Predicting Product Returns in E-Commerce: The Contribution of Mahalanobis Feature Extraction

Completed Research Paper

Patrick Urbanke
University of Göttingen
Platz der Göttinger Sieben 5
37073 Göttingen
patrick-axel.urbanke@wiwi.uni-goettingen.de

Johann Kranz
University of Göttingen
Platz der Göttinger Sieben 5
37073 Göttingen
jkranz@wiwi.uni-goettingen.de

Lutz Kolbe
University of Göttingen
Platz der Göttinger Sieben 5
37073 Göttingen
Lutz.Kolbe@wiwi.uni-goettingen.de

Abstract

Product returns are a major challenge in e-commerce that severely affect the economic and ecological sustainability of the industry. While many static one-size-fits-all approaches to limit product returns have been proposed, there is a gap in the literature regarding strategies based on individual consumption patterns. We introduce a decision support system for the prediction of product returns, including a new approach for large-scale feature extraction. This system can be used as the basis for a returns strategy that allows online retailers to intervene before problematic transactions even take place. Using a dataset containing 1,149,262 purchases obtained from a major German online retailer, we demonstrate that our decision support system can identify consumption patterns associated with a high product return rate at sufficient accuracy for such a strategy to be feasible. We also show that the system outperforms a wide selection of state-of-the art classification and dimensionality reduction algorithms.

Keywords: Machine learning, e-commerce, online retail, product returns, statistical methods, business intelligence, decision support systems,

Introduction

Even though e-commerce has become an important industry that is growing rapidly, many online retailers fail to be profitable (Rigby, 2014). One important reason for this are product returns, which constitute a considerable cost factor. A significant portion of online retailers report that lowering the rate of product returns by 10% could increase profitability by over 20% (Pur et al., 2013). The overall return rate varies, but for online retailers specializing in fashion, it is often higher than 50% of all purchases (Asdecker, 2015).

Product returns are also an important factor contributing to the carbon footprint of e-commerce as a business model. Several studies have evaluated whether e-commerce is environmentally friendlier than traditional retailers, with inconclusive results (Fuchs, 2008; Weber et al., 2008; Williams and Tagami, 2002). It is widely recognized that the "last mile"of the delivery chain (delivering the product to the customer's doorstep) is the most energy intensive (Browne et al., 2008; Halldórsson et al., 2010; Song et al., 2009). Therefore, reducing product returns will have significant impact on the sustainability of e-commerce, both economically and ecologically.

However, simply prohibiting product returns is not an option. Many countries require online retailers to give their customers the right to return products within a certain period of time after purchase. For instance, the minimum required by law in all member states of the European Union is 14 days.[1] In addition, product returns are inherently part of online retailers' business model. Theoretical and empirical studies have demonstrated that a more lenient return policy is associated with a positive impact on customer satisfaction (Cassill, 1998), purchase rates (Wood, 2001), future buying behavior (Petersen and Kumar, 2009) or customers' emotional responses (Suwelack et al., 2011). Many researchers have tried to identify the optimal rate of return (Ketzenberg and Zuidwijk, 2009; Padmanabhan and Png, 1997), in other words the return policy that offers the ideal trade-off between the costs associated with product returns and the beneficial impact they have.

In this study, we take a different approach: There is a gap in the current literature on product returns which focuses on optimizing a static one-size-fits-all (i.e. Bonifield et al. 2010; Petersen and Kumar 2009) return policy rather than a dynamic, customer-specific return strategy (Walsh et al., 2014). We fill this gap by proposing a system of prediction and targeted intervention that is able to identify consumption patterns associated with an extremely high rate of product returns and prevent such transactions from taking place. Such consumption patterns might be associated with fraud or impulse shopping. Fraud (such as ordering an item of clothing, wearing it for a special occasion and then returning it to the retailer) has been identified as an important factor for product returns (Wachter et al., 2012) with about 8% of all product returns estimated to be fraudulent (Speights and Hilinski, 2005). In addition, online shopping has been shown to encourage impulsive or even compulsive consumption patterns, which leads to increased return rates (LaRose, 2001). In the absence of a strategy targeting customers who excessively return products, they will effectively be cross-subsidized by those who are more responsible.

We envision a system that is able to assess the likelihood of a product return in real-time while customers put together their shopping basket. Before they even hit the "order" button, the system will be able to predict which products are most likely to be returned. If the likelihood of a product return is too high, the system can intervene. Such a strategy is known as demarketing (Kotler and Levy, 1971). In developing this system, we recognize that product returns are inherently part of online retailers' business model and therefore concentrate on extreme cases with a very high likelihood of a product return. The large majority of customers would not be affected by this strategy. We will briefly discuss several possibilities for such interventions:

1. Limiting customers' payment options - customers who try to make purchases that are very likely to be returned are requested to pay in advance, via credit card or to use an online payment service. This strategy is motivated by the fact that customers who pay after delivery are about twice as likely to return an item they purchased than those who pay in advance (Asdecker, 2015). The reason for this phenomenon is that when returning products, customers who pay after delivery do not have to make sure that their money is refunded. Instead, they simply never transfer the money in the first place. This suggests that requiring advance payment or direct debit could be an effective strategy against excessive product returns. Pay-after-delivery is the a common means of payment in many countries [2].

2. Artificially increasing delivery time - customers are told that products for which the system predicts a high likelihood of being returned are currently out of stock in the hope that customers will then remove them from their shopping basket. Of course, such a policy can adversely affect customer loyalty, should be used with care and only for very extreme cases.

3. Moral suasion - when a customer tries to make purchase that is very likely to be returned, a pop-up window reminds him or her of the environmental impact associated with product returns. Environmental labeling has been shown to effectively influence consumer behavior (Aguilar and Cai, 2010; Bjørner et al., 2004a,b; D'Souza et al., 2006).

4. Rejecting the transaction - in very extreme cases, online retailers may choose to outright reject the order. Consumer protection laws protect customers' rights after a transaction takes place, but they cannot force companies to accept the transaction in the first place.

[1] Directive 2011/83/EU of the European Parliament and of the Council of 25 October 2011, http://eur-lex.europa.eu/Surpriser/Surpriser.do?uri=OJ:L:2011:304:0064:0088:en:PDF, retrieved 2015-03-16

[2] https://www.about-payments.com/knowledge-base/methods, retrieved 2015-04-08

The purpose of this study is to develop the most important prerequisite for the described strategy, namely a method for accurately predicting product returns. To the best of our knowledge, this study is the first in the academic literature to use machine learning techniques for the identification of consumption patterns associated with a high product return rate. So far, machine learning has only been applied to product returns in very different settings such as forecasting returns of end-of-life goods (Clottey et al., 2012; Toktay et al., 2004). In addition, surveys among online retailers have shown that dynamic predictive methods are not yet widespread in practice and mainly based on very simple indicators such a customer's past return rate (Pur et al., 2013).

Our approach is based on a large dataset containing 1,149,262 purchases obtained from a major German online retailer specializing in fashion. We develop a decision support system including a new algorithm for linear dimensionality reduction, called *Mahalanobis feature extraction*. We demonstrate that Mahalanobis feature extraction is effective for predicting product returns when combined with an adaptive boosting classification algorithm and show that this combination outperforms a wide selection of benchmarks.

The remainder of this study is organized as follows: In the next section we will review related literature. We then introduce our dataset establishing the need for a dimensionality reduction algorithm. After that, we develop Mahalanobis feature extraction. In the subsequent section we introduce our research methodology and research hypothesis. We then present the results of our evaluation. In the final two section, we discuss our results and conclude.

Literature Review

The issue of product returns has been investigated in numerous previous studies. These studies have demonstrated that allowing customers to return products should be understood as being inherently part of a retailer's business model and tried to identify optimal return policies. A number of previous studies have also suggested that the way products are presented can have significant impact on the likelihood of a product return. Finally, the issue of fraudulent behavior has been thoroughly investigated.

Product Returns in E-Commerce

Many authors have demonstrated the importance of product returns to online retailers' business models. It has been shown that a more lenient product return policy has a positive impact on customers' willingness to purchase products (Autry, 2005; Bower and Maxham III, 2012; Petersen and Kumar, 2009; Stock et al., 2006). A more lenient policy towards product returns can also have a positive impact on customer satisfaction (Cassill, 1998; Dissanayake and Singh, 2008), customer loyalty (Mollenkopf et al., 2007) or perceived fairness (Pei et al., 2014). There are two theoretical explanations for this phenomenon. First, a lenient return policy reduces the risk customers take when purchasing products online (Che, 1996; Heiman et al., 2001; Li et al., 2013; Schmidt et al., 1999; Wood, 2001) and can be interpreted through the lens of finance theory as an option (Anderson et al., 2009; Heiman et al., 2002). Second, allowing customers to return products purchased online can be interpreted as a signal for higher product quality (Bonifield et al., 2010).

Theoretical research has demonstrated that there is an optimal rate of return, where the positive impact and the costs associated with product return are in balance (Yan, 2009) and several theoretical models have been proposed to identify the ideal level of leniency. Such an optimal rate of return exists even when taking competition into account (Li et al., 2012). In competitive environments with limited self space, offering generous return policies even for perishable goods is the only Nash equilibrium of the game (Bandyopadhyay and Paul, 2010). When demand is known, return policies increase competition (Padmanabhan and Png, 1997). The optimal return policy based on a straight-forward two-period model for customer preferences for prices and return policies can be demonstrated to be relatively insensitive to parameter assumptions (Ketzenberg and Zuidwijk, 2009). Specific practical insight gained from such models includes the recommendation that marketing campaigns that decrease price sensitiveness of customers will lead to higher profits (Mukhopadhyay and Setoputro, 2004).

Other authors take a more practical approach to the issue of product returns, investigating the impact of specific product return strategies. We can differentiate between monetary strategies such as offering dis-

counts for not returning product, procedure instruments such safety packaging and customer-centric strategies which focus on individual customers such as giving product advice (Walsh et al., 2014). A widely discussed example for a monetary strategy is to charge for the shipping costs associated with product returns (the common practice is to offer free returns). Such a strategy can be shown to lower prices (Ancarani et al., 2009; Chen and Bell, 2009). It also is recognized that information provision can have significant impact on the likelihood of a product return (Zhou et al., 2006). For instance, presenting items of clothing in an emotionally charged way (such as attractive models against a beautiful background) rather than a neutral way will increase the likelihood of a product return (De et al., 2012). On the other hand, positively framing the brands can decrease the likelihood of a product return (Bechwati and Siegal, 2005). Sellers can also use the price and restocking fee as a means of targeting different customer groups (Shulman et al., 2009).

There is empirical evidence that online shopping encourages impulsive (spontaneous and unplanned) or even compulsive (addictive) consumption patterns or can overwhelm customers which leads to an increase in product returns (LaRose, 2001; Rabinovich et al., 2011). Another problem is the issue of fraud: Customers sometimes abuse consumer protection rights (Heiman et al., 2001) or violate social norms (Autry et al., 2007). A typical example would be to purchase an item of clothing, wear it on one or two occasions and then return it to the retailer (Harris, 2010; King and Dennis, 2006; Wachter et al., 2012). This is the most frequent form of fraud (Speights and Hilinski, 2005). In more serious cases, the returned item is not the item that the customer originally ordered or the item was broken by the customer (Harris, 2010; King and Dennis, 2006; Wachter et al., 2012).

Predictive Analytics in Information Systems Research

Predictive analytics has been recognized as an important part of information systems research (Agarwal and Dhar, 2014; Shmueli and Koppius, 2011). Machine learning algorithms have been successfully applied to a wide variety of problem domains such as detecting financial fraud (Abbasi et al., 2012), identifying fake websites (Abbasi et al., 2010), detecting credit card fraud (Bhattacharyya et al., 2011), sales forecasting (Choi et al., 2011), recommender systems (Sahoo et al., 2012) or credit scoring (Zhang et al., 2010).

However, few authors have suggested the use of data analysis to address the issue of product returns. Yu and Wang (2008) propose a hybrid mining approach to divide customers into different segments and offering different return policies to these segments. Some authors propose methods of forecasting returns of end-of-life goods (Clottey et al., 2012; Toktay et al., 2004), which is particularly relevant to electronic products or product that can be environmentally hazardous.

Implications

In summary, there are two important lessons to be drawn from the existing literature which are highly relevant for the development of a decision support system for product returns and its subsequent evaluation.

1. Product returns are inherently part of online retailers' business models. In order to stay competitive with more traditional forms of retail, online retailers, particularly in the fashion industry, must give customers the right to return products if they find they do not like them after purchase.

2. Some customers abuse these rights. Due to impulsive, compulsive or even fraudulent behavior, we often observe transactions that are not profitable.

We argue that there is an important gap in the academic literature in that there currently is no strategy that is both practical and compatible with online retailers' business model. In reality, it may be difficult to implement an optimal return policy based on very abstract mathematical models such as the ones proposed by Bandyopadhyay and Paul (2010); Ketzenberg and Zuidwijk (2009); Li et al. (2012); Mukhopadhyay and Setoputro (2004); Padmanabhan and Png (1997); Shulman et al. (2009, 2011) or Yan (2009). We also believe that a strategy of charging for the shipping costs associated with returns, as frequently proposed in the literature (Ancarani et al., 2009; Chen and Bell, 2009), will punish customers who have legitimate reasons for returning products and is not compatible with online retailers' business model. Information provision may have an impact on product returns, but the statistical significance is only due to the size of the dataset and its impact from a business perspective is actually quite low (De et al., 2012).

A successful and practical strategy should focus on identifying consumption patterns associated with a very high return rate *before the transaction even takes place*. To date there is no such strategy in the academic literature. In fact, there is very little research of customer-based return management strategies (Walsh et al., 2014). Also, there are very few studies that propose the use of machine learning techniques to address the issue of product returns. We believe such a strategy to be particularly promising as machine learning has been successfully applied to many different business intelligence problems.

Data Collection

We cooperated with a major German online retailer that specializes in fashion. For the purposes of this study, we analyzed a dataset consisting of a total of 1,149,262 product purchase from July 2014 to November 2014. The return rate over that period of time was 57.3%.

We separate the indicators contained in our dataset into two categories: Numeric indicators (such as a customer's past return rate or the total number of products in a basket), and nominal indicators (such as the product's brand or the sales channel). Nominal indicators differ from numeric indicators in that they need to be encoded using dummy variables. From a technical point of view, it is beneficial both in terms of computing memory and CPU time to represent our dataset in sparse matrix format, which only stores the non-zero entries while being mathematically equivalent to standard encoding techniques traditionally used in machine learning.

We also differentiate between three levels, namely indicators on the product level (such as a product's brand or color), which may be different for every product in a basket, the basket level (such as the time of the purchase or the sales channel), which are similar for all products in a basket and the customer level (such as a customer's past return rate), which is attributed to a particular customer.

For reasons of data privacy, we abstain from analyzing any personal information, relying on completely pseudonumized data instead. When calculating a customer's past return rate, we included an additional dataset of purchases and returns between July 2013 and June 2014 (the twelve months prior to our actual dataset). When creating these features, we were guided by the assumption that a customer's past return rate will be a good predictor for his or her future return rate and that compulsive shoppers tend to accumulate a large number of similar products.

An overview of all indicators used is provided in Table 1.

In total, our dataset consists of 5868 dummy variables, 10 numeric features and 1,149,262 samples. The overall density of the dataset is roughly 0.33%, demonstrating that the use of sparse matrix format reduces the memory requirement by over 99%.

Mahalanobis Feature Extraction

Most machine learning algorithms do not scale to datasets like the one used in this study or are not applicable to sparse matrices. In this section, we develop a new approach for dimensionality reduction that is particularly useful for large-scale sparse matrices.

Motivation

Suppose we have an $(I \times J)$-matrix $\mathbf{X}$, I being the number of samples, J being the number of features and $I > J$. Suppose further that we have a vector $\mathbf{Y}$ of length I, our goal being to train algorithms to predict $\mathbf{Y}$ using the data contained in $\mathbf{X}$. $\mathbf{Y}$ can be continuous or discrete, therefore the framework is equally applicable to classification and regression problems.

Instead of training our algorithms directly on $\mathbf{X}$, we first transform $\mathbf{X}$, using a $(J \times J^{ext})$-matrix $\mathbf{W}$, J^{ext} being the number of features we want to extract, and $J^{ext} << J$. Let $\mathbf{X^{ext}} := \mathbf{XW}$ be the $(I \times J^{ext})$-matrix containing our extracted features. Our goal is to optimize $\mathbf{W}$, such that we maximize the predictive power $\mathbf{X^{ext}}$ has for $\mathbf{Y}$.

Indicator	Category	Number of features
Product level		
Product *brand*	nominal	672
Product *color*	nominal	2055
Product *size*	nominal	1053
Product *target group*	nominal	6
Activity product is designed for	nominal	51
Product *category*	nominal	65
Product *subcategory*	nominal	783
Sales channel from which customer reached product	nominal	882
Number of times the *exact same product* is in basket	numeric	1
Number of products in basket with *same category* and *target group* in basket	numeric	1
Number of products in basket with same *subcategory* and target group in basket	numeric	1
Number of products in basket with *same brand* and *target group* in basket	numeric	1
Number of products in basket that share *identical characteristics* with the *exception of product size*	numeric	1
Number of products in basket that share *identical characteristics* with the *exception of product color*	numeric	1
Number of times a *similar product*, differing only in size and/or color, is in basket (on the basis of product ID)	numeric	1
Basket level		
Hour of the day at which basket was ordered	nominal	24
Platform from which basket was ordered	nominal	6
Device from which basket was ordered	nominal	32
Operating system from which basket was ordered	nominal	50
Web browser from which basket was ordered	nominal	185
Payment method	nominal	4
Total number of products in basket	numeric	1
Customer level		
Customer's past return rate	numeric	1
Number of products customer previously purchased	numeric	1
Total number of dummy variables		**5868**
Total number of numeric features		**10**
Total number of features		**5878**

Table 1. Overview of the indicators used

The predictive power $\mathbf{X}^{\mathbf{ext}}$ has for $\mathbf{Y}$ is measured using a statistical approach originally proposed by Urbanke et al. (2014). This approach was originally developed for the evaluation of predictive methods. It measures the probability of a set of correlation coefficients under the null hypothesis that the correlation coefficients are a product of a random reshuffling of the data. The test statistic is the squared Mahalanobis distance of the observation from the expected value under the null hypothesis. Under certain assumptions it can be interpreted as being chi-squared distributed.

The approach proposed in this study is to maximize the Mahalanobis distance of the observation from the expected value under the null hypothesis, thus minimizing the probability that the null hypothesis is true. In that, we diverge from the usual machine learning practice of using Bayesian statistics and maximum likelihood, relying on frequentist statistics instead.

Basic Idea

Let x_{ij}, y_i and x_{ij}^{ext} be the element in the i^{th} row and j^{th} column of **X**, **Y** and $\mathbf{X^{ext}}$ respectively. Let z_d be defined as follows:

$$z_d = \sum_{i=1}^{I} x_{id}^{ext} y_i. \tag{1}$$

Let **Z** be a vector of length J^{ext} containing all z_d. We then establish the null hypothesis that $\mathbf{X^{ext}}$ has no predictive power for **Y** and is the product of a random reshuffling of **Y**. Let E(**Z**) be the expected value of **Z** and **V** be the variance-covariance-matrix under the null hypothesis. E(**Z**) can then be calculated as follows (Urbanke et al., 2014):

$$E(z_d) = \frac{\sum_{i=1}^{I} x_{id}^{ext} \sum_{i=1}^{I} y_i}{I}. \tag{2}$$

V can be calculated as follows (Urbanke et al., 2014):

$$cov(z_d, z_{d'}) = \frac{\left(I \sum_{i=1}^{I} x_{id}^{ext} x_{id'}^{ext} - \sum_{i=1}^{I} x_{id}^{ext} \sum_{i=1}^{I} x_{id'}^{ext}\right)\left(I \sum_{i=1}^{I} y^2 - (\sum_{i=1}^{I} y)^2\right)}{I^2(I-1)}. \tag{3}$$

Our goal is to optimize **W** in order to maximize $(\mathbf{Z} - E(\mathbf{Z}))'\mathbf{V^{-1}}(\mathbf{Z} - E(\mathbf{Z}))$. Note that the dimensionality of **V** depends on the number of *extracted* features, rather than the number of *original* features. Since the number of extracted features is, by definition of dimensionality reduction, a lot smaller than the number of original features, the approach can be applied to very large-scale problems.

Because maximizing $(\mathbf{Z} - E(\mathbf{Z}))'\mathbf{V^{-1}}(\mathbf{Z} - E(\mathbf{Z}))$ is equivalent to maximizing the Mahalanobis distance from **Z** to E(**Z**) under the null hypothesis, we call our new algorithm *Mahalanobis feature extraction*.

Numerical Approximation

We use a stochastic gradient descent approach with minibatch updating using a previously defined learning rate ℓ. In every iteration, **W** is updated as follows:

$$w_{cd}^{new} = w_{cd}^{old} + \ell * \frac{\partial (\mathbf{Z}-E(\mathbf{Z}))'\mathbf{V^{-1}}(\mathbf{Z}-E(\mathbf{Z}))}{\partial w_{cd}^{old}}. \tag{4}$$

The partial derivative of $(\mathbf{Z} - E(\mathbf{Z}))'\mathbf{V^{-1}}(\mathbf{Z} - E(\mathbf{Z}))$ with respect to a single element w_{cd} is calculated as follows:

$$\frac{\partial (\mathbf{Z}-E(\mathbf{Z}))'\mathbf{V^{-1}}(\mathbf{Z}-E(\mathbf{Z}))}{\partial w_{cd}} = 2\left(\frac{\partial \mathbf{Z}}{\partial w_{cd}} - \frac{\partial E(\mathbf{Z})}{\partial w_{cd}}\right)'\mathbf{V^{-1}}(\mathbf{Z} - E(\mathbf{Z})) - (\mathbf{Z} - E(\mathbf{Z}))'\mathbf{V^{-1}}\frac{\partial \mathbf{V}}{\partial w_{cd}}\mathbf{V^{-1}}(\mathbf{Z} - E(\mathbf{Z})). \tag{5}$$

Recall that **Z** is defined in (1). The partial derivative of any element in **Z** with respect to a single element w_{cd} is calculated as follows:

$$\frac{\partial z_a}{\partial w_{cd}} = 0, \text{ if } a \neq d. \tag{6}$$

$$\frac{\partial z_a}{\partial w_{cd}} = \sum_{i=1}^{I} x_{ic} y_i, \text{ if } a = d.$$

Recall that $E(\mathbf{Z})$ is defined in (2). The partial derivative of any element in $E(\mathbf{Z})$ with respect to a single element w_{cd} is calculated as follows:

$$\frac{\partial E(z_a)}{\partial w_{cd}} = 0, \text{ if } a \neq d. \tag{7}$$

$$\frac{\partial E(z_a)}{\partial w_{cd}} = \frac{\sum_{i=1}^{I} x_{ic} \sum_{i=1}^{I} y_i}{I}, \text{ if } a = d.$$

Recall that $cov(z_a^b, z_{a'}^{b'})$ is defined in (3). The partial derivative of any element of **V** with respect to a single element $w_{c,d}$ is calculated as follows:

$$\frac{\partial cov(z_a^b, z_{a'}^{b'})}{\partial w_{cd}} = 0, \text{ if } a, a' \neq d. \tag{8}$$

$$\frac{\partial cov(z_a^b, z_{a'}^{b'})}{\partial w_{cd}} = \frac{\left(I \sum_{i=1}^{I} x_{ia} x_{ia'}^{ext} - \sum_{i=1}^{I} x_{ia} \sum_{i=1}^{I} x_{ia'}^{ext}\right)\left(I \sum_{i=1}^{I} y^2 - (\sum_{i=1}^{I} y)^2\right)}{I^2(I-1)}, \text{ if } a = d, a \neq a'.$$

$$\frac{\partial cov(z_a^b, z_{a'}^{b'})}{\partial w_{cd}} = 2\frac{\left(I \sum_{i=1}^{I} x_{ia} x_{ia}^{ext} - \sum_{i=1}^{I} x_{ia} \sum_{i=1}^{I} x_{ia}^{ext}\right)\left(I \sum_{i=1}^{I} y^2 - (\sum_{i=1}^{I} y)^2\right)}{I^2(I-1)}, \text{ if } a = a' = d.$$

For all batches:
reduce1: Calculate $\sum_{i \in batch} y_i$
reduce2: Calculate $\sum_{i \in batch} y_i^2$
reduce3: Calculate $\sum_{i \in batch} x_{ic}$ for all c
reduce4: Calculate $\sum_{i \in batch} x_{ic} y_i$ for all c
In every iteration, for all batches:
Calculate $x_{id}^{ext} = \sum_{j=1}^{J} x_{ij} w_{jd}$ for all $i \in batch, d$
reduce5: Calculate $\sum_{i \in batch} x_{id}^{ext}$ for all d
reduce6: Calculate $\sum_{i \in batch} x_{id}^{ext} y_i$ for all d
reduce7: Calculate $\sum_{i \in batch} x_{id}^{ext} x_{id'}^{ext}$ for all d, d'
reduce8: Calculate $\sum_{i \in batch} x_{ic} x_{id}^{ext}$ for all c, d
Calculate $(\mathbf{Z} - E(\mathbf{Z}))' \mathbf{V^{-1}} (\mathbf{Z} - E(\mathbf{Z}))$ as defined in (1), (2) and (3)
Update **W** as described in (4) and (5)
Repeat until convergence or maximum number of iterations is reached

Figure 1. Expression of the approach in pseudocode

Implementation, Parallelization and Complexity

We separate the necessary tasks into two groups: The first group consists of those tasks that are independent from **W** and have to be calculated only once. The second group consists of those tasks that are dependent on **W** and have to be recalculated in every iteration. Since we use a minibatch updating approach, we separate both **X** and **Y** into equally sized batches. $(\mathbf{Z} - E(\mathbf{Z}))' \mathbf{V^{-1}} (\mathbf{Z} - E(\mathbf{Z}))$ and its derivatives are then not calculated on the entire dataset, but on the individual batches.

It turns out that the sufficient statistics can be comfortably calculated using eight reduce operations, all of which are iterated over individual samples. This is a particularly useful finding, because it implies that the algorithm can be easily parallelized in a distributed-memory system and is therefore applicable to very large problems. The reduce operations are displayed in Figure 1.

Note that we iterate over the columns when calculating sums over **X** and $\mathbf{X^{ext}}$. For reasons of cache-efficiency, this is generally not advisable, but even more problematic in our setting, since **X** is a sparse matrix. For that reason, we divide **X** along its vertical dimension into smaller matrices corresponding to the batches. We then transpose these smaller matrices. In addition, we calculate the transpose of $\mathbf{X^{ext}}$ rather than $\mathbf{X^{ext}}$ itself.

The algorithm is particularly scalable to large problems, because it takes $\mathcal{O}(I)$ and $\mathcal{O}(J)$ time. This means that if we double the number of samples I or double the number of features J , we would expect training time to double as well. In practice, it may even take less than $\mathcal{O}(I)$ time, because we use a minibatch updating approach and would therefore expect the algorithm to take fewer iterations of the entire dataset until convergence. Because of the matrix inversion, the algorithm is expected to take $\mathcal{O}\left((J^{ext})^3\right)$ time, but since J^{ext} is small by the definition of dimensionality reduction, this should not pose much of a problem.

We implement the algorithm in C++ and write an interface to Python using Cython. We parallelize the code using pthreads.

Methodology and Research Hypotheses

We compare our framework with a selection of state-of-the-art techniques commonly used for dimensionality reduction, both supervised and unsupervised. These techniques are chosen on the basis of their applicability to very large-scale sparse matrices and the availability of appropriate implementations. We also test a number of classification algorithms that have been demonstrated to be successful across a large variety of problem domains (Fernández-Delgado et al., 2014).

Our analysis is conducted using the programming language Python and the machine learning library scikit-learn, developed by Pedregosa et al. (2011), as it offers excellent support for sparse matrices.

To structure our analysis, we develop a set of research hypotheses. Our goal is to demonstrate that our pro-

posed solution for predicting product returns, a combination of adaptive boosting and Mahalanobis feature extraction does not only yield sufficiently reliable prediction, but also outperforms standard algorithms in this particular problem setting. This leads us to our first research hypothesis:

> *Research Hypothesis 1a. A combination of adaptive boosting and Mahalanobis feature extraction will yield more accurate predictions for product returns than state-of-the-art predictive algorithms and techniques for feature extraction or feature selection.*

In addition, there are some algorithms that are suitable for large-scale problems without necessitating dimensionality reduction. In particular, we consider a logistic regression and a linear kernel support vector machine. Linear kernel support vector machines scale very well when trained in their primal form (as opposed to the more common dual form) using a stochastic gradient descent algorithm. Hence, our second research hypothesis:

> *Research Hypothesis 1b. A combination of adaptive boosting and Mahalanobis feature extraction will yield more accurate predictions for product returns than a logistic regression and a linear kernel support vector machine that was trained on the entire dataset.*

Finally, we demonstrate that our framework fulfills the purpose it was designed for and provides predictions that are reasonably accurate for companies to work with. In particular, we address the p-value problem as discussed by Lin et al. (2013), who argue that hypothesis tests become less meaningful for large sample sizes and that researchers should be more concerned with what is *interesting* rather than what is *statistically significant*.

We approach this problem in two ways: First, based on our considerations that product returns are part of the business model for online retailers, we reason that an effective strategy for prediction and prevention should focus on extreme cases. Business solutions for this problem should be measured on the basis of their effectiveness at identifying product purchase intentions with a very high likelihood of a product return. Second, we follow the advise by Cohen (1992) and consider our predictive method's effect size.

> *Research Hypothesis 2. A combination of adaptive boosting and Mahalanobis feature extraction will identify consumption patterns associated with a very high rate of product returns at a level of accuracy that is sufficient for a strategy of prediction and intervention to be feasible.*

To conduct an out-of-sample-analysis, we separate our dataset into five equally sized batches, using four of them as our training set and the last one as our testing set. Samples are randomly assigned to the batches. We then corroborate our results by repeating our experiment using a different batch as the testing set every time. We also separate the 10 numeric features, which do not require dimensionality reduction, from the 5868 dummy variables, which do require dimensionality reduction. Using different techniques for feature extraction and feature selection (described below), we reduce the 5868 dummy variables to 10 numeric features, which we then combine with the original 10 numeric features to form a dataset consisting of 20 numeric features. The reduction to 10 features in particular is a compromise between reducing the computing resources required and keeping information loss to a minimum. All feature extractors and algorithms are trained on the training set. The evaluation is conducted using the testing set. The testing set is not involved in the training either the algorithms or the feature extractors in any way.

Results

To evaluate the first hypothesis, we compare Mahalanobis feature extraction to a total of six techniques for feature extraction or feature selection, namely principal component analysis, linear discriminant analysis, a randomized truncated singular value decomposition, a feature selector that selects features based on their univariate chi-squared statistic, a random projection and non-negative matrix factorization. As an seventh "feature extraction technique" we simply ignore the nominal indicators.

We use each of these techniques on the 5868 dummy variables and reduce them to 10 numeric features. We then combine these 10 extracted features with the 10 original numeric features to form a dataset containing 20 features. The only exception is linear discriminant analysis: Since the number of extracted features cannot be larger than one in our case, we only extract a single feature. Even though linear discriminant analysis is

a very popular algorithm for supervised dimensionality reduction, it actually does not scale very well: In our case, we would have to calculate and invert two (5868×5868)-variance-covariance matrices as opposed to the (10×10)-variance-covariance matrices necessary for Mahalanobis feature extraction. To keep the problem manageable, we first extract 100 features using principal component analysis and then train the linear discriminant analysis on top of these features. Such an approach is very common and implemented as the standard procedure in some machine learning libraries such as dlib C++ (King, 2009).

We also compare adaptive boosting to a total seven other classification algorithms, namely CART, extremely randomized trees, which are highly randomized bootstrapped decision trees, gradient boosting, linear discriminant analysis, since it can be used both as classifier and a feature extractor, logistic regression, random forest and a linear kernel support vector machine, since non-linear kernels do not scale to this problem size.

We combine each of the above-mentioned feature extraction techniques to each of the algorithms. Taking into account that we also try to ignore all nominal features, this amounts to a total of 64 (= 8 classifiers x 8 feature extractors) different combinations.

Since the logistic regression and linear support vector machines scale to large problem sizes (if trained using appropriate algorithms), we also train these algorithms on the entire training set, meaning both all numeric features and dummy variables.

We then calculate the correlation coefficient (Pearson's r) between the out-of-sample probabilistic predictions and the actual class variables in the testing set. Pearson's r assumes a value of 1 (best value possible) for perfect correlation between the probabilistic predictions and the actual class variables and a value of 0 for no correlation at all. It is an important measure for two reasons: First, a high correlation between the probabilistic predictions and the actual class variables implies a good ability to identify extreme cases with a high probability of a product return, which is the ultimate goal of this study. Second, Pearson's r is explicitly proposed to measure the effect size of an indicator (Cohen, 1992). Results are reported in Table 2. The best classifier given a particular feature extractor is underlined. The best feature extractor given a particular classifier is in *italics*.

	NEF	**Ada-Boost**	**CART**	**ERT**	**GB**	**LDA**	**LR**	**RF**	**SVM**
MahaFeatExt	10	*0.409*	*0.380*	*0.397*	*0.382*	*0.358*	*0.364*	*0.402*	*0.300*
PCA	10	0.389	0.354	0.373	0.351	0.298	0.308	0.379	0.262
RanTSVD	10	0.384	0.348	0.368	0.343	0.301	0.311	0.373	0.266
LDA	1	0.384	0.375	0.384	0.370	0.329	0.337	0.387	0.290
ChiSelect	10	0.379	0.366	0.372	0.349	0.310	0.319	0.372	0.272
RanProj	10	0.361	0.329	0.349	0.331	0.281	0.292	0.354	0.261
NMF	10	0.387	0.354	0.369	0.347	0.291	0.301	0.375	0.251
NoNominal	10	0.349	0.342	0.344	0.331	0.271	0.282	0.348	0.250
OrigData	5868	-	-	-	-	-	0.371	-	0.301

Number of extracted features = **NEF**, Mahalanobis feature extraction = **MahaFeatExt**, principal component analysis = **PCA**, linear discriminant analysis = **LDA**, randomized truncated singular value decomposition = **RanTSVD**, feature selector based on univariate chi-squared statistic = **ChiSelect**, random projection = **RanProj**, non-negative matrix factorization = **NMF**, no nominal indicators = **NoNominal**, classifiers trained on original dataset (where possible) = **OrigData**
Adaptive boosting = **AdaBoost**, classification and regression trees = **CART**, extremely randomized trees = **ERT**, gradient boosting = **GB**, linear discriminant analysis = **LDA**, logistic regression = **LR**, random forest = **RF**, linear kernel support vector machine = **SVM**
Best classifier given feature extractor is underlined. Best feature extractor given classifier is in *italics*.

Table 2. Pearson's r between probabilistic out-of-sample predictions and actual class labels

We find that a combination of adaptive boosting and Mahalanobis feature extraction achieves the highest predictive accuracy. We also find that Mahalanobis feature extraction offers the highest predictive accuracy given any classifier when compared to any other method of feature extraction. In fact, a comparison between the logistic regression and the linear support vector machine trained on the original dataset and the same classifiers trained on the features extracted using Mahalanobis feature extraction shows that there is

not much difference in predictive accuracy. This implies that even though we have reduced 5868 dummy variables to 10 numeric features (a reduction in the number of features by over 99.8%), the information loss associated with that reduction is not large and considerably smaller than for any other feature extraction algorithm.

We test whether this outperformance is statistically significant. Results are shown in Table 3. We find that the results are statistically significant at the 1%-level for all predictive methods.

	AdaBoost	**CART**	**ERT**	**GB**	**LDA**	**LR**	**RF**	**SVM**
MahaFeatExt	-	<0.001	0.000	0.000	0.000	0.000	<0.01	0.000
PCA	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
RanTSVD	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LDA	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
ChiSelect	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
RanProj	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
NMF	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
NoNominal	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
OrigData	-	-	-	-	-	0.000	-	0.000

Mahalanobis feature extraction = **MahaFeatExt**, principal component analysis = **PCA**, linear discriminant analysis = **LDA**, randomized truncated singular value decomposition = **RanTSVD**, feature selector based on univariate chi-squared statistic = **ChiSelect**, random projection = **RanProj**, non-negative matrix factorization = **NMF**, no nominal indicators = **NoNominal**, classifiers trained on original dataset (where possible) = **OrigData**

Adaptive boosting = **AdaBoost**, classification and regression trees = **CART**, extremely randomized trees = **ERT**, gradient boosting = **GB**, linear discriminant analysis = **LDA**, logistic regression = **LR**, random forest = **RF**, linear kernel support vector machine = **SVM**

Table 3. Statistical significance (p-value) of outperformance of MahaFeatExt + AdaBoost over predictive method

In addition to analyzing Pearson's r, we calculate precision and recall: In our literature review, we have argued that product returns are inherently part of online retailers' business model and that a system for prediction and prevention should focus on extreme cases, namely product purchase intentions with a very high probability of a product return. Systems for the prediction and prevention of product returns should be measured by their ability to identify such extreme cases. For our evaluation this implies that *precision* is considerably more important than *recall*. In fact, considering that product returns are part of online retailers' business model, a very high recall is actually undesirable. As long as recall is sufficiently high for the system to have a measurable impact, we should focus on maximizing the precision with which the system identifies product returns.

On the basis of the training set, we define the 10%- and the 5%-threshold. These thresholds are chosen such that the system would be expected to intervene for the 10% or 5% of all purchase intentions that are expected to be most likely to result in a product return. We then calculate precision and recall for all approaches based on the 10%- and 5%-threshold. We have excluded the support vector machine from this table, as a probabilistic interpretation of the results in not possible. Results are reported in Table 4.

We find that the combination of Mahalanobis feature extraction and adaptive boosting achieves the highest precision at both the 10%- and the 5%-threshold, even though it does not achieve the highest recall.

In addition, we plot the receiver operating characteristic (ROC) curve. Since the extreme cases we are interested in are located in the lower left-hand part of the ROC curve, we concentrate on that section (see Figure 2). For comparison, we show all feature extractors used in this study in conjunction with the classifier that performed best given the particular feature extractor (in other words, the underlined combinations in Table 2). Our results demonstrate that the classifier based on Mahalanobis feature extraction offers a better trade-off between the true positive rate and the false positive rate at all relevant levels.

In addition to a statistical analysis of the proposed method, we also analyze its usefulness as defined in Hypothesis 2. Lin et al. (2013) argue that hypothesis tests become less meaningful for large sample sizes and that researchers should be more concerned with what is *interesting* rather than what is *statistically*

significant. Especially since the focus of this study is to provide a solution for a specific and important business problem (product returns in e-commerce), we provide evidence that the methodology proposed in this study can be used as the basis for a system for prediction and prevention of product returns.

		AdaBoost	CART	ERT	GB	LDA	LR	RF
Threshold at 10%:								
MahaFeatExt	**P**	*0.848*	0.823	*0.843*	0.818	*0.820*	*0.820*	*0.844*
	R	0.141	*0.142*	0.144	0.143	*0.143*	*0.142*	0.144
PCA	**P**	0.843	0.807	0.831	0.816	0.789	0.791	0.834
	R	0.140	0.140	0.143	0.142	0.138	0.139	0.143
RanTSVD	**P**	0.839	0.811	0.830	0.815	0.789	0.791	0.816
	R	0.138	0.140	0.143	0.142	0.137	0.138	0.142
LDA	**P**	0.835	*0.825*	0.837	*0.829*	0.805	0.804	0.838
	R	0.143	*0.142*	*0.146*	0.144	0.141	0.140	*0.145*
ChiSelect	**P**	0.835	0.817	0.824	0.811	0.788	0.788	0.826
	R	*0.144*	*0.142*	0.143	0.141	0.138	0.138	0.143
RanProj	**P**	0.827	0.797	0.815	0.803	0.779	0.779	0.819
	R	0.135	0.137	0.140	0.140	0.137	0.136	0.141
NMF	**P**	0.842	0.810	0.829	0.817	0.784	0.787	0.832
	R	0.140	0.139	0.144	*0.147*	0.137	0.138	0.142
NoNominal	**P**	0.819	0.810	0.814	0.801	0.773	0.775	0.817
	R	0.142	0.140	0.142	0.140	0.136	0.135	0.142
OrigData	**P**	-	-	-	-	-	0.822	-
	R	-	-	-	-	-	0.142	-
Threshold at 5%:								
MahaFeatExt	**P**	*0.867*	0.835	*0.860*	*0.852*	*0.832*	*0.825*	*0.864*
	R	0.069	0.071	0.072	*0.073*	*0.072*	*0.072*	0.072
PCA	**P**	0.865	0.826	0.851	0.835	0.799	0.797	0.855
	R	0.069	*0.072*	0.072	0.072	0.069	0.070	0.072
RanTSVD	**P**	0.863	0.830	0.852	0.835	0.801	0.801	0.854
	R	0.068	0.071	*0.073*	*0.073*	0.070	0.070	0.072
LDA	**P**	0.855	*0.843*	0.856	0.848	0.813	0.811	0.856
	R	0.072	*0.072*	*0.073*	*0.073*	0.070	0.071	*0.073*
ChiSelect	**P**	0.861	0.836	0.848	0.832	0.801	0.794	0.851
	R	*0.073*	*0.072*	0.072	0.072	0.070	0.069	0.072
RanProj	**P**	0.853	0.811	0.839	0.823	0.790	0.786	0.845
	R	0.066	0.070	0.072	0.071	0.069	0.069	0.072
NMF	**P**	0.861	0.830	0.850	0.834	0.799	0.795	0.853
	R	0.069	0.071	0.072	0.072	0.070	0.069	0.072
NoNominal	**P**	0.846	0.831	0.836	0.824	0.780	0.780	0.840
	R	0.072	*0.072*	0.072	0.070	0.067	0.068	0.072
OrigData	**P**	-	-	-	-	-	0.830	-
	R	-	-	-	-	-	0.072	-

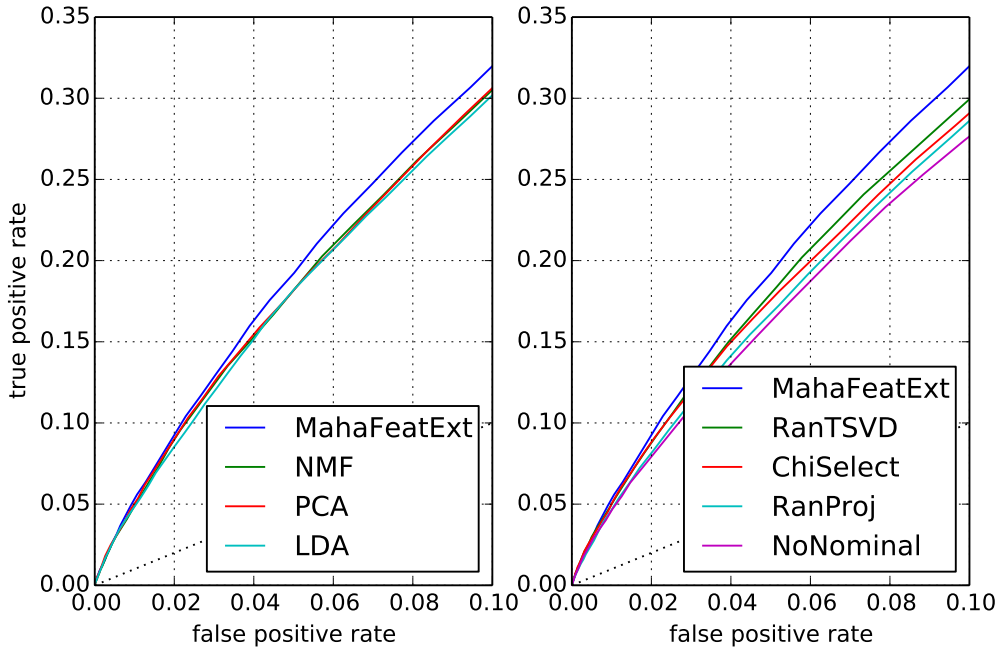
Mahalanobis feature extraction = **MahaFeatExt**, principal component analysis = **PCA**, linear discriminant analysis = **LDA**, randomized truncated singular value decomposition = **RanTSVD**, feature selector based on univariate chi-squared statistic = **ChiSelect**, random projection = **RanProj**, non-negative matrix factorization = **NMF**, no nominal indicators = **NoNominal**, classifiers trained on original dataset (where possible) = **OrigData**

Adaptive boosting = **AdaBoost**, classification and regression trees = **CART**, extremely randomized trees = **ERT**, gradient boosting = **GB**, linear discriminant analysis = **LDA**, logistic regression = **LR**, random forest = **RF**, linear kernel support vector machine = **SVM**

Precision = **P**, Recall = **R**

Table 4. Precision and recall for different methods and thresholds

We follow the advise by Cohen (1992) who propose to calculate Pearson's r to measure the effect size. A value



Mahalanobis feature extraction = **MahaFeatExt**, principal component analysis = **PCA**, linear discriminant analysis = **LDA**, randomized truncated singular value decomposition = **RanTSVD**, feature selector based on univariate chi-squared statistic = **ChiSelect**, random projection = **RanProj**, non-negative matrix factorization = **NMF**, no nominal indicators = **NoNominal**

Figure 2. Out-of-sample ROC-curves for different feature extractors, in combination with best classifier

of 0.3 implies a medium effect size whereas a value of 0.5 implies a large effect size. Given that the predictive model proposed in this paper has a value of 0.409 (see Table 2), we conclude that our model achieves a medium to large effect size.

We also investigate the business value of the proposed system using precision and recall as calculated in in Table 4. These figures can be interpreted as follows: Suppose we were to construct a model of prediction and intervention based on a combination of adaptive boosting and Mahalanobis feature extraction. If we choose the 10%-threshold, the system would intervene for about 10% of all purchase intentions. 84.8% of all interventions would be justified. Assuming a reasonably effective intervention strategy, we would then be able to reduce the number of product returns by up to 14.1%. Considering that the company which provided the datasets sells several millions of products every year and every product return incurs costs of several euros, implementing such a system could easily reduce annual costs by a six-digit figure. We have no reason to believe that these findings could not be equally applied to other online retailers in fashion. Overall, this suggests that our findings have the potential for significant business impact.

To corroborate our results, we repeat our experiment five times, using a different one of the five batches described above as our testing set. We find that our results are very robust: The combination of Mahalanobis feature extraction and adaptive boosting remains the best performing combination every time we repeat the experiment and outperforms all other combinations in a statistically significant manner. All figures calculated change only slightly, thus demonstrating that our system has finding has business value is not idiosyncratic to a specific choice of a training and testing set, but can be relied upon as a basis for an intervention strategy.

We use the framework suggested by Urbanke et al. (2014) to test whether our predictive model generates additional information or could have been constructed by a combination of extant methods. In other words, we test whether its out-of-sample predictions remain statistically significant *when corrected for* the out-of-

sample predictions generated by all other methods. We find that they do so at very high significance levels. For comparison, we do the same for all other predictive methods we considered in this study. Results are reported in Table 5.

We find that despite the large sample size, the majority of predictive methods do not generate out-of-sample predictions that remain statistically significant when corrected for other methods. This implies that, unlike most other predictive methods we investigated, our predictive method generates additional information that extant methods cannot provide.

	AdaBoost	**CART**	**ERT**	**GB**	**LDA**	**LR**	**RF**	**SVM**
MahaFeatExt	0.0000	0.0793	0.6288	0.5045	0.000	0.0000	0.008	0.2581
PCA	0.0000	0.0227	0.8243	0.8069	0.4696	0.3593	0.9890	0.9676
RanTSVD	0.0000	0.0925	0.3716	0.0142	0.9199	0.8450	0.9154	0.4362
LDA	0.6830	0.8910	0.0000	0.0757	0.0158	0.0020	0.400	0.0767
ChiSelect	0.0000	0.9354	0.7415	0.0653	0.0069	0.0227	0.1368	0.2486
RanProj	0.0000	0.3916	0.2112	0.0425	0.2003	0.2899	0.7785	0.7643
NMF	0.0000	0.0176	0.7117	0.9756	0.3451	0.1831	0.8520	0.0327
NoNominal	0.0000	0.3108	0.0642	0.3470	0.0233	0.0553	0.0283	0.0710
OrigData	-	-	-	-	-	0.0000	-	0.0789

Mahalanobis feature extraction = **MahaFeatExt**, principal component analysis = **PCA**, linear discriminant analysis = **LDA**, randomized truncated singular value decomposition = **RanTSVD**, feature selector based on univariate chi-squared statistic = **ChiSelect**, random projection = **RanProj**, non-negative matrix factorization = **NMF**, no nominal indicators = **NoNominal**, classifiers trained on original dataset (where possible) = **OrigData**

Adaptive boosting = **AdaBoost**, classification and regression trees = **CART**, extremely randomized trees = **ERT**, gradient boosting = **GB**, linear discriminant analysis = **LDA**, logistic regression = **LR**, random forest = **RF**, linear kernel support vector machine = **SVM**

Table 5. Statistical significance (p-value) of predictive method when corrected for all other predictive methods

Discussion, Limitations and Implications for Further Research

The purpose of this study was to develop an innovative decision support system for the prediction of product returns in e-commerce. We have shown that our decision support system outperforms state-of-the-art methods in this particular problem domain and can generate predictions that are accurate enough for a system of prediction and targeted intervention to be feasible.

Using a large scale dataset related to product returns in e-commerce, we compared Mahalanobis feature extraction to a selection of the state-of-the art algorithms and feature selectors and demonstrated that it is able to outperform all of these methods given any of the classification algorithms we considered. This outperformance was statistically highly significant. We were also able to show that a combination of adaptive boosting and Mahalanobis feature extraction was able to generate new predictive information that other predictive methods could not achieve.

We also evaluated the business value of our decision support system. We demonstrated that the method is able to accurately identify cases in which the likelihood of a product being returned is extremely high, suggesting that it enables a strategy of targeted intervention. In that we were able to fill an important gap in the existing academic literature on product returns: Whereas extant literature focused on optimizing the return policy for all customers, we proposed a strategy that focuses on individual customers and is therefore more compatible with online retailers' business model. Surveys among online retailers have demonstrated that such methods are not yet widespread: Less than 50% of online retailers analyze the likelihood of products being returned at all, 60% of those that do, do so only occasionally and they generally use only very simple measures such as a customers' past return rate to so (Pur et al., 2013).

We recognize that whether and to what extent a strategy of prediction and targeted intervention should be implemented is essentially a business decision. We have therefore provided a choice of different thresholds and calculated the accuracy with which our model can identify the likelihood of product returns given each

of these thresholds. We have also discussed a range of possible intervention strategies ranging from soft interventions such as moral suasion, hard interventions such as not allowing a transaction to take place or a compromise such as limiting payment options, which makes product returns more tedious and forces customers to think twice without actually preventing the transaction.

Considering that many online retailers report that profitability would increase by more than 20%, if they could achieve a 10% reduction in the rate of product returns (Pur et al., 2013), even a conservative intervention strategy based on a predictive model such as the one proposed in this study could constitute a meaningful contribution to many companies' overall profit margins while at the same time reducing their carbon footprint. In addition, the large majority of responsible customers, who would not be targeted by the approach we propose, would benefit in that they would not have cross-subsidize irresponsible consumption patterns.

However, the success of "prediction and targeted intervention" as proposed in this study depends on the effectiveness of intervention mechanisms and customer acceptance. We believe that the use of field experiments to test different strategies in conjunction with a predictive model, such as the one proposed in this study, could provide further insight into the feasibility of different intervention strategies as well as customer acceptance. This would allow us to embed the predictive method presented in this paper into a larger customer lifetime value concept.

We have also introduced a new algorithm for linear dimensionality reduction, *Mahalanobis feature extraction,* and demonstrated that it outperforms state-of-the-art dimensionality reduction algorithms at statistically highly significant levels in the context of this particular problem domain. In principle, the algorithm is applicable to any problem domain which requires the reduction of large-scale sparse matrices. Because most customer databases contain categorical data, the algorithm is applicable to many business intelligence problems. We therefore believe that the usefulness of *Mahalanobis feature extraction* may transcend this particular problem domain. Future research will have to further examine the extent of its usefulness for other business intelligence problems.

Conclusion

In this paper, we developed a framework for predicting product returns in e-commerce by introducing *Mahalanobis feature extraction* as a new method of dimensionality reduction. We used an extensive dataset obtained from a major German online retailer specializing in fashion to evaluate our model. We demonstrated that *Mahalanobis feature extraction* in combination with an adaptive boosting algorithm outperforms a wide selection of benchmarks and shown that our model can effectively identify consumption patterns associated with a high rate of product returns.

References

Abbasi, A., Albrecht, C., Vance, A., and Hansen, J. 2012. "MetaFraud: A Meta-Learning Framework for Detecting Financial Fraud." *MIS Quarterly* (36:4), pp. 1293–A12.

Abbasi, A., Zhang, Z., Zimbra, D., Chen, H., and Nunamaker, J., Jay F. 2010. "Detecting Fake Websites: The Contribution of Statistical Learning Theory," *MIS Quarterly* (34:3), pp. 435–461.

Agarwal, R., and Dhar, V. 2014. "Editorial—Big Data, Data Science, and Analytics: The Opportunity and Challenge for IS Research," *Information Systems Research* (25:3), pp. 443–448.

Aguilar, F. X., and Cai, Z. 2010. "Conjoint effect of environmental labeling, disclosure of forest of origin and price on consumer preferences for wood products in the US and UK," *Ecological Economics* (70:2), pp. 308–316.

Ancarani, F., Gerstner, E., Posselt, T., and Radic, D. 2009. "Could higher fees lead to lower prices?" *Journal of Product & Brand Management* (18:4), pp. 297–305.

Anderson, E. T., Hansen, K., and Simester, D. 2009. "The option value of returns: Theory and empirical evidence," *Marketing Science* (28:3), pp. 405–423.

Asdecker, B. 2015. "Statistiken Retouren Deutschland - Definition," http://www.retourenforschung.de/, accessed: 2015-03-16.

Autry, C. W. 2005. "Formalization of reverse logistics programs: A strategy for managing liberalized returns," *Industrial Marketing Management* (34:7), pp. 749–757.

Autry, C. W., Hill, D. J., and O'Brien, M. 2007. "Attitude toward the customer: a study of product returns episodes," *Journal of Managerial Issues* pp. 315–339.

Bandyopadhyay, S., and Paul, A. A. 2010. "Equilibrium returns policies in the presence of supplier competition," *Marketing Science* (29:5), pp. 846–857.

Bechwati, N. N., and Siegal, W. S. 2005. "The Impact of the Prechoice Process on Product Returns," *Journal of Marketing Research* (42:3), pp. 358–367.

Bhattacharyya, S., Jha, S., Tharakunnel, K., and Westland, J. C. 2011. "Data mining for credit card fraud: A comparative study," *Decision Support Systems* (50:3), pp. 602–613.

Bjørner, T. B., Hansen, L. G., and Russell, C. S. 2004a. "Environmental labeling and consumers' choice—an empirical analysis of the effect of the Nordic Swan," *Journal of Environmental Economics and Management* (47:3), pp. 411–434.

Bjørner, T. B., Hansen, L. G. a., and Russell, C. S. 2004b. "Environmental labeling and consumers' choice—an empirical analysis of the effect of the Nordic Swan," *Journal of Environmental Economics and Management* (47:3), pp. 411–434.

Bonifield, C., Cole, C., and Schultz, R. L. 2010. "Product returns on the Internet: A case of mixed signals?" *Journal of Business Research* (63:9), pp. 1058–1065.

Bower, A. B., and Maxham III, J. G. 2012. "Return shipping policies of online retailers: Normative assumptions and the long-term consequences of fee and free returns," *Journal of Marketing* (76:5), pp. 110–124.

Browne, M., Rizet, C., Leonardi, J., and Allen, J. 2008. "Analysing energy use in supply chains: the case of fruits and vegetables and furniture," *Proceedings of the Logistics Research Network Conference* pp. 1–6.

Cassill, N. L. 1998. "Do customer returns enhance product and shopping experience satisfaction?" *The International Review of Retail, Distribution and Consumer Research* (8:1), pp. 1–13.

Che, Y.-K. 1996. "Customer return policies for experience goods," *The Journal of Industrial Economics* pp. 17–24.

Chen, J., and Bell, P. C. 2009. "The impact of customer returns on pricing and order decisions," *European Journal of Operational Research* (195:1), pp. 280–295.

Choi, T.-M., Yu, Y., and Au, K.-F. 2011. "A hybrid SARIMA wavelet transform method for sales forecasting," *Decision Support Systems* (51:1), pp. 130–140.

Clottey, T., Benton, W. C., and Srivastava, R. 2012. "Forecasting product returns for remanufacturing operations," *Decision Sciences* (43:4), pp. 589–614.

Cohen, J. 1992. "A power primer." *Psychological bulletin* (112:1), p. 155.

De, P., Hu, Y., and Rahman, M. S. 2012. "An Empirical Investigation of the Effects of Product-Oriented Web Technologies on Product Returns," *Working Paper* pp. 1–34.

Dissanayake, D., and Singh, M. 2008. "Managing returns in e-business," *Journal of Internet Commerce* (6:2), pp. 35–49.

D'Souza, C., Taghian, M., and Lamb, P. 2006. "An empirical study on the influence of environmental labels on consumers," *Corporate Communications: An International Journal* (11:2), pp. 162–173.

Fernández-Delgado, M., Cernadas, E., Barro, S., and Amorim, D. 2014. "Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?" *Journal of Machine Learning Research* (15), pp. 3133–3181.

Fuchs, C. 2008. "The implications of new information and communication technologies for sustainability," *Environment, Development and Sustainability* (10:3), pp. 291–309.

Halldórsson, Á., Kovács, G., Edwards, J. B., McKinnon, A. C., and Cullinane, S. L. 2010. "Comparative analysis of the carbon footprints of conventional and online retailing: A "last mile" perspective," *International Journal of Physical Distribution & Logistics Management* (40:1/2), pp. 103–123.

Harris, L. C. 2010. "Fraudulent consumer returns: exploiting retailers' return policies," *European Journal of Marketing* (44:6), pp. 730–747.

Heiman, A., McWilliams, B., Zhao, J., and Zilberman, D. 2002. "Valuation and management of money-back guarantee options," *Journal of retailing* (78:3), pp. 193–205.

Heiman, A., McWilliams, B., and Zilberman, D. 2001. "Demonstrations and money-back guarantees: market mechanisms to reduce uncertainty," *Journal of Business Research* (54:1), pp. 71–84.

Ketzenberg, M. E., and Zuidwijk, R. A. 2009. "Optimal pricing, ordering, and return policies for consumer goods," *Production and Operations Management* (18:3), pp. 344–360.

King, D. E. 2009. "Dlib-ml: A machine learning toolkit," *The Journal of Machine Learning Research* (10), pp. 1755–1758.

King, T., and Dennis, C. 2006. "Unethical consumers: Deshopping behaviour using the qualitative analysis of theory of planned behaviour and accompanied (de) shopping," *Qualitative Market Research: An International Journal* (9:3), pp. 282–296.

Kotler, P., and Levy, S. J. 1971. "Demarketing, yes, demarketing," *Harvard Business Review* (49:6), p. 74.

LaRose, R. 2001. "On the Negative Effects of E-Commerce: A Sociocognitive Exploration of Unregulated On-line Buying," *Journal of Computer-Mediated Communication* (6:3), pp. 1–37.

Li, Y., Wei, C., and Cai, X. 2012. "Optimal pricing and order policies with B2B product returns for fashion products," *International Journal of Production Economics* (135:2), pp. 637–646.

Li, Y., Xu, L., and Li, D. 2013. "Examining relationships between the return policy, product quality, and pricing strategy in online direct selling," *International Journal of Production Economics* (144:2), pp. 451–460.

Lin, M., Lucas Jr, H. C., and Shmueli, G. 2013. "Research commentary-too big to fail: large samples and the p-value problem," *Information Systems Research* (24:4), pp. 906–917.

Mollenkopf, D. A., Rabinovich, E., Laseter, T. M., and Boyer, K. K. 2007. "Managing internet product returns: a focus on effective service operations," *Decision Sciences* (38:2), pp. 215–250.

Mukhopadhyay, S. K., and Setoputro, R. 2004. "Reverse logistics in e-business: optimal price and return policy," *International Journal of Physical Distribution & Logistics Management* (34:1), pp. 70–89.

Padmanabhan, V., and Png, I. P. 1997. "Manufacturer's return policies and retail competition," *Marketing Science* (16:1), pp. 81–94.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., and Dubourg, V. 2011. "Scikit-learn: Machine learning in Python," *The Journal of Machine Learning Research* (12), pp. 2825–2830.

Pei, Z., Paswan, A., and Yan, R. 2014. "E-tailer's return policy, consumer's perception of return policy fairness and purchase intention," *Journal of Retailing and Consumer Services* (21:3), pp. 249–257.

Petersen, J. A., and Kumar, V. 2009. "Are product returns a necessary evil? Antecedents and consequences," *Journal of Marketing* (73:3), pp. 35–51.

Pur, S., Stahl, E., Wittmann, M., Wittmann, G., and Weinfurter, S. 2013. *Retourenmanagement im Online Handel - Das Beste daraus machen*, Regensburg: ibi research an der Universität Regensburg GmbH.

Rabinovich, E., Sinha, R., and Laseter, T. 2011. "Unlimited shelf space in Internet supply chains: Treasure trove or wasteland?" *Journal of Operations Management* (29:4), pp. 305–317.

Rigby, D. 2014. "Online Shopping Isn't as Profitable as You Think," https://hbr.org/2014/08/online-shopping-isnt-as-profitable-as-you-think/, accessed: 2015-04-20.

Sahoo, N., Singh, P. V., and Mukhopadhyay, T. 2012. "A Hidden Markov Model for Collaborative Filtering." *MIS Quarterly* (36:4), pp. 1329–1356.

Schmidt, R. A., Sturrock, F., Ward, P., and Lea-Greenwood, G. 1999. "Deshopping-the art of illicit consumption," *International Journal of Retail & Distribution Management* (27:8), pp. 290–301.

Shmueli, G., and Koppius, O. R. 2011. "Predictive Analytics in Information Systems Research." *MIS Quarterly* (35:3), pp. 553–572.

Shulman, J. D., Coughlan, A. T., and Savaskan, R. C. 2009. "Optimal restocking fees and information provision in an integrated demand-supply model of product returns," *Manufacturing & Service Operations Management* (11:4), pp. 577–594.

Shulman, J. D., Coughlan, A. T., and Savaskan, R. C. 2011. "Managing Consumer Returns in a Competitive Environment," *Management Science* (57:2), pp. 347–362.

Song, L., Cherrett, T., McLeod, F., and Guan, W. 2009. "Addressing the Last Mile Problem," *Transportation Research Record: Journal of the Transportation Research Board* (2097), pp. 9–18.

Speights, D., and Hilinski, M. 2005. "Return fraud and abuse: How to protect profits," *Retailing Issues Letter* (17:1), pp. 1–6.

Stock, J., Speh, T., and Shear, H. 2006. "Managing product returns for competitive advantage," *MIT Sloan Management Review* (48:1), pp. 57–62.

Suwelack, T., Hogreve, J., and Hoyer, W. D. 2011. "Understanding money-back guarantees: cognitive, affective, and behavioral outcomes," *Journal of retailing* (87:4), pp. 462–478.

Toktay, L. B., van der Laan, E. A., and de Brito, M. P. 2004. *Managing product returns: the role of forecasting*, Springer.

Urbanke, P., Kranz, J., and Kolbe, L. 2014. "A Unified Statistical Framework for Evaluating Predictive Methods," *Proceedings of the 35th International Conference on Information Systems (ICIS), Auckland, December 14-17, 2014* pp. 1–12.

Wachter, K., Vitell, S. J., Shelton, R. K., and Park, K. 2012. "Exploring consumer orientation toward returns: unethical dimensions," *Business Ethics: A European Review* (21:1), pp. 115–128.

Walsh, G., Möhring, M., Koot, C., and Schaarschmidt, M. 2014. "Preventive product returns management systems - A review and model," *Proceedings of the European Conference on Information Systems (ECIS) 2014, Tel Aviv, Israel, June 9-11* pp. 1–12.

Weber, C., Hendrickson, C., Jaramillo, P., Matthews, S., Nagengast, A., and Nealer, R. 2008. "Life cycle comparison of traditional retail and e-commerce logistics for electronic products: A case study of buy.com," *Green Design Institute, Carnegie Mellon University* .

Williams, E., and Tagami, T. 2002. "Energy Use in Sales and Distribution via E-Commerce and Conventional Retail: A Case Study of the Japanese Book Sector," *Journal of Industrial Ecology* (6:2), pp. 99–114.

Wood, S. L. 2001. "Remote purchase environments: the influence of return policy leniency on two-stage decision processes," *Journal of Marketing Research* (38:2), pp. 157–169.

Yan, R. 2009. "Product categories, returns policy and pricing strategy for e-marketers," *Journal of Product & Brand Management* (18:6), pp. 452–460.

Yu, C.-C., and Wang, C.-S. 2008. "A hybrid mining approach for optimizing returns policies in e-retailing," *Expert Systems with Applications* (35:4), pp. 1575–1582.

Zhang, D., Zhou, X., Leung, S. C., and Zheng, J. 2010. "Vertical bagging decision trees model for credit scoring," *Expert Systems with Applications* (37:12), pp. 7838–7843.

Zhou, H., Reid, R. D., and Benton Jr., W. 2006. "The drivers of product return in the information age," *International Journal of Internet and Enterprise Management* (4:2), pp. 100–117.