005). To measure the change in shopping behavior, differential features are created using the lag features. Differential features represent the change in feature values over a period of time.

To test with which time window the model performs best, three additional datasets are created. Each dataset covers a particular time window, ranging from Week T till T-1 to Week T until T-3. Each of the datasets will run through the selected model as identified in our first task to measure prediction performance according to the specified performance metrics. Later, the prediction performances are compared to discover the impact of the various time windows.

4 Results

Results of the three defined tasks are presented in this section.

4.1 Classification

Results. To find the most suitable algorithm for our prediction task, the classifiers as listed in the previous section are applied to the data and their predicting performance is measured. The performances of the algorithms are compared to two baseline models. The AUC score of the first baseline model equals 0.6343, thus this naïve rule performs poor on distinguishing between classes. The second baseline model gains an AUC score of 0.5302. A score close to 0.5 indicates that this baseline is an almost completely random classifier.

In Table 3 the best AUC scores of the multiple algorithms with their hyperparameter settings are listed from highest to lowest. This table reveals that all applied classifiers perform better than the baseline models. In addition, it discloses that the random forest algorithm outperforms the other algorithms in terms of the AUC score.

Table 3. Ranking of the best AUC scores of each algorithm.

AUC score	Algorithm	Hyperparameter settings
0.7756	Random forest	Impurity: Gini Max depth: 6 Max number of trees: 10
0.7740	Multilayer perceptron	Hidden layer size: 400 Activation function: Logistic Solver: Adam Alpha: 0.001
0.7719	Decision tree	Impurity: Gini Max depth: 4
0.7590	Logistic regression	Solver: Liblinear
0.7230	Gaussian naïve Bayes	*NA*
0.6751	Support vector machine	Kernel: Linear

Discussion. Naïve Bayes and SVM are the two worst performing classifiers. This is surprising as these classifiers acquired the highest AUC scores in a study by De Cnudde and Martens (2015), when predicting future behavior in a loyalty program. According to the *no free lunch* theory the phenomenon of random forest performing best is due to the underlying dataset (Caruana & Niculescu-Mizil, 2006). Notwithstanding, the random forest algorithm has shown to perform at least reasonable within a wide range of domains (Liaw & Wiener, 2002). While this does not provide support for the fact that random forest outperforms all other algorithms, it does demonstrate that the random forest performing well is not a surprise per se. Apart from this, a random forest is known for its variance reduction and robustness as it iterates over multiple trees. As a result,

prediction performances of random forests typically exceed those of single decision trees (Han, Kamber, & Pei, 2016).

4.2 Feature importance

Results. Looking at the point-biserial correlation coefficients of each input feature and the target feature, we see that the three features which gain the highest coefficients are related to the number of stamps a household has collected. These features are respectively *DigitalCollectedStampsCML* ($r_{pb} = 0.2389$), *DigitalStampsBalance* ($r_{pb} = 0.2230$) and *Redeem_Potential* ($r_{pb} = 0.2119$). The fourth and fifth features in the ranking are respectively *Push_Cumulative* ($r_{pb} = 0.1478$) and *Push* ($r_{pb} = 0.1290$).

Regarding feature importance assigned by the algorithms, those hyperparameter settings which gained the highest AUC score are used. In accordance with results of the point-biserial correlation coefficient, all four algorithms selected a most important feature associated with the number of collected stamps. Particularly the random forest algorithm agrees with the correlation coefficients, as both their selected top four encompass the exact same feature ranking. Another observation is that the importance scores of the first and second most important features only differ marginally. The third, fourth and fifth most important features in general have a significantly lower score compared to the first and second.

To determine whether the top five features assigned by the algorithms are sufficient for prediction, additional models are made which solely contain the five most important features. While for the decision tree, random forest and logistic regression no or minimal changes in AUC score are observed, the AUC score of the SVM increased to a value comparable to those of the other classifiers.

Applying feature ablation to each of the top five models, a general observation is that for each of the algorithms there are two or three features which have a larger effect on the performance compared to the other features. Overall, features which contribute most to a model are *DigitalStampsBalance*, *DigitalCollectedStampsCml, Push* and *Push_Cumulative*, which is in accordance with outcomes of the point-biserial correlation coefficients.

Discussion. Overall, results show that features related to the saving process are deemed most important. Intuitively this makes sense as the number of stamps a household has collected so far and the amount of full stamp cards they currently possess indicate whether a household can redeem an item in the first place. When they do not have sufficient stamps yet, the opportunity to redeem most probably is absent.

Aside from this, attributes indicating the number of push messages sent to a household reveal to be the second most important for prediction. Results of Wijnen (2017) and Andrews et al. (2016) specify that push notifications may positively influence transactional loyalty. This implies that consumers receiving push messages on average spend more than they would have in case they did not receive push alerts. This provides a possible reason for the fact that the number of sent push notification to a household shows to be an important indicator in the model. Namely, a higher spend level leads to

more collected stamps. As mentioned above, the number of collected stamps and full stamp cards are crucial features in the prediction model.

4.3 Impact of time window

Results. To determine the optimal time window, we employ our best performing model, the random forest classifier. Various maximum tree depths are tried, since there is no universal way of determining the optimal depth. The highest AUC score is obtained using a time window of T-1 and setting a maximum depth of 9. This AUC score equals 0.7761 and realizes a minor increase of 0.0005 compared to the highest AUC score on the regular dataset. Alternative time windows or maximum depth settings did not improve the performance.

Discussion. As enlarging the time window appears to barely influence the AUC score, it can be concluded that the random forest algorithm is not affected by changing time windows. Outcomes of Dorotic et al. (2014), Kivetz et al. (2006) and Taylor and Neslin (2005) indicate that consumer behavior alters only right before the moment of redemption takes place. These results might explain why using data from the week prior to redemption is sufficient for prediction and that adding data from additional prior weeks does not improve the prediction.

Nonetheless, the fact that taking a larger time window does not influence the performance of the random forest does not mean that data covering a certain period is useless. To illustrate, results of the feature importance investigation reveal that *DigitalCollectedStampsCml and DigitalStampsBalance* are two fundamental features. *DigitalCollectedStampsCml* implicitly holds information from previous weeks, as each consecutive week the amount of collected stamps is added to this total. Likewise, in the feature *DigitalStampsBalance* information from prior weeks is incorporated implicitly. That is, the number of traded and collected stamps are respectively distracted from and added to the balance of a prior week. Therefore, completely discarding data collected over a period larger than a week is not recommended.

5 Conclusions and Future Research

The goal of this study has been to construct a model which predicts whether a household that participates in a digital FLP will trade their full stamp card and redeem an item in a forthcoming week. To that end, data from a recent FLP at a grocery retailer is used and a binary classification task is performed.

A first observation is that all selected algorithms performed better at predicting next week redemptions compared to simple rules. From the applied algorithms, the random forest outperformed the other classifiers. Decision tree, logistic regression and neural network obtain slightly lower performance, but still performed well.

With regards to feature importance, we found that features related to the stamp collection process and the number of sent push messages can be considered most important for predicting whether a household will redeem in a forthcoming week. Furthermore,

findings from time window optimization state that the random forest's prediction performance is not influenced by expanding the time window of the features.

In short, as a conclusion, it can be said that our best model performs fair on discriminating between households which will redeem in an upcoming week and those that will not redeem next week. This best model encompasses the random forest classifier, which obtained an AUC score of 0.7756. The five most important features as assigned by the random forest classifier show to be sufficient for prediction. The other features can be omitted from the model, while the prediction performance remains similar. A last general conclusion is that adding feature values from prior weeks does not improve the prediction performance. Hence, feature values corresponding to a current week are sufficient for prediction.

As for future work, we note that our scope was limited in terms of exploring possibilities to maximize the predictive power of our model. To start, while seven different algorithms with multiple hyperparameter settings have been applied to the data, still more algorithms exist which might be suitable for our prediction task. For example, given that the random forest and decision tree obtained relatively high AUC scores, exploring alternative ensemble methods might be worthwhile. A second direction of future research is to add an extra layer to our model. An implicit assumption we make is that households are going to redeem at least once during the program, while in fact there might be households which do not redeem an item at all. This leads to a large number of false positives. A recommendation therefore is to first create a model which predicts whether a household is likely to redeem at least one item, and then apply our model only to those that are likely to redeem. Another, related, suggestion is to solely predict whether a household will redeem in a next week in case that particular household has a full stamp card. Particularly, from our feature importance analyses it can be concluded that *DigitalStampsBalance* contributes significantly to the model. If that feature value is too low and hence there is no full stamp card, there is no possibility for redemption. A final recommendation deals with the time window of the features. Although we found that adding lag and differential features does not influence prediction performance, cumulative and balance features have proven to be important for prediction. Optimizing time windows did not lie within the scope of this research, but investigating this in more detail might result in a better predicting model.

References

1. Allaway, A. W., Berkowitz, D., D'Souza, G.: Spatial diffusion of a new loyalty program through a retail market. Journal of Retailing 79(3), 137-151 (2013).
2. Andrews, M., Goehring, J., Hui, S., Pancras, J., Thornswood, L.: Mobile promotions: A framework and research priorities. Journal of Interactive Marketing 34, 15-24 (2016).
3. Ashley, C., Noble, S. M., Donthu, N., Lemon, K. N.: Why customers won't relate: Obstacles to relationship marketing engagement. J. of Business Research 64(7), 749-756 (2011).
4. Breugelmans, E., Bijmolt, T. H., Zhang, J., Basso, L. J., Dorotic, M., Kopalle, P., Minnema, A., Mijnlieff, W. J., Wünderlich, N. V.: Advancing research on loyalty programs: a future research agenda. Marketing Letters 26(2), 127-139 (2015).
5. Caruana, R., Niculescu-Mizil, A.: An empirical comparison of supervised learning algorithms. In: 23rd International Conference on Machine learning, pp. 161-168. ACM (2006).

6. Danaher, P. J., Sajtos, L., Danaher, T. S.: Does the reward match the effort for loyalty program members?. Journal of Retailing and Consumer Services 32, 23-31 (2016).
7. De Cnudde, S., Martens, D.: Loyal to your city? a data mining analysis of a public service loyalty program. Decision Support Systems 73, 74-84 (2015).
8. Dorotic, M., Verhoef, P., Fok, D., Bijmolt, T.: Reward redemption effects in a loyalty program when customers choose how much and when to redeem. International Journal of Research in Marketing 31(4), 339-355 (2014).
9. Fernández-Villaverde, J., Krueger, D.: Consumption over the life cycle: Facts from consumer expenditure survey data. The Review of Economics and Statistics 89(3), 552-565 (2007).
10. Han, J., Kamber, M., Pei, J.: Data mining: Concepts and techniques. Elsevier/Morgan Kaufmann, Amsterdam (2012).
11. Huang, J., Ling, C. X.: Using AUC and accuracy in evaluating learning algorithms. IEEE Transactions on knowledge and Data Engineering 17(3), 299-310 (2005).
12. Kim, S., Wang, R., Malthouse, E.: The effects of adopting and using a brand's mobile application on customers' subsequent purchase behavior. Journal of Interactive Marketing 31, 28-41 (2015).
13. Kivetz, R., Urminsky, O., Zheng, Y.: The goal-gradient hypothesis resurrected: Purchase acceleration, illusionary goal progress, and customer retention. Journal of Marketing Research 43(1), 39-58 (2006).
14. Kornbrot, D.: Point biserial correlation. Wiley StatsRef: Statistics Reference Online (2005).
15. Lee, C. H., Gutierrez, F., Dou, D.: Calculating feature weights in naive Bayes with kullback-leibler measure. In: Int. Conference on Data Mining, pp. 1146-1151. IEEE (2011).
16. Liaw, A., Wiener, M.: Classification and regression by randomForest. R news 2(3), 18-22 (2002).
17. Lim, S., Lee, B.: Loyalty programs and dynamic consumer preference in online markets. Decision Support Systems 78, 104-112 (2015).
18. Ma, X., Wu, Y. J., Wang, Y., Chen, F., Liu, J.: Mining smart card data for transit riders' travel patterns. Transportation Research Part C: Emerging Technologies 36, 1-12 (2013).
19. Meyer-Waarden, L.: The Effects of Loyalty Programs on Consumer Lifetime Duration and Share of Wallet. Journal of Retailing 83(2), 223-236 (2007).
20. Ngai, E., Xiu, L., Chau, D.: Application of data mining techniques in customer relationship management: A literature review and classification. Expert Systems with Applications: Part 2 36(2), 2592-2602 (2009).
21. Reutterer, T., Hornik, K., March, N., Gruber, K.: A data mining framework for targeted category promotions. Journal of Business Economics 87(3), 337-358 (2017).
22. Rossi, F.: Lower price or higher reward? Measuring the effect of consumers' preferences on reward programs. Management Science Articles in Advance, 1-20 (2017).
23. Taylor, G. A., Neslin, S. A.: The current and future sales impact of a retail frequency reward program. Journal of Retailing 81(4), 293-305 (2005).
24. Tsang, M. M., Ho, S. C., Liang, T. P.: Consumer attitudes toward mobile advertising: An empirical study. International journal of electronic commerce 8(3), 65-78 (2004).
25. Weiss, G. M.: Data mining in telecommunications. In: Data Mining and Knowledge Discovery Handbook, pp. 1189-1201. Springer, Boston, MA. (2005).
26. Wijnen, F.: Pushing Toward Consumer Loyalty: The Impact of Mobile Push Notification on Emotional and Transactional Loyalty (Master's thesis). Faculty of Leadership and Management, University of Amsterdam, Amsterdam, the Netherlands (2017).