REPUBLIC OF TURKEY

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

PREDICTING LAPSING CUSTOMERS WITH LOGISTIC REGRESSION APPROACH IN RETAIL

ÇAĞDAŞ KANAR

MSc. THESIS

DEPARTMENT of STATISTICS

PROGRAM of STATISTICS

ADVISER

ASST. PROF. İBRAHİM DEMİR

İSTANBUL, 2014

REPUBLIC OF TURKEY

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

PREDICTING LAPSING CUSTOMERS WITH LOGISTIC REGRESSION APPROACH IN RETAIL

A thesis submitted by Çağdaş KANAR in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE** is approved by the committee on 30.05.2014 in Department of Statistics, Graduate of Statistics Program.

Thesis Adviser

Asst. Prof. Dr. İbrahim DEMİR

Yıldız Technical University

Approved By the Examining Committee

Asst. Prof. Dr. İbrahim DEMİR

Yıldız Technical University _____

Prof. Dr. Ali Hakan BÜYÜKLÜ, Member

Yıldız Technical University _____

Assoc. Prof. Dr. Reşat KŐŞKER, Member

Yıldız Technical University _____

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my advisor Asst. Prof. Dr. İbrahim Demir for the continuous support of my graduate study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my graduate study.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Dr. Ali Hakan Büyüklü and Assoc. Prof. Dr. Reşat Köşker, for their encouragement, insightful comments, and hard questions.

My sincere thanks also goes to my managers David Walter, Ben Duke and Ashley Flubacher for encouraging me in completion of this study.

Last but not the least, I would like to thank my family: my parents Necla and Metin Kanar and my brother Çağrı Kanar for supporting me spiritually throughout my life.

June, 2014

Çağdaş KANAR

TABLE of CONTENTS

Page

LIST of SYMBOLS

X	data matrix, consisting of the set of all data vectors
n	the number of row vectors in the data set
p	the number of elements in each row vector
L	the number of dimensions in the dimensionally reduced subspace
u	vector of empirical means
s	vector of empirical standard deviations
h	vector of all 1's
B	deviations from the mean of each column j of the data matrix
Z	z-scores
C	covariance matrix
R	correlation matrix
V	matrix consisting of the set of all eigenvectors of C
D	diagonal matrix consisting of the set of all eigenvalues of C
W	matrix of basis vectors
T	matrix consisting of n row vectors

LIST of ABBREVIATIONS

AIC	Akaike Information Criterion
CA	Correspondence Analysis
ChiSq	Chi-Square
DCT	Discrete Cosine Transform
DEV	Development Model
DF	Degrees of Freedom
EOF	Empirical Orthogonal Functions
FN	False Negaives
FP	False Positives
GAMMA	The Goodman-Kruskal Gamma method
GCONV	Convergence Criterion
GS	Gram–Schmidt Re-orthogonalization Algorithm
IDL	Interactive Data Language
Log L	Log-likelihood
LR	Logistic Regression
NIPALS	Non-linear Iterative Partial Least Squares
PCA	Principal Components Analysis
QPW	Quantity per Week
SC	Schwarz Criterion
SPW	Spend per Week
SVD	Singular Value Decomposition
TAU-A	Kendall's Tau-a
TN	True Negatives
TP	True Positives
TRISS	Trauma and Injury Severity Score
VAL	Validation Model
VPW	Visits per Week
WALD	Wald Chi-Square

LIST of FIGURES

Page

LIST of TABLES

Page

ABSTRACT

PREDICTING LAPSING CUSTOMERS WITH LOGISTIC REGRESSION APPROACH in RETAIL

Çağdaş KANAR

Department of Statistics

MSc. Thesis

Advisor: Asst. Prof. Dr. İbrahim DEMİR

Data mining is the process of exploring meaningful information in big and complex data to create a valuable business strategy. One of the major application fields of data mining is, predicting customers having a tendency of disconnecting from the services of the company. Also called as churn analysis, it provides predictive information to the companies to score customers having churn risk and then enables them developing retention strategies such as targeted campaigns.

This study is executed by using transaction data of customers enrolled in loyalty programme which belongs to a multinational retail company operating in Turkey. It is aimed to score customers having a churn tendency in next 13 weeks and then helping to develope retention strategies based on these scores.

In order to explore churn customer profiles , Factor Analysis and Logistic Regression methods are applied and results of the application is presented.

Keywords: Principal Components Analysis, Logistic Regression, Churn Analysis, Customer Scoring

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

ÖZET

PERAKENDE SEKTÖRÜNDE LOJİSTİK REGRESYON YAKLAŞIMIYLA KAYIP MÜŞTERİ TAHMİNLEMESİ

Çağdaş KANAR

İstatistik Anabilim Dalı

Yüksek Lisans Tezi

Tez Danışmanı: Yrd. Doç. Dr. İbrahim DEMİR

Veri madenciliği, büyük veri kümeleri içindeki anlamlı bilgiyi ortaya çıkarma sürecidir. Veri madenciliğinin yaygın olarak kullanıldığı uygulama alanlarından biri, ayrılma eğilimi gösteren müşterilerin tahmin edilmesidir. Churn adı verilen bu analiz, şirketlerin kaybetme potansiyeli olan müşterilerini elde tutmaya dönük özel pazarlama kampanyalarını geliştirmelerini sağlamaya yöneliktir. Bu çalışma, Türkiye'de perakende sektöründe faaliyet gösteren çok uluslu bir firmanın, ayrılma eğilimi gösteren müşterilerini belirleyerek; bu müşteriler üzerinde doğrulanan modelin geleceğe dönük skorlama ile gelecek 13 hafta içerisinde şirket hizmetlerinden ayrılma riski gösteren müşterileri tahminleme ve bu müşterilere dönük elde tutma stratejilerine katkı sağlamayı hedeflemektedir. Ayrılacak müşteri profilini belirlemek için Faktör Analizi ve Lojistik Regresyon Analizi teknikleri kullanılmış ve uygulamanın sonuçları sunulmuştur.

Anahtar Kelimeler: Temel Bileşenler Analizi, Lojistik Regresyon, Churn Analizi, Musteri Skorlama

SECTION 1

INTRODUCTION

1.1 Summary of Literature

Data mining refers to discover knowledge from a large amount of data. In this paper, we discuss the application of data mining including logistic regression and principal component analysis to predict the churn of loyalty card users of a retail chain. The retailers can take corresponding actions to retain the customers according to the suggestion of the models. With today's cost-cutting and intensive competitive pressure, more companies start to focus on Customer Relationship Management (CRM). The unknown future behaviors of the customers are quite important to CRM. Hence, it is of crucial importance to detect the customers' future decision then the company can take corresponding actions early [1]. The customers who stop using the company's products are usually called churners. Finding the churners can help companies retain their customers. Gustafsson, Johnson, and Roos [2] studied telecommunication services to examine the effects of customer satisfaction and behavior on customer retention. Results indicated a need for CRM managers to more accurately determine customer satisfaction in order to reduce customer churn. One of the major reasons for this is that it costs less to retain existing customers than to acquire new customers [3].

It costs up to five times as much to make a sale to a new customer as it does to make an additional sale to an existing customer [4]. And, it is becoming more evident that the only way to remain a leader in this industry is to not only be customer-driven but also focus on building long-term relationships.

Due to the development of information technology, many companies have accumulated a large amount of data. Analyzing this data can help the manager make the right marketing decision and pinpoint the right customer to market. Because of the large

amount of accumulated data and serious churn related to credit card holders, it is a very good field in which to predict churn.

Several studies have proved the effectiveness of the power of customer retention. A bank is able to increase its profits by 85% due to a 5% improvement in the retention rate [5]. Van den Poel and Larivière [6] calculated the financial impact of a one percent increase in customer retention rate. The power of the model can stay for a relatively long time. According to the research of Neslin, the churn models in the data typically still perform very well if used to predict churn for a database compiled 3 months after the calibration data [7].

As the economy and new marketing techniques develop in Turkey, a large amount loyalty cards are issued. But many of the card holders are not active (or called churn holders). With increasing retail competition, customers are able to choose among multiple retailers and easily exercise their right of switching from one retailer to another. If retailers can predict future behaviors before the customers stop shopping from them, they can market to retain these customers.

1.2 Purpose of Thesis

The main purpose of this paper is not to provide a new data mining algorithm, but to focus on the application of the churn prediction, to provide a framework of understanding the knowledge of the card holders' hidden pattern using the data of a retailer. From the data preparation to useful knowledge, the goal is application of churn prediction. In this paper, we introduce a way to complete churn prediction considering spend and visits. The rest of the paper is organized as follows. The definition of churn and the summary of the algorithms and criteria are introduced in Section 2. The model used in the research and the modeling process based on principal components and logistic regression are presented in Section 3. In Section 4, we conclude and present the results.

1.3 Findings

In this model based on 8 weeks of loyalty card customer transaction data, spend, visits, weeks, spend per week, quantity per week and visits per week are researched to calculate the probability of a loyal customer will churn in the next 13 weeks.

The model requires principal components built to reduce the number of variables. According to principal components analysis , 6 variables are reduced to 2 factors where spend, spend per week and quantity per week are loaded into factor 1 and visits, weeks and visits per week are loaded into factor 2. Afterwards, through using these factors logistic regression model applied to identify key levers of short-term (13 weeks) churn by profiling of the risk score deciles.

SECTION 2

DATA and METHOD

Model uses 8 weeks of transaction data (spend, quantity, visits, weeks visited) to develop probability estimate. SAS programming used to set up the churn model. The model code simply consists of two parts as model builder and scoring. In the first part, firstly factor loadings are calculated by "proc factor" procedure, secondly "proc logistic" procedure is used to to build a simple customer retention model from basic shopping behavior over 8 weeks and in the second part "proc score" procedure used to estimate probability of a loyal customer who will churn in the next 13 weeks.

2.1 Data

2.1.1 The Data Source

An anonymous Turkish retailer provides the data for this study and is extracted from a data warehouse. All of the data is integrated at the level of the customer. No matter when the customers open the loyalty card accounts at any branch of the retailer, the data warehouse can identify the customers by name and identification number of the customers. All of the customers in the warehouse are indexed by an unique customer number. The data warehouse records all the past changes to the card. Taking the balance of the card, for example, once the balance of the card changed, there will be an additional row for the new balance and the last balance is retained.

Model uses 8 weeks of transaction data (spend, quantity, visits, weeks visited) to develop probability estimate (same 8 weeks used for Shopper Habits segmentation used as a infrastructure for the model).

2.1.2 Shopper Habits Segmentation

Shopper Habits is the best operational view of behavioural engagement. Although it does not prove emotional connection or true 'loyalty' to a retailer, it has consistently been proven to align with the following metrics :

Table 2.1 – Shopper habit metrics

Metric	**Observed Alignment**
Share of Wallet	Loyal customers typically spend more in the category but also spend a greater share of their available money with that retailer
Retention	Loyal customers are more likely to continue shopping the retailer from one period to the next
Breadth	Loyal customers engage more broadly across the store or the extended brand.

2.1.3 Shopper Habits Segmentation Inputs

The 4 core ingredients into Shopper Habits are the timeframe for scoring and then within that Visit Frequency, Spend and Recency.

Table 2.2 – Shopper habit model inputs

Metric	**Rationale & Best Practice**
Timeframe	• Score over a timeframe that covers the buying cycle and seasonality • Ideally 2+ years of data to understand what % of customers repeat within that timeframe (NOTE what do we do if repurchase is much longer than this e.g. auto retailers, fine jewellery etc) • Score the segmentation using the period of time that includes 90+% of repeaters repeating

	• The greater the frequency of customer purchase cycles, the tighter/smaller that segmentation timeframes can be • Set the length of time used for scoring to smooth any major spikes in sales/seasonality to ensure that the segmentation is not more volatile
Visit frequency	• Provides a measure of purchase stability – we look at how many trips a customer makes in the scoring timeframe as well as trip patterns • The more often a customer visits the store (no matter what the channel), the easier it is to engage with them • The more often a customer visits the store, the more often she is exposed to the marketing and messaging, and the more she is building a habit of coming to the retailer to fill her needs
Spend	• Can be a variety of metrics, we typically seek a spend metric with low correlation to visit frequency. Metrics reviewed might be: • Total spend in timeframe • Spend per visit • Average unit retail • Spend per quarter/month/week shopped • Spend & visit thresholds are initially set as High (top 20%), Medium (30%) and Low (Remaining 50%) of customers • Spend can be rounded up or down slightly to create simple breaks that are easy for the business to retain and understand • Polynomial line fitting (order/power 2) can also assist in visibly showing skew in the spend or visit distribution • Spend is one of the strongest ‘votes’ a customer can give to a

	retailer; they are investing in the retailer's goods and services • Confirmation bias – the more a customer spends at a retailer, the more she convinces herself that it was a smart decision
Recency	• Allows us to filter among customers with only a small number of purchase visits to differentiate new customers (recent shoppers) vs. lapsing (have not shopped recently) • Recency has shown correlation to retention; the longer a customer has gone since last visit, the higher the likelihood of a decline in shopping habit

2.1.4 Shopper Habit Segment Definitions

Table 2.3 – Shopper habit segmentation

High Level	**Low Level**
Loyals a.k.a Best Customers, Loyalists, Most Engaged	Premium Loyal Valuable
Non Loyals a.k.a Unengaged, Opportunity, Less Loyal	Potential Uncommitted Lapsing Goneaway

2.1.5 Data Description

The original data in our study consists of three main tables and the largest table has over 1 million records. Customer metrics table includes 69 variables and unique for each week. In customer metrics table we flag customers' weekly loyalty segments based on the methodology explained in section 2.1.3. Since we run the model over 8 weeks period we have 8 unique tables coming alongside with customer metrics table. Customer spend table includes 5 variables which are customer id, customer code, weekly spend, weekly quantities purchased and weekly shopping visits made. Similar to customer metrics table this table repeats for each week and so we have 8 unique tables coming alongside with customer spend table. Lastly, we use time table which includes

12 variables including date, week, month, quarter and year variables related to transactions.

According to the design of the model which will be explained in model section, we calculated the derivative variables. There are 86 variables reflecting the complete information of the customer. The independent variables are calculated from the data during the observation period of September 2012 through November 2012 (8 weeks) and 13 weeks post end week to capture churners which starts at the end week of observation period and checks 13 weeks starting from this week. So, the observation period for checking churners is between November 2012 and February 2013. The independent variable (Y) is calculated from the data during the performance period of November 2012 through February 2013 (13 weeks). According the definition of Section 2.1.5, we define that the customer who was in loyals segment (i.e. Premium or Valuable) during the 8 weeks observation period and moved to non-loyals segment (i.e. Lapsing or Goneaway) during the post-observation period is a churner.

y_i =
- 1 if loyal customer i moved to non-loyal segment in post observation period
- 0 if loyal customer i hasn't moved to non-loyal segment in post observation period

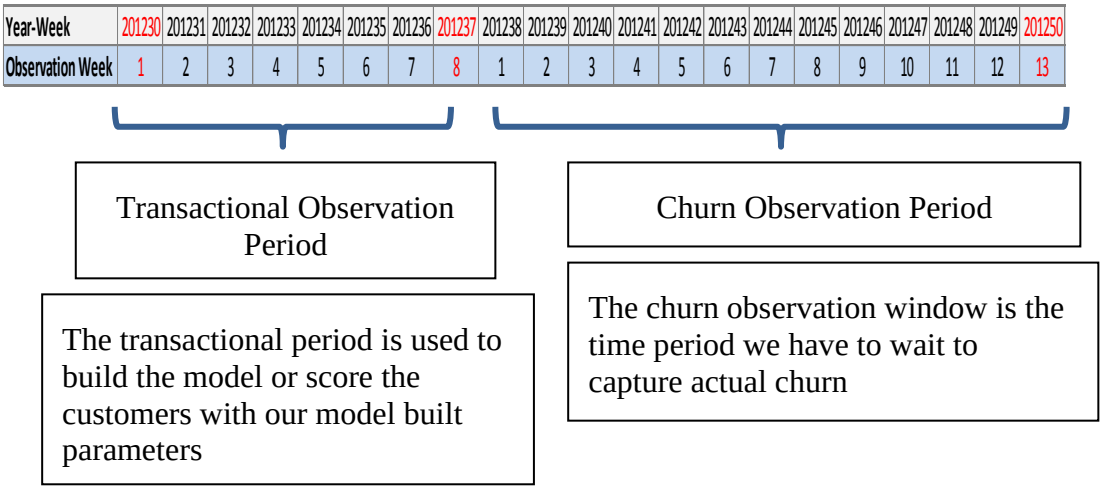


Figure 2.1 – Churn model illustration

2.2 Principal Components Analysis

The central idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of a large number of interrelated variables, while retaining as much as possible of the variation present in the data set.

This is done by transforming to a new set of variables, the principal components (PCs), which are uncorrelated, and which are ordered so that the first few retain most of the variation present in all of the original variables. The number of principal components is less than or equal to the number of original variables. This transformation is defined in a way that the first principal component has the largest possible variance, and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to the preceding components. Principal components are ensured to be independent if the data set is jointly normally distributed. And also PCA is sensitive to the relative scaling of the original variables.

We can also name it as discrete Karhunen–Loève transform (KLT) in signal processing, the Hotelling transform in multivariate quality control, proper orthogonal decomposition (POD) in mechanical engineering, singular value decomposition (SVD) of X, eigenvalue decomposition (EVD) of XTX in linear algebra, factor analysis, Eckart–Young theorem, or Schmidt–Mirsky theorem in psychometrics, empirical orthogonal functions (EOF) in meteorological science, empirical eigenfunction decomposition , empirical component analysis , quasiharmonic modes, spectral decomposition in noise and vibration, and empirical modal analysis in structural dynamics. This can change according to the field of the study.

PCA was invented in 1901 by Karl Pearson,[8] and it was as an analogue of the principal axes theorem in mechanics; it was later independently developed by Harold Hotelling in the 1930s [9]. The method is mostly used as a tool in exploratory data analysis and for making predictive models. PCA can be created by eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix, usually after mean centering the data matrix for each attribute [10]. The results of a PCA are usually discussed in terms of component scores, sometimes called factor scores , and loadings [11].

PCA is a very simple method among eigenvector-based multivariate analyses. In general, its operation can be considered as exploring the internal structure of the data in a way that best explains the variance of the data. If a multivariate dataset is visualised as a set of coordinates in a high-dimensional data space , PCA can give the user with a lower-dimensional picture, a projection of this object when viewed from its most informative viewpoint. This is done by using only the first few principal components so that the dimensionality of the transformed data is reduced.

So obtaining the principal components is entirely equivalent to finding the eigenvectors of the covariance matrix: each eigenvector gives us one principal component. PCA is also related to canonical correlation analysis (CCA). CCA defines coordinate systems which describe the cross-covariance between two datasets while PCA defines a new orthogonal coordinate system that describes the variance in a single dataset.

2.2.1 Details of Principal Component Analysis

PCA can be defined [12] as an orthogonal linear transformation which transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate, the second greatest variance on the second coordinate, and so on.

If we consider a data matrix, X, with zero empirical mean , where each of the n rows represents a different repetition of the experiment, and each of the p columns gives a particular kind of a data.

Mathematically, the transformation is defined by a set of p-dimensional vectors of weights or loadings $w_{(k)} = (w_1,...,w_p)_{(k)}$ that map each row vector $X_{(i)}$ of X to a new vector of principal component scores $t_{(i)} = (t_1,...,t_p)_{(i)}$ given by

$$t_{k(i)} = x_{(i)}.w_{(k)} \tag{2.1}$$

in such a way that the individual variables of t considered over the data set successively inherit the maximum possible variance from x, with each loading vector w constrained to be a unit vector.

2.2.1.1 First Component of PCA

The first loading vector $w_{(1)}$ thus has to satisfy the equation below:

$$w_{(1)} = argmax_{\|w\|=1}\left\{\sum_i (t_1)^2{}_{(i)}\right\} = argmax_{\|w\|=1}\left\{\sum_i \left(x_{(i)}.w\right)^2\right\} \tag{2.2}$$

Similarly, writing this in matrix form gives

$$w_{(1)} = argmax_{\|w\|=1}\{\|Xw\|\} = argmax_{\|w\|=1}\{w^T X^T Xw\} \tag{2.3}$$

Since $w_{(1)}$ has been defined to be a unit vector, it also satisfies

$$w_{(1)} = argmax_{\|w\|=1} \left\{ \frac{w^T X^T X w}{w^T w} \right\} \quad (2.4)$$

The quantity to be maximised can be recognised as a Rayleigh quotient. A standard result for a symmetric matrix such as X^TX is that the quotient's maximum possible value is the largest eigenvalue of the matrix, which occurs when w is the corresponding eigenvector.

After finding $w_{(1)}$, the first component of a data vector $x_{(i)}$ can then be given as a score $t_{1(i)} = x_{(i)} \cdot w_{(1)}$ in the transformed co-ordinates, or as the corresponding vector in the original variables, $\{x_{(i)} \cdot w_{(1)}\} w_{(1)}$.

2.2.1.2 Further Components

The kth component can be found by subtracting the first k − 1 principal components from X:

$$\widehat{X}_{k-1} = X - \sum_{s=1}^{k-1} X_{w(s)} w_{(s)}^T \quad (2.5)$$

and then finding the loading vector which extracts the maximum variance from this new data matrix

$$w_{(k)} = arg\ max_{\|w\|=1} \left\{ \left\| \hat{X}_{k-1} w \right\|^2 \right\} = arg\ max_{\|w\|=1} \left\{ \frac{w^T \hat{x}_{k-1}^T \hat{x}_{k-1} w}{w^T w} \right\} \quad (2.6)$$

It turns out that this gives the remaining eigenvectors of XTX, with the maximum values for the quantity in brackets given by their corresponding eigenvalues.

The kth principal component of a data vector x(i) can therefore be given as a score tk(i) = x(i) · w(k) in the transformed co-ordinates, or as the corresponding vector in the space of the original variables, {x(i) · w(k)} w(k), where w(k) is the kth eigenvector of XTX.

The full principal components decomposition of X can therefore be given as

$$T = XW \quad (2.7)$$

where W is a p-by-p matrix whose columns are the eigenvectors of X^TX.

2.2.1.3 Covariances of PCA

X^TX can be recognised as proportional to the empirical sample covariance matrix of the dataset X.

The sample covariance Q between two of the different principal components over the dataset is given by

$$Q(PC_{(j)}, PC_{(k)}) \propto (X_{w(j)}).(X_{w(k)}) \tag{2.8}$$

$$= w_{(j)}^T X^T X_{w(k)}$$

$$= w_{(j)}^T \lambda_{(k)} w_{(k)}$$

$$= \lambda_{(k)} w_{(j)}^T w_{(k)}$$

where the eigenvector property of $w_{(k)}$ has been used to move from line 2 to line 3. However eigenvectors $w_{(j)}$ and $w_{(k)}$ corresponding to eigenvalues of a symmetric matrix are orthogonal (if the eigenvalues are different), or can be orthogonalised (if the vectors happen to share an equal repeated value). The product in the final line is therefore zero; there is no sample covariance between different principal components over the dataset.

Another way to characterise the principal components transformation is therefore as the transformation to coordinates which diagonalise the empirical sample covariance matrix.

In matrix form, the empirical covariance matrix for the original variables can be written

$$Q \propto X^T X = W \Lambda W^T \tag{2.9}$$

The empirical covariance matrix between the principal components becomes

$$W^T Q W \propto W^T W \Lambda W^T W = \Lambda \tag{2.10}$$

where Λ is the diagonal matrix of eigenvalues $\lambda_{(k)}$ of X^TX

($\lambda_{(k)}$ being equal to the sum of the squares over the dataset associated with each component k: $\lambda_{(k)} = \Sigma_i \, t_k^2{}_{(i)} = \Sigma_i \, (x_{(i)} \cdot w_{(k)})^2$) (2.11)

2.2.1.4 Dimensionality Reduction

Transformation T = X W maps a data vector $x_{(i)}$ from an original space of p variables to a new space of p variables which are not correlated over the dataset. However, not all the principal components need to be kept. Keeping only the first L principal

components, produced by using only the first L loading vectors, gives the following transformation

$$T_L = XW_L$$

where the matrix T_L now has n rows but only L columns. By construction, of all the transformed data matrices with only L columns, this score matrix maximises the variance in the original data that has been preserved, while minimising the total squared reconstruction error $||T - T_L||^2$.

Such dimensionality reduction can be a very useful step for visualising and processing high-dimensional datasets, while still retaining as much of the variance in the dataset as possible. For example, selecting L = 2 and keeping only the first two principal components finds the two-dimensional plane through the high-dimensional dataset in which the data is most spread out, so if the data contains clusters these too may be most spread out, and therefore most visible to be plotted out in a two-dimensional diagram; whereas if two directions through the data are chosen at random, the clusters may be much less spread apart from each other, and may in fact be much more likely to substantially overlay each other, making them indistinguishable.

Similarly, in regression analysis, the larger the number of explanatory variables allowed, the greater is the chance of overfitting the model, producing conclusions that fail to generalise to other datasets. One approach, especially when there are strong correlations between different possible explanatory variables, is to reduce them to a few principal components and then run the regression against them, a method called principal component regression.

Dimensionality reduction may also be appropriate when the variables in a dataset are noisy. If each column of the dataset contains independent identically distributed Gaussian noise, then the columns of T will also contain similarly identically distributed Gaussian noise (such a distribution is invariant under the effects of the matrix W, which can be thought of as a high-dimensional rotation of the co-ordinate axes). However, with more of the total variance concentrated in the first few principal components compared to the same noise variance, the proportionate effect of the noise is less and the first components achieve a higher signal-to-noise ratio. PCA thus can have the effect of concentrating much of the signal into the first few principal components, which can

usefully be captured by dimensionality reduction; while the later principal components may be dominated by noise, and so disposed of without great loss.

2.2.1.5 Singular Value Decomposition

The SVD can be considered to be a general method for understanding change of basis. The principal components transformation can also be associated with another matrix factorisation, the singular value decomposition (SVD) of X,

$$X = U \sum W^T \tag{2.12}$$

Here Σ is a n-by-p rectangular diagonal matrix of positive numbers $\sigma_{(k)}$, called the singular values of X; U is an n-by-n matrix, the columns of which are orthogonal unit vectors of length n called the left singular vectors of X; and W is a p-by-p whose columns are orthogonal unit vectors of length p and called the right singular vectors of X.

In terms of this factorisation, the matrix X^TX can be written

$$X^TX = W \sum U^T U \sum W^T$$

$$= W \sum^2 W^T \tag{2.13}$$

Comparison with the eigenvector factorisation of X^TX establishes that the right singular vectors W of X are equivalent to the eigenvectors of X^TX, while the singular values $\sigma_{(k)}$ of X are equal to the square roots of the eigenvalues $\lambda_{(k)}$ of X^TX.

Using the singular value decomposition the score matrix T can be written as

$$T = XW$$

$$= U \sum W^T W$$

$$= U\sum \tag{2.14}$$

so each column of T is given by one of the left singular vectors of X multiplied by the corresponding singular value.

Efficient algorithms exist to calculate the SVD of X without having to form the matrix X^TX, so computing the SVD is now the standard way to calculate a principal components analysis from a data matrix, unless only a handful of components are required.

As with the eigendecomposition, a truncated n-by-L score matrix TL can be got by considering only the first L largest singular values and their singular vectors:

$$T_L = U_L \sum_L = XW_L \quad (2.15)$$

The truncation of a matrix M or T using a truncated singular value decomposition in this way produces a truncated matrix that is the nearest possible matrix of rank L to the original matrix, in the sense of the difference between the two having the smallest possible Frobenius norm, a result known as the Eckart–Young theorem.

2.2.2 Some Further Considerations on PCA

At this point, it is worth to mention about some further considerations about PCA. If we thinks there's given a set of points in Euclidean space, the first principal component should correspond to a line which passes through the multidimensional mean and minimizes the sum of squares of the distances of the points from the line. The second principal component corresponds to the same concept after all correlation with the first principal component has been subtracted from the points. The singular values in Σ are the square roots of the eigenvalues of the matrix XTX. Each eigenvalue is proportional to the portion of the variance that is correlated with each eigenvector. The sum of all the eigenvalues is equal to the sum of the squared distances of the points from their multidimensional mean. PCA essentially rotates the set of points around their mean in order to align with the principal components. This moves as much of the variance as possible into the first few dimensions. Then, the values in the other dimensions, have tendency to be small and may be dropped with minimum loss of information. So, PCA is often used in this manner for dimensionality reduction. PCA has the distinction of being the optimal orthogonal transformation for keeping the subspace that has largest variance. This advantage, however, needs greater computational requirements if compared, for example to the discrete cosine transform, and in particular to the DCT-II which is simply known as the DCT. Nonlinear dimensionality reduction techniques tend to be more complex than PCA.

PCA is sensitive to the scaling of the variables. If we have just two variables and they have the same sample variance and are positively correlated, then the PCA will entail a rotation by 45° and the loadings for the two variables with respect to the principal component will be equal. But if we multiply all values of the first variable by 100, then

the principal component will be almost the same as that variable, with a small contribution from the other variable, whereas the second component will be almost aligned with the second original variable. This means that whenever the different variables have different units PCA is a somewhat arbitrary method of analysis. One way of making the PCA less arbitrary is to use variables scaled so as to have unit variance, by standardizing the data and hence use the autocorrelation matrix instead of the autocovariance matrix as a basis for PCA. However, this compresses the fluctuations in all dimensions of the signal space to unit variance.

Mean subtraction is necessary for performing PCA to ensure that the first principal component describes the direction of maximum variance. If mean subtraction is not performed, the first principal component might instead correspond more or less to the mean of the data. A mean of zero is needed for finding a basis that minimizes the mean square error of the approximation of the data [13].

PCA is equivalent to empirical orthogonal functions (EOF), a name which is used in meteorology.

A neural network with a linear hidden layer is similar to PCA. Upon convergence, the weight vectors of the K neurons in the hidden layer will form a basis for the space spanned by the first K principal components. Unlike PCA, this technique will not necessarily produce orthogonal vectors.

PCA is a popular primary technique in pattern recognition. It is not, however, optimized for class separability [14]. An alternative is the linear discriminant analysis, which take s this into account.

Another application of PCA is reducing the number of parameters in the process of generating computational models of oil reservoirs [15].

2.2.3 Symbols and Abbreviations

Table 2.4 Table of symbols and abbreviations used in PCA

Symbol	Meaning	Dimensions	Indices
$X = \{X[i,j]\}$	data matrix, consisting of the set of all data vectors, one vector per row	$n \times p$	$i = 1 \ldots n$

Table of symbols and abbreviations used in PCA(continued)

n	the number of row vectors in the data set	1×1	*scalar*
p	the number of elements in each row vector	1×1	*scalar*
L	the number of dimensions in the dimensionally reduced subspace, $1 \leq L \leq p$	1×1	*scalar*
$u = \{u[j]\}$	vector of empirical means, one mean for each column *j* of the data matrix	$p \times 1$	$j = 1 \ldots p$
$s = \{s[j]\}$	vector of empirical standard deviations, one standard deviation for each column j of the data matrix	$p \times 1$	$j = 1 \ldots p$
$h = \{h[i]\}$	vector of all 1's	$1 \times n$	$i = 1 \ldots n$
$B = \{B[i,j]\}$	deviations from the mean of each column *j* of the data matrix	$n \times p$	$i = 1 \ldots n$ $j = 1 \ldots p$
$Z = \{Z[i,j]\}$	z-scores, computed using the mean and standard deviation for each row *m* of the data matrix	$n \times p$	$i = 1 \ldots n$ $j = 1 \ldots p$
$C = \{C[k,l]\}$	covariance matrix	$p \times p$	$k = 1 \ldots p$ $l = 1 \ldots p$

Table of symbols and abbreviations used in PCA(continued)

$R = \{R[k,l]\}$	correlation matrix	$p \times p$	$k = 1 \ldots p$ $l = 1 \ldots p$
$V = \{V[j,k]\}$	matrix consisting of the set of all eigenvectors of C, one eigenvector per column	$p \times p$	$j = 1 \ldots p$ $k = 1 \ldots p$
$D = \{D[k,l]\}$	diagonal matrix consisting of the set of all eigenvalues of C along its principal diagonal, and 0 for all other elements	$p \times p$	$k = 1 \ldots p$ $l = 1 \ldots p$
$W = \{W[j,k]\}$	matrix of basis vectors, one vector per column, where each basis vector is one of the eigenvectors of C, and where the vectors in W are a sub-set of those in V	$p \times L$	$j = 1 \ldots p$ $k = 1 \ldots L$
$T = \{T[i,k]\}$	matrix consisting of *n* row vectors, where each vector is the projection of the corresponding data vector from matrix X onto the basis vectors contained in the columns of matrix W	$n \times L$	$i = 1 \ldots n$ $k = 1 \ldots L$

2.2.4 Properties and Limitations of PCA

We'll describe properties and limitations of Principal Components Analysis in this section.

2.2.4.1 Properties

Property 1: For any integer q, 1 ≤ q ≤ p, consider the orthogonal linear transformation

$$y = B'x \tag{2.16}$$

where y is a q-element vector and B' is a (q × p) matrix, and let $\sum_y = B' \sum B$ be the variance-covariance matrix for y.Then the trace of $\sum_y$,denoted $tr(\sum_y)$, is

maximized by taking $B = A_q$, where A_q consists of the first q columns of A (B' is the transposition of B)

Property 2: Consider again the orthonormal transformation

$$y = B'\,x \tag{2.17}$$

with x, B, A and $\sum_y$ defined as before. Then $tr(\sum_y)$ is minimized by taking $B = A_q^*$ where A_q^* consists of the last q columns of A.

The statistical implication of this property is that the last few PCs are not simply unstructured left-overs after removing the important PCs. Because these last PCs have variances as small as possible they are useful in their own right. They can help to detect unsuspected near-constant linear relationships between the elements of x, and they may also be useful in regression, in selecting a subset of variables from x, and in outlier detection.

Property 3: (the Spectral Decomposition of ∑)

$$\sum = \lambda_1\, \alpha_1\alpha_1' + \lambda_2\, \alpha_2\alpha_2' + \cdots + \lambda_p\, \alpha_p\alpha_p' \tag{2.18}$$

Before we look at its usage, we first look at diagonal elements,

$$var(x_j) = \sum_{k=1}^{P} \lambda_k\, \alpha_{kj}^2 \tag{2.19}$$

Then, perhaps the main statistical implication of the result is that not only can we decompose the combined variances of all the elements of x into decreasing contributions due to each PC, but we can also decompose the whole covariance matrix into contributions $\lambda_k\, \alpha_k\alpha_k'$ from each PC. Although not strictly decreasing, the elements of $\lambda_k\, \alpha_k\alpha_k'$ will tend to become smaller as k increases, as $\lambda_k\, \alpha_k\alpha_k'$ decreases for increasing k , whereas the elements of α_k tend to stay about the same size because of the normalization constraints: $\alpha_k\alpha_k' = 1$, $k = 1,2, \ldots, p$

2.2.4.2 Limitations

As noted above, the results of PCA depend on the scaling of the variables.
The applicability of PCA is limited by certain assumptions [17] made in its derivation.

2.2.4.3 Relationship Between PCA and Information Theory

The claim that the PCA used for dimensionality reduction preserves most of the information of the data is misleading. Indeed, without any assumption on the signal model, PCA cannot help to reduce the amount of information lost during dimensionality reduction, where information was measured using Shannon entropy [18].

Under the assumption that

$$x = s + n \tag{2.20}$$

i.e., that the data vector x is the sum of the desired information-bearing signal s and a noise signal n one can show that PCA can be optimal for dimensionality reduction also from an information-theoretic point-of-view.

In particular, Linsker showed that if s is Gaussian and n is Gaussian noise with a covariance matrix proportional to the identity matrix, the PCA maximizes the mutual information $I(y; s)$ between the desired information s and the dimensionality-reduced output $y = W_L^T x$.[19]

If the noise is still Gaussian and has a covariance matrix proportional to the identity matrix, but the information-bearing signal s is non-Gaussian, PCA at least minimizes an upper bound on the information loss, which is defined as [20][21]

$$I(x; s) - I(y; s) \tag{2.21}$$

The optimality of PCA is also preserved if the noise n is iid and at least more Gaussian than the information-bearing signal s [22]. In general, even if the above signal model holds, PCA loses its information-theoretic optimality as soon as the noise n becomes dependent.

2.2.5 Covariance Method in Computing PCA

The following is a detailed description of PCA using the covariance method. But note that it is better to use the singular value decomposition.

The goal is to transform a given data set X of dimension p to an alternative data set Y of smaller dimension L. Equivalently, we are seeking to find the matrix Y, where Y is the Karhunen–Loève transform (KLT) of matrix X:

$$Y = \mathbb{KLT}\{X\} \tag{2.22}$$

2.2.5.1 Organize the Data Set

Suppose you have data comprising a set of observations of p variables, and you want to reduce the data so that each observation can be described with only L variables, L < p. Suppose further, that the data are arranged as a set of n data vectors $x_1 \ldots x_n$ with each x_i representing a single grouped observation of the p variables.

- Write $x_1 \ldots x_n$ as row vectors, each of which has p columns.
- Place the row vectors into a single matrix X of dimensions n × p.

2.2.5.2 Calculate the Empirical Mean

- Find the empirical mean along each dimension j = 1, ..., p.
- Place the calculated mean values into an empirical mean vector u of dimensions p × 1.

$$u[j] = \frac{1}{N}\sum_{i=1}^{n} X[i,j] \tag{2.23}$$

2.2.5.3 Calculate the Deviations From the Mean

Mean subtraction is an integral part of the solution towards finding a principal component basis that minimizes the mean square error of approximating the data. Hence we proceed by centering the data as follows:

- Subtract the empirical mean vector u from each row of the data matrix X.
- Store mean-subtracted data in the n × p matrix B.

$$B = X - hu^T \tag{2.24}$$

where h is an n × 1 column vector of all 1s:

$$h[i] = 1 \quad for\ i = 1, \ldots, n \tag{2.25}$$

2.2.5.4 Find the Covariance Matrix

- Find the p × p empirical covariance matrix C from the outer product of matrix B with itself:

$$C = \frac{1}{n-1} B^*.B \tag{2.26}$$

where * is the conjugate transpose operator. Note that if B consists entirely of real numbers, which is the case in many applications, the "conjugate transpose" is the same as the regular transpose.

- Please note that outer products apply to vectors. For tensor cases we should apply tensor products, but the covariance matrix in PCA is a sum of outer products between its sample vectors; indeed, it could be represented as B*.B. See the covariance matrix sections on the discussion page for more information.
- The reasoning behind using N-1 instead of N to calculate the covariance is Bessel's correction.

2.2.5.5 Find the Eigenvectors and Eigenvalues of the Covariance Matrix

- Compute the matrix V of eigenvectors which diagonalizes the covariance matrix C:

$$V^{-1}CV = D \tag{2.27}$$

where D is the diagonal matrix of eigenvalues of C. This step will typically involve the use of a computer-based algorithm for computing eigenvectors and eigenvalues. These algorithms are readily available as sub-components of most matrix algebra systems, such as SAS, R, MATLAB,[23][24] Mathematica,[25] SciPy, IDL (Interactive Data Language), or GNU Octave as well as OpenCV.

- Matrix D will take the form of an M × M diagonal matrix, where

$$D[k,l] = \lambda_k \quad for\ k = 1 = j \tag{2.28}$$

is the jth eigenvalue of the covariance matrix C, and

$$D[k,l] = 0 \quad for\ k \neq 1 \tag{2.29}$$

- Matrix V, also of dimension p × p, contains p column vectors, each of length p, which represent the p eigenvectors of the covariance matrix C.
- The eigenvalues and eigenvectors are ordered and paired. The jth eigenvalue corresponds to the jth eigenvector.

2.2.5.6 Rearrange the Eigenvectors and Eigenvalues

- Sort the columns of the eigenvector matrix V and eigenvalue matrix D in order of decreasing eigenvalue.
- Make sure to maintain the correct pairings between the columns in each matrix.

2.2.5.7 Compute the Cumulative Energy Content for Each Eigenvector

- The eigenvalues represent the distribution of the source data's energy among each of the eigenvectors, where the eigenvectors form a basis for the data. The cumulative energy content g for the jth eigenvector is the sum of the energy content across all of the eigenvalues from 1 through j:

$$g[j] = \sum_{k=1}^{j} D[k,k] \quad for \;\; j = 1, \dots, p \tag{2.30}$$

2.2.5.8 Select a Subset of the Eigenvectors as Basis Vectors

- Save the first L columns of V as the p × L matrix W:

$$W[k,l] = V[k,l] \quad for \;\; k = 1, \dots, p \;\;\; l = 1, \dots, L \tag{2.31}$$

where

$$1 \leq L \leq p \tag{2.32}$$

- Use the vector g as a guide in choosing an appropriate value for L. The goal is to choose a value of L as small as possible while achieving a reasonably high value of g on a percentage basis. For example, you may want to choose L so that the cumulative energy g is above a certain threshold, like 90 percent. In this case, choose the smallest value of L such that

$$\frac{g[L]}{g[p]} \geq 0.9 \tag{2.33}$$

2.2.5.9 Convert the Source Data to Z-Scores

- Create an p × 1 empirical standard deviation vector s from the square root of each element along the main diagonal of the diagonalized covariance matrix C. (Note, that scaling operations do not commute with the KLT thus we must scale by the variances of the already-decorrelated vector, which is the diagonal of C) :

$$s = \{s[j]\} = \left\{\sqrt{C[j,j]}\right\} \quad for\ j = 1, \dots, p \tag{2.34}$$

- Calculate the n × p z-score matrix:

$$Z = \frac{B}{h.s^T} \tag{2.35}$$

- Note: While this step is useful for various applications as it normalizes the data set with respect to its variance, it is not integral part of PCA/KLT

2.2.5.10 Project the Z-Scores of the Data onto the New Basis

- The projected vectors are the columns of the matrix

$$T = Z.W = \mathbb{KLT}\{X\} \tag{2.36}$$

- The rows of matrix T represent the Karhunen–Loeve transforms (KLT) of the data vectors in the rows of matrix X.

2.2.6 Derivation of PCA Using the Covariance Method

Let X be a d-dimensional random vector expressed as column vector. Without loss of generality, assume X has zero mean.

We want to find $\{*\}$ a $d \times d$ orthonormal transformation matrix P so that PX has a diagonal covariant matrix (i.e. PX is a random vector with all its distinct components pairwise uncorrelated).

A quick computation assuming P were unitary yields:

$$\begin{aligned} var(PX) &= \mathbb{E}[PX(PX)^{\dagger}] \\ &= \mathbb{E}[PX\ X^{\dagger}P^{\dagger}] \\ &= P\mathbb{E}[X\ X^{\dagger}]P^{\dagger} \\ &= P\ var(X)P^{-1} \end{aligned} \tag{2.37}$$

Hence $(*)$ holds if and only if $var(X)$ were diagonalisable by P.

This is very constructive, as $var(X)$ is guaranteed to be a non-negative definite matrix and thus is guaranteed to be diagonalisable by some unitary matrix.

2.2.6.1 Iterative Computation

In practical implementations especially with high dimensional data , the covariance method is rarely used because it is not efficient. One way to compute the first principal component efficiently [22] is shown in the following pseudo-code, for a data matrix X with zero mean, without ever computing its covariance matrix

$r = a\ random\ vector\ of\ lenght\ p$

$doc\ times:$

$s = 0$ ($a\ vector\ of\ lenght\ p$)

$for\ each\ row\ x \in X$

$s = s + (x.r)x$

$r = \frac{s}{|s|}$

return r.

This algorithm is simply an efficient way of calculating XTX r, normalizing, and placing the result back in r. It avoids the np2 operations of calculating the covariance matrix. r will typically get close to the first principal component of X within a small number of iterations, c. Subsequent principal components can be computed by subtracting component r from X and then repeating this algorithm to find the next principal component. However this simple approach is not numerically stable if more than a small number of principal components are required, because imprecisions in the calculations will additively affect the estimates of subsequent principal components. More advanced methods build on this basic idea, as with the closely related Lanczos algorithm.

One way to compute the eigenvalue that corresponds with each principal component is to measure the difference in mean-squared-distance between the rows and the centroid, before and after subtracting out the principal component. The eigenvalue that corresponds with the component that was removed is equal to this difference.

2.2.6.2 The NIPALS Method

For very high-dimensional datasets, such as those generated in the omics sciences it is usually only necessary to compute the first few PCs. The non-linear iterative partial least squares (NIPALS) algorithm calculates t_1 and w_1^T from X. The outer

product, $t_1 w_1^T$ can then be subtracted from X leaving the residual matrix E_1. This can be then used to calculate subsequent PCs [26]. This results in a dramatic reduction in computational time since calculation of the covariance matrix is avoided.

However, for large data matrices, or matrices that have a high degree of column collinearity, NIPALS suffers from loss of orthogonality due to machine precision limitations accumulated in each iteration step [27] A Gram–Schmidt (GS) re-orthogonalization algorithm is applied to both the scores and the loadings at each iteration step to eliminate this loss of orthogonality [28].

2.2.7 Relation Between PCA and K-means Clustering

It has been shown recently [29] [30] that the relaxed solution of K-means clustering, specified by the cluster indicators, is given by the PCA principal components, and the PCA subspace spanned by the principal directions is identical to the cluster centroid subspace specified by the between-class scatter matrix. Thus PCA automatically projects to the subspace where the global solution of K-means clustering lies, and thus facilitates K-means clustering to find near-optimal solutions [44].

2.2.8 Relation Between PCA and Factor Analysis

Principle components creates variables that are linear combinations of the original variables. The new variables have the property that the variables are all orthogonal. The principle components can be used to find clusters in a set of data. PCA is a variance-focused approach seeking to reproduce the total variable variance, in which components reflect both common and unique variance of the variable. PCA is generally preferred for purposes of data reduction but not when detect the latent construct or factors.

Factor analysis is similar to principle component analysis, in that factor analysis also involves linear combinations of variables. Different from PCA, factor analysis is a correlation-focused approach seeking to reproduce the inter-correlations among variables, in which the factors “represent the common variance of variables, excluding unique variance [31]. Factor analysis is generally used when the research purpose is detecting data structure or causal modeling.

2.2.9 Correspondence Analysis

Correspondence analysis (CA) was developed by Jean-Paul Benzécri [32] and is conceptually similar to PCA, but scales the data (which should be non-negative) so that rows and columns are treated equivalently. It is traditionally applied to contingency tables. CA decomposes the chi-squared statistic associated to this table into orthogonal factors.[33] Because CA is a descriptive technique, it can be applied to tables for which the chi-squared statistic is appropriate or not. Several variants of CA are available including detrended correspondence analysis and canonical correspondence analysis. One special extension is multiple correspondence analysis, which may be seen as the counterpart of principal component analysis for categorical data [34].

2.2.10 Software / Source Code

In SAS, PROC FACTOR offers principal components analysis. We can specify the following statements with the FACTOR procedure:

PROC FACTOR <options> ;

VAR variables ;

PRIORS communalities ;

PARTIAL variables ;

FREQ variable ;

WEIGHT variable ;

BY variables ;

Source code used for creating the model will be given in the appendix.

2.3 Logistic Regression

In statistics, logistic regression or logit regression is a type of probabilistic statistical classification model.[35] It is also used to predict a binary response from a binary predictor, used for predicting the outcome of a categorical dependent variable based on one or more predictor variables or features. That is, it is used in estimating empirical values of the parameters in a qualitative response model. The probabilities describing the possible outcomes of a single trial are modeled, as a function of the explanatory (predictor) variables, using a logistic function. Frequently "logistic regression" is used

to refer specifically to the problem in which the dependent variable is binary that is, the number of available categories is two and problems with more than two categories are referred to as multinomial logistic regression or, if the multiple categories are ordered, as ordered logistic regression.

Logistic regression measures the relationship between a categorical dependent variable and one or more independent variables, which are usually continuous, by using probability scores as the predicted values of the dependent variable [36]. As such it treats the same set of problems as does probit regression using similar techniques.

2.3.1 Fields and Examples of Applications

Logistic regression is used extensively in numerous disciplines, including the medical and social science fields. For example, the Trauma and Injury Severity Score (TRISS), which is widely used to predict mortality in injured patients, was originally developed by Boyd et al. using logistic regression [37]. Logistic regression might be used to predict whether a patient has a given disease (e.g. diabetes), based on observed characteristics of the patient (age, gender, body mass index, results of various blood tests, etc.). Another example might be to predict whether an American voter will vote Democratic or Republican, based on age, income, gender, race, state of residence, votes in previous elections [38]. The technique can also be used in engineering, especially for predicting the probability of failure of a given process, system or product [39] [40]. It is also used in marketing applications such as prediction of a customer's propensity to purchase a product or cease a subscription, etc. In economics it can be used to predict the likelihood of a person's choosing to be in the labor force, and a business application would be to predict the likehood of a homeowner defaulting on a mortgage. Conditional random fields, an extension of logistic regression to sequential data, are used in natural language processing.

2.3.2 Basics of Logistic Regression

Logistic regression can be binomial or multinomial. Binomial or binary logistic regression deals with situations in which the observed outcome for a dependent variable can have only two possible types (for example, "dead" vs. "alive"). Multinomial logistic regression deals with situations where the outcome can have three or more possible types (e.g., "disease A" vs. "disease B" vs. "disease C"). In binary logistic regression,

the outcome is usually coded as "0" or "1", as this leads to the most straightforward interpretation. If a particular observed outcome for the dependent variable is the noteworthy possible outcome (referred to as a "success" or a "case") it is usually coded as "1" and the contrary outcome (referred to as a "failure" or a "noncase") as "0". Logistic regression is used to predict the odds of being a case based on the values of the independent variables (predictors). The odds are defined as the probability that a particular outcome is a case divided by the probability that it is a noncase.

Like other forms of regression analysis, logistic regression makes use of one or more predictor variables that may be either continuous or categorical data. Unlike ordinary linear regression, however, logistic regression is used for predicting binary outcomes of the dependent variable rather than continuous outcomes. Given this difference, it is necessary that logistic regression take the natural logarithm of the odds of the dependent variable being a case which can be referred to as the logit or log-odds to create a continuous criterion as a transformed version of the dependent variable. Thus the logit transformation is referred to as the link function in logistic regression although the dependent variable in logistic regression is binomial, the logit is the continuous criterion upon which linear regression is conducted [41].

The logit of success is then fit to the predictors using linear regression analysis. The predicted value of the logit is converted back into predicted odds via the inverse of the natural logarithm, namely the exponential function. Therefore, although the observed dependent variable in logistic regression is a zero-or-one variable, the logistic regression estimates the odds, as a continuous variable, that the dependent variable is a success. In some applications the odds are all that is needed. In others, a specific yes-or-no prediction is needed for whether the dependent variable is or is not a case; this categorical prediction can be based on the computed odds of a success, with predicted odds above some chosen cut-off value being translated into a prediction of a success.

2.3.3 Logistic Function, Odds Ratio and Logit

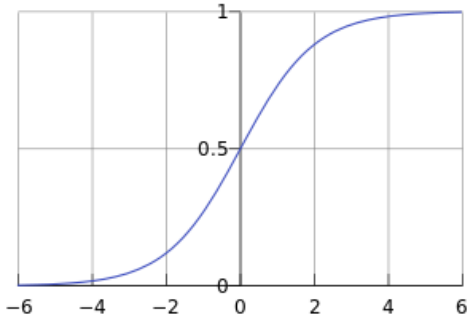
An explanation of logistic regression begins with an explanation of the logistic function, which always takes on values between zero and one:

$$F(t) = \frac{e^t}{e^t+1} = \frac{1}{1+e^{-t}} \tag{2.38}$$

and viewing t as a linear function of an explanatory variable x (or of a linear combination of explanatory variables), the logistic function can be written as:

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x+1}} = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}} \quad (2.39)$$

This will be interpreted as the probability of the dependent variable equalling a "success" or "case" rather than a failure or non-case. We also define the inverse of the logistic function, the logit:



$$g(x) = ln\frac{\pi(x)}{1-\pi(x)} = \beta_0 + \beta_1 x \quad (2.40)$$

$$\frac{\pi(x)}{1-\pi(x)} = e^{\beta_0+\beta_1 x} \quad (2.41)$$

Figure 2.2 – The logistic function, with $\beta_0 + \beta_1 x$ on the horizontal axis and $\pi(x)$ on the vertical axis

A graph of the logistic function $\pi(x)$ is shown in Figure 1. The input is the value of $\beta_0 + \beta_1 x$ and the output is $\pi(x)$. The logistic function is useful because it can take an input with any value from negative infinity to positive infinity, whereas the output $\pi(x)$ is confined to values between 0 and 1 and hence is interpretable as a probability. In the above equations, g(x) refers to the logit function of some given linear combination x of the predictors, 'ln' denotes the natural logarithm, $\pi(x)$ is the probability that the dependent variable equals a case, β_0 is the intercept from the linear regression equation (the value of the criterion when the predictor is equal to zero), $\beta_1 x$ is the regression coefficient multiplied by some value of the predictor, and base e denotes the exponential function.

The formula for $\pi(x)$ illustrates that the probability of the dependent variable equaling a case is equal to the value of the logistic function of the linear regression expression.

This is important in that it shows that the value of the linear regression expression can vary from negative to positive infinity and yet, after transformation, the resulting expression for the probability $\pi(x)$ ranges between 0 and 1. The equation for $g(x)$ illustrates that the logit (i.e., log-odds or natural logarithm of the odds) is equivalent to the linear regression expression. Likewise, the next equation illustrates that the odds of the dependent variable equaling a case is equivalent to the exponential function of the linear regression expression. This illustrates how the logit serves as a link function between the probability and the linear regression expression. Given that the logit ranges between minus infinity and infinity, it provides an adequate criterion upon which to conduct linear regression and the logit is easily converted back into the odds.

If there are multiple explanatory variables, then the above expression $\beta_0 + \beta_1 x$ can be revised to $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m$ Then when this is used in the equation relating the logged odds of a success to the values of the predictors, the linear regression will be a multiple regression with m explanators; the parameters β_j for all j = 0, 1, 2, ... m are all estimated.

2.3.4 Model Fitting

2.3.4.1 Estimation

2.3.4.1.1 Maximum Likelihood Estimation

The regression coefficients are usually estimated using maximum likelihood estimation.[42] Unlike linear regression with normally distributed residuals, it is not possible to find a closed-form expression for the coefficient values that maximizes the likelihood function, so an iterative process must be used instead, for example Newton's method. This process begins with a tentative solution, revises it slightly to see if it can be improved, and repeats this revision until improvement is minute, at which point the process is said to have converged.

In some instances the model may not reach convergence. When a model does not converge this indicates that the coefficients are not meaningful because the iterative process was unable to find appropriate solutions. A failure to converge may occur for a

number of reasons: having a large proportion of predictors to cases, multicollinearity, sparseness, or complete separation.

Having a large proportion of variables to cases results in an overly conservative Wald statistic and can lead to nonconvergence.

Multicollinearity refers to unacceptably high correlations between predictors. As multicollinearity increases, coefficients remain unbiased but standard errors increase and the likelihood of model convergence decreases. To detect multicollinearity amongst the predictors, one can conduct a linear regression analysis with the predictors of interest for the sole purpose of examining the tolerance statistic [9] used to assess whether multicollinearity is unacceptably high.

Sparseness in the data refers to having a large proportion of empty cells (cells with zero counts). Zero cell counts are particularly problematic with categorical predictors. With continuous predictors, the model can infer values for the zero cell counts, but this is not the case with categorical predictors. The reason the model will not converge with zero cell counts for categorical predictors is because the natural logarithm of zero is an undefined value, so final solutions to the model cannot be reached. To remedy this problem, researchers may collapse categories in a theoretically meaningful way or may consider adding a constant to all cells.

Another numerical problem that may lead to a lack of convergence is complete separation, which refers to the instance in which the predictors perfectly predict the criterion – all cases are accurately classified. In such instances, one should reexamine the data, as there is likely some kind of error.

Although not a precise number, as a general rule of thumb, logistic regression models require a minimum of 10 events per explaining variable.

2.3.4.1.2 Minimum Chi-Squared Estimator for Grouped Data

While individual data will have a dependent variable with a value of zero or one for every observation, with grouped data one observation is on a group of people who all share the same characteristics like demographic characteristics; in this case the researcher observes the proportion of people in the group for whom the response variable falls into one category or the other. If this proportion is neither zero nor one for any group, the minimum chi-squared estimator involves using weighted least squares to

estimate a linear model in which the dependent variable is the logit of the proportion: that is, the log of the ratio of the fraction in one group to the fraction in the other group.

2.3.4.2 Evaluating Goodness of Fit

Goodness of fit in linear regression models is generally measured using the R^2. Since this has no direct analog in logistic regression, various methods including the following can be used instead.

2.3.4.2.1 Deviance and Likelihood Ratio Tests

In linear regression analysis, one is concerned with partitioning variance via the sum of squares calculations variance in the criterion is essentially divided into variance accounted for by the predictors and residual variance. In logistic regression analysis, deviance is used in lieu of sum of squares calculations. Deviance is analogous to the sum of squares calculations in linear regression and is a measure of the lack of fit to the data in a logistic regression model. Deviance is calculated by comparing a given model with the saturated model that's a model with a theoretically perfect fit. This computation is called the likelihood-ratio test:

$$D = -2ln\frac{likelihood\ of\ the\ fitted\ model}{likelihood\ of\ the\ saturated\ model} \tag{2.42}$$

In the above equation D represents the deviance and ln represents the natural logarithm. The results of the likelihood ratio (the ratio of the fitted model to the saturated model) will produce a negative value, so the product is multiplied by negative two times its natural logarithm to produce a value with an approximate chi-squared distribution. Smaller values indicate better fit as the fitted model deviates less from the saturated model. When assessed upon a chi-square distribution, nonsignificant chi-square values indicate very little unexplained variance and thus, good model fit. Conversely, a significant chi-square value indicates that a significant amount of the variance is unexplained.

Two measures of deviance are particularly important in logistic regression: null deviance and model deviance. The null deviance represents the difference between a model with only the intercept which means no predictors and the saturated model. And,

the model deviance represents the difference between a model with at least one predictor and the saturated model. In this respect, the null model provides a baseline upon which to compare predictor models. Given that deviance is a measure of the difference between a given model and the saturated model, smaller values indicate better fit. Therefore, to assess the contribution of a predictor or set of predictors, one can subtract the model deviance from the null deviance and assess the difference on a X^2_{s-p}, chi-square distribution with degree of freedom equal to the difference in the number of parameters estimated.

Let,

$$D_{null} = -2ln\frac{likelihood\ of\ null\ model}{likelihood\ of\ saturated\ model} \quad (2.43)$$

$$D_{fitted} = -2ln\frac{likelihood\ of\ fitted\ model}{likelihood\ of\ saturated\ model}$$

Then,

$$D_{fitted} - D_{null} =$$
$$-2ln\frac{likelihood\ of\ null\ model}{likelihood\ of\ saturated\ model} - \left(-2ln\frac{likelihood\ of\ fitted\ model}{likelihood\ of\ saturated\ model}\right)$$

$$= -2\left(ln\frac{likelihood\ of\ null\ model}{likelihood\ of\ saturated\ model} - ln\frac{likelihood\ of\ fitted\ model}{likelihood\ of\ saturated\ model}\right)$$

$$= -2ln\frac{\frac{likelihood\ of\ null\ model}{likelihood\ of\ saturated\ model}}{\frac{likelihood\ of\ fitted\ model}{likelihood\ of\ saturated\ model}}$$

$$= -2ln\frac{likelihood\ of\ null\ model}{likelihood\ of\ fitted\ model} \quad (2.44)$$

If the model deviance is significantly smaller than the null deviance then one can conclude that the predictor or set of predictors significantly improved model fit. This is analogous to the F-test used in linear regression analysis to assess the significance of prediction.

2.3.4.2.2 Pseudo-R^2s

In linear regression the squared multiple correlation, R^2 is used to assess goodness of fit as it represents the proportion of variance in the criterion that is explained by the predictors. In logistic regression analysis, there is no agreed upon analogous measure, but there are several competing measures each with limitations. Three of the most commonly used indices are examined on this page beginning with the likelihood ratio R^2, R^2_L:

$$R_L^2 = \frac{D_{null} - D_{model}}{D_{null}} \tag{2.45}$$

This is the most analogous index to the squared multiple correlation in linear regression. It represents the proportional reduction in the deviance wherein the deviance is treated as a measure of variation analogous but not identical to the variance in linear regression analysis.One limitation of the likelihood ratio R^2 is that it is not monotonically related to the odds ratio, meaning that it does not necessarily increase as the odds ratio increases and does not necessarily decrease as the odds ratio decreases.

The Cox and Snell R^2 is an alternative index of goodness of fit related to the R^2 value from linear regression. The Cox and Snell index is problematic as its maximum value is .75, when the variance is at its maximum. The Nagelkerke R^2 provides a correction to the Cox and Snell R^2 so that the maximum value is equal to one. Nevertheless, the Cox and Snell and likelihood ratio R2s show greater agreement with each other than either does with the Nagelkerke R^2. Of course, this might not be the case for values exceeding .75 as the Cox and Snell index is capped at this value. The likelihood ratio R^2 is often preferred to the alternatives as it is most analogous to R2 in linear regression, is independent of the base rate and varies between 0 and 1.

A word of caution is in order when interpreting pseudo-R^2 statistics. The reason these indices of fit are referred to as pseudo R^2 is because they do not represent the proportionate reduction in error as the R^2 in linear regression does. Linear regression assumes homoscedasticity, that the error variance is the same for all values of the criterion. Logistic regression will always be heteroscedastic the error variances differ for each value of the predicted score. For each value of the predicted score there would be a different value of the proportionate reduction in error. Therefore, it is inappropriate

to think of R^2 as a proportionate reduction in error in a universal sense in logistic regression.

2.3.4.2.3 Hosmer–Lemeshow Test

The Hosmer–Lemeshow test uses a test statistic that asymptotically follows a chi^2 distribution to assess whether or not the observed event rates match expected event rates in subgroups of the model population.

2.3.4.2.4 Evaluating Binary Classification Performance

If the estimated probabilities are to be used to classify each observation of independent variable values as predicting the category that the dependent variable is found in, the various methods below for judging the model's suitability in out-of-sample forecasting can also be used on the data that were used for estimation accuracy, precision which is also called as positive predictive value, recall which is also called sensitivity, specificity and negative predictive value. In each of these evaluative methods, an aspect of the model's effectiveness in assigning instances to the correct categories is measured.

2.3.5 Coefficients

After fitting the model, it is likely that researchers will want to examine the contribution of individual predictors. To do so, they will want to examine the regression coefficients. In linear regression, the regression coefficients represent the change in the criterion for each unit change in the predictor. In logistic regression, however, the regression coefficients represent the change in the logit for each unit change in the predictor. Given that the logit is not intuitive, researchers are likely to focus on a predictor's effect on the exponential function of the regression coefficient that points the odds ratio. In linear regression, the significance of a regression coefficient is assessed by computing a t-test. In logistic regression, there are several different tests designed to assess the significance of an individual predictor, most notably the likelihood ratio test and the Wald statistic.

2.3.5.1 Likelihood Ratio Test

The likelihood-ratio test test discussed above to assess model fit is also the recommended procedure to assess the contribution of individual predictors to a given model. In the case of a single predictor model, one simply compares the deviance of the

predictor model with that of the null model on a chi-square distribution with a single degree of freedom. If the predictor model has a significantly smaller deviance, then one can conclude that there is a significant association between the predictor and the outcome. Given that some common statistical packages (e.g., SAS, SPSS) do not provide likelihood ratio test statistics, it can be more difficult to assess the contribution of individual predictors in the multiple logistic regression case. To assess the contribution of individual predictors one can enter the predictors hierarchically, comparing each new model with the previous to determine the contribution of each predictor.

2.3.5.2 Wald Statistic

Alternatively, when assessing the contribution of individual predictors in a given model, one may examine the significance of the Wald statistic. The Wald statistic, analogous to the t-test in linear regression, is used to assess the significance of coefficients. The Wald statistic is the ratio of the square of the regression coefficient to the square of the standard error of the coefficient and is asymptotically distributed as a chi-square distribution.

$$W_j = \frac{B_j^2}{SE_{B_j}^2} \tag{2.46}$$

Although several statistical packages (e.g., SPSS, SAS) report the Wald statistic to assess the contribution of individual predictors, the Wald statistic has limitations. When the regression coefficient is large, the standard error of the regression coefficient also tends to be large increasing the probability of Type-II error. The Wald statistic also tends to be biased when data are sparse.

2.3.6 Model Suitability

A way to measure a model's suitability is to assess the model against a set of data that was not used to create the model [43]. The class of techniques is called cross-validation. This holdout model assessment method is particularly valuable when data are collected in different settings (e.g., at different times or places) or when models are assumed to be generalizable.

To measure the suitability of a binary regression model, one can classify both the actual value and the predicted value of each observation as either 0 or 1. The predicted value

of an observation can be set equal to 1 if the estimated probability that the observation equals 1 is above $\frac{1}{2}$, and set equal to 0 if the estimated probability is below $\frac{1}{2}$. Here logistic regression is being used as abinary classification model. There are four possible combined classifications:

- prediction of 0 when the holdout sample has a 0 (True Negatives, the number of which is TN)
- prediction of 0 when the holdout sample has a 1 (False Negatives, the number of which is FN)
- prediction of 1 when the holdout sample has a 0 (False Positives, the number of which is FP)
- prediction of 1 when the holdout sample has a 1 (True Positives, the number of which is TP)

These classifications are used to calculate accuracy, precision (also called positive predictive value), recall (also called sensitivity), specificity and negative predictive value:

$Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$ =fraction of observations with correct predicted classification.

$Precision = PositivePredictiveValue = \frac{TP}{TP+FP}$ = fraction of observations with correct predicted classification.

$NegativePredictiveValue = \frac{TN}{TN+FN}$ =fraction of predicted negatives that are correct.

$Recall = Sensitivity = \frac{TP}{TP+FN}$ = fraction of observations that are actually 1 with a correct predicted classification.

$Specificity = \frac{TN}{TN+FP}$ = fraction of observations that are actually 0 with a correct predicted classification.

2.3.7 Software / Source Code

Proc Logistic procedure in SAS used for modelling logistic regression.

These statements are used with PROC LOGISTIC:

PROC LOGISTIC DATA=SAS-data-set < options > ;

MODEL response = independents < / options >;

BY variables;

OUTPUT <OUT=SAS-data-set>

<keyword=name ... keyword=name>

/ <ALPHA=value>;

WEIGHT variable;

2.4 Scoring Churners

Proc Score procedure in SAS used to score churn customers based on logsitic regression model.

The SCORE procedure multiplies values from two SAS data sets, one containing coefficients (for example, factor-scoring coefficients or regression coefficients) and the other containing raw data to be scored using the coefficients from the first data set. The result of this multiplication is a SAS data set containing linear combinations of the coefficients and the raw data values.

Many statistical procedures output coefficients that PROC SCORE can apply to raw data to produce scores. The new score variable is formed as a linear combination of raw data and scoring coefficients. For each observation in the raw data set, PROC SCORE multiplies the value of a variable in the raw data set by the matching scoring coefficient from the data set of scoring coefficients. This multiplication process is repeated for each variable in the VAR statement. The resulting products are then summed to produce the value of the new score variable. This entire process is repeated for each observation in the raw data set. In other words, PROC SCORE cross multiplies part of one data set with another.

The following statements are available in the SCORE procedure :

PROC SCORE DATA=SAS-data-set < options > ;

BY variables ;

ID variables ;

VAR variables ;

SECTION 3

APPLICATION and FINDINGS

In the first part we will build the customer retention model from basic shopping behavior over 8 wks. Firstly we will use Principal Components method to create factor variables by using customer spend, customer puchased quantity, customer visits, customer distinct visit week number, visit per week, spend per week and quantity per week variables.

Afterwards we will be applying logistic regression based on these variables. By the way we will obtain our customer retention model by using 8 weeks of data.

Lastly we will score our customers to predict whether they will churn or not after 13 weeks.

3.1 Customer Retention Model

Our observation period starts from week 30 of 2013 and continues through week 37. Firstly we have captured loyal customers at week 37 and by using this customer list which includes 39,901 loyal customers, we have created necessary customer metrics for each week starting from week 30 to week 37 to understand customer behaviours during 8 week period. Model development dataset defined by using loyals from end_week which will be used used to develop model parameters. After merging 8 weeks data together churn variable was added to model by checking customers' loyalty segment at week 50 (i.e. 13 weeks after obeservation period). If a loyal customer moved to non-loyal segment it is flagged as 1 othwerwise 0. Final dataset includes 39,901 customers but we will show a display of the customer table including 20 customers as a sample view:

Table 3.1 Final customer metrics table

	dib_cust_id	dib_cust_code	spend	quantity	visits	weeks	VPW	SPW	QPW	churn
1	3231105000003...	000000000000069...	611.27	94	3	3	1	203.75666667	31.333333333	0
2	3231105000026...	000000000000053...	294.69	42	6	3	2	98.23	14	0
3	3231105000028...	000000000000009...	211.94	52	7	4	1.75	52.985	13	0
4	3231105000051...	000000000000090...	413.64	80	7	5	1.4	82.728	16	0
5	3231105000052...	000000000000082...	249.11	91	19	8	2.375	31.13875	11.375	0
6	3231105000054...	000000000000025...	209.8	46	7	3	2.333333...	69.933333333	15.333333333	0
7	3231105000065...	000000000000025...	276.58	68	5	4	1.25	69.145	17	1
8	3231105000067...	000000000000023...	419.55	50	7	4	1.75	104.8875	12.5	1
9	3231105000068...	000000000000005...	173.11	57	3	3	1	57.703333333	19	0
10	3231105000094...	000000000000088...	324.09	57	5	4	1.25	81.0225	14.25	1
11	3231105000096...	000000000000080...	336.59	95	3	3	1	112.19666667	31.666666667	0
12	3231105000098...	000000000000064...	324.22	69	3	3	1	108.07333333	23	0
13	3231105000102...	000000000000086...	258.16	52	3	3	1	86.053333333	17.333333333	1
14	3231105000117...	000000000000088...	197.41	51	5	4	1.25	49.3525	12.75	0
15	3231105000124...	000000000000015...	584.42	30	8	7	1.142857...	83.488571429	4.2857142857	0
16	3231105000125...	000000000000032...	261.94	33	5	4	1.25	65.485	8.25	1
17	3231105000128...	000000000000033...	1508.15	200	39	8	4.875	188.51875	25	0
18	3231105000132...	000000000000039...	446.11	111	10	6	1.666666...	74.351666667	18.5	1
19	3231105000133...	000000000000071...	323.2	74	5	4	1.25	80.8	18.5	0
20	3231105000142...	000000000000045...	813.51	190	6	5	1.2	162.702	38	0

Then principal compinents were built based on varimax rotation. Varimax rotation is a change of coordinates used in principal component analysis and factor analysis that maximizes the sum of the variances of the squared loadings (squared correlations between variables and factors). Intuitively, this is achieved if, (a) any given variable has a high loading on a single factor but near-zero loadings on the remaining factors and if (b) any given factor is constituted by only a few variables with very high loadings on this factor while the remaining variables have near-zero loadings on this factor. If these conditions hold, the factor loading matrix is said to have "simple structure," and varimax rotation brings the loading matrix closer to such simple structure (as much as the data allow). From the perspective of individuals measured on the variables, varimax seeks a basis that most economically represents each individual—that is, each individual can be well described by a linear combination of only a few basis functions..

One way of expressing the varimax criterion formally is this:

$$R_{VARIMAX} = \arg max_R \left(\sum_{j=1}^{k} \sum_{i=1}^{p} (\Lambda R)_{ij}^4 - \frac{\gamma}{p} \sum_{j=1}^{k} \left(\sum_{i=1}^{p} (\Lambda R)_{ij}^2 \right)^2 \right) \quad (3.1)$$

where $\gamma = 1$ for VARIMAX.

Table 3.2 Customer metrics table including factor loadings

	dib_cust_id	spend	quantity	visits	weeks	VPW	SPW	QPW	churn	Factor1	Factor2
1	323110500000...	611.27	94	3	3	1	203.75666...	31.33333...	0	0.6527175...	-1.284217284
2	323110500002...	294.69	42	6	3	2	98.23	14	0	0.0127584...	-0.841565934
3	323110500002...	211.94	52	7	4	1.75	52.985	13	0	-0.309588...	-0.448792841
4	323110500005...	413.64	80	7	5	1.4	82.728	16	0	-0.243709...	-0.242869894
5	323110500005...	249.11	91	19	8	2.375	31.13875	11.375	0	-0.886156...	1.2601870865
6	323110500005...	209.8	46	7	3	2.333333...	69.933333...	15.33333...	0	-0.077468...	-0.725346798
7	323110500006...	276.58	68	5	4	1.25	69.145	17	1	-0.199214...	-0.606538597
8	323110500006...	419.55	50	7	4	1.75	104.8875	12.5	1	-0.076879...	-0.543114696
9	323110500006...	173.11	57	3	3	1	57.703333...	19	0	-0.124260...	-0.991639326
10	323110500009...	324.09	57	5	4	1.25	81.0225	14.25	1	-0.180871...	-0.621039789
11	323110500009...	336.59	95	3	3	1	112.19666...	31.66666...	0	0.2724628...	-1.122478263
12	323110500009...	324.22	69	3	3	1	108.07333...	23	0	0.1403421...	-1.091860673
13	323110500010...	258.16	52	3	3	1	86.053333...	17.33333...	1	-0.027230...	-1.037516959
14	323110500011...	197.41	51	5	4	1.25	49.3525	12.75	0	-0.346776...	-0.558634684
15	323110500012...	584.42	30	8	7	1.142857...	83.488571...	4.285714...	0	-0.602261...	0.4085887791
16	323110500012...	261.94	33	5	4	1.25	65.485	8.25	1	-0.331992...	-0.576275126
17	323110500012...	1508.15	200	39	8	4.875	188.51875	25	0	0.3763810...	1.7353809673
18	323110500013...	446.11	111	10	6	1.666666...	74.351666...	18.5	1	-0.345296...	0.1940951993
19	323110500013...	323.2	74	5	4	1.25	80.8	18.5	0	-0.125602...	-0.632052433
20	323110500014...	813.51	190	6	5	1.2	162.702	38	0	0.4421491...	-0.50680056

Table 3.3 Descriptive statistics of variables

Input Data Type	Raw Data
Number of Records Read	39901
Number of Records Used	39901
N for Significance Tests	39901

Means and Standard Deviations from 39901 Observations		
Variable	Mean	Std Dev
spend	562.75854	747.89079
visits	12.08659	16.57172
weeks	5.29578	1.67514
SPW	109.47755	123.62447
QPW	24.75045	23.38931
VPW	2.10533	2.54556

Table 3.4 Eigenvalues of correlation matrix

Initial Factor Method: Principal Components

Prior Communality Estimates: ONE

Eigenvalues of the Correlation Matrix: Total = 6 Average = 1				
	Eigenvalue	Difference	Proportion	Cumulative
1	3.63484959	2.27107163	0.6058	0.6058
2	1.36377797	0.64538506	0.2273	0.8331
3	0.71839291	0.50728455	0.1197	0.9528
4	0.21110835	0.15891227	0.0352	0.9880
5	0.05219608	0.03252098	0.0087	0.9967
6	0.01967510		0.0033	1.0000

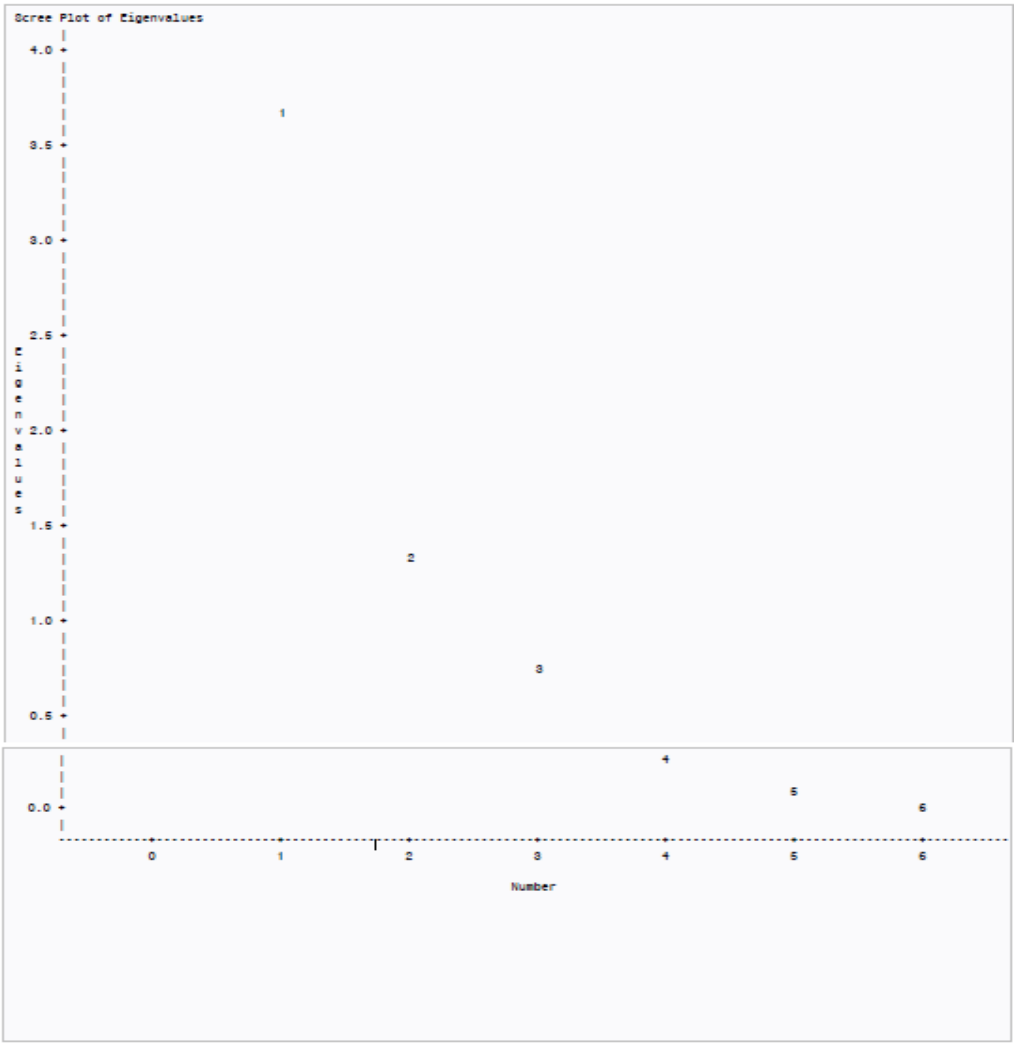


Figure 3.1 Scree plot of eigenvalues

2 factors will be retained by the NFACTOR criterion.

Table 3.5 Factor Patterns

Factor Pattern			
	Factor1	Factor2	
spend	89 *	-26	
visits	82 *	50	*
weeks	22	79	*
SPW	83 *	-47	*
QPW	88 *	-28	
VPW	82 *	35	

Printed values are multiplied by 100 and rounded to the nearest integer. Values greater than 0.4 are flagged by an '*'.

Table 3.6 Variances explained by each factor

Variance Explained by Each Factor	
Factor1	Factor2
3.6348496	1.3637780

Table 3.7 Final communality estimates

Final Communality Estimates: Total = 4.998628					
spend	visits	weeks	SPW	QPW	VPW
0.85724754	0.91397098	0.67262546	0.91056070	0.84533295	0.79888993

Table 3.8 Rotated factor patterns according to varimax rotation

The FACTOR Procedure
Rotation Method: Varimax

Orthogonal Transformation Matrix		
	1	2
1	0.87587	0.48254
2	-0.48254	0.87587

Rotated Factor Pattern			
	Factor1	Factor2	
spend	90 *	20	
visits	48 *	83	*
weeks	-19	80	*
SPW	95 *	-2	
QPW	90 *	18	
VPW	55 *	70	*

Printed values are multiplied by 100 and rounded to the nearest integer. Values greater than 0.4 are flagged by an '*'.

Variance Explained by Each Factor	
Factor1	Factor2
3.1060383	1.8925893

Final Communality Estimates: Total = 4.998628					
spend	visits	weeks	SPW	QPW	VPW
0.85724754	0.91397098	0.67262546	0.91056070	0.84533295	0.79888993

Table 3.9 Scoring coefficients estimated by regression

The FACTOR Procedure
Rotation Method: Varimax

Scoring Coefficients Estimated by Regression

Squared Multiple Correlations of the Variables with Each Factor	
Factor1	Factor2
1.0000000	1.0000000

Standardized Scoring Coefficients		
	Factor1	Factor2
spend	0.30655	-0.05005
visits	0.02103	0.42750
weeks	-0.22565	0.53650
SPW	0.36737	-0.19477
QPW	0.30981	-0.06286
VPW	0.07427	0.33405

Based on the final table shown at table 3.2 including factor loadings, logistic regression model applied.

Table 3.10 Model information

The LOGISTIC Procedure

Model Information	
Data Set	WORK.PC
Response Variable	churn
Number of Response Levels	2
Weight Variable	splitwgt
Model	binary logit
Optimization Technique	Fisher's scoring

Number of Observations Read	39901
Number of Observations Used	28011
Sum of Weights Read	28011
Sum of Weights Used	28011

Response Profile			
Ordered Value	churn	Total Frequency	Total Weight
1	0	24590	24590.000
2	1	3421	3421.000

Probability modeled is churn=1.

Note: 11890 observations having nonpositive frequencies or weights were excluded since they do not contribute to the analysis.

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	20794.490	20273.116
SC	20802.731	20297.837
-2 Log L	20792.490	20267.116

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	525.3747	2	<.0001
Score	492.7765	2	<.0001
Wald	457.9552	2	<.0001

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-2.0454	0.0195	10975.9392	<.0001
Factor1	1	0.1376	0.0176	61.4155	<.0001
Factor2	1	-0.4559	0.0239	362.7911	<.0001

Table 3.11 Odd ratio estimates

The LOGISTIC Procedure

Note: 11890 observations having nonpositive frequencies or weights were excluded since they do not contribute to the analysis.

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Factor1	1.147	1.109	1.188
Factor2	0.634	0.605	0.664

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	63.2	Somers' D	0.278
Percent Discordant	35.4	Gamma	0.282
Percent Tied	1.4	Tau-a	0.060
Pairs	84122390	c	0.639

Table 3.12 Development data churn scores

DEVELOPMENT DATA			
Decile	Candidates	Predicted Churn Rate	Actual churn Rate
1	2,801	19.7%	21.5%
2	2,801	16.5%	20.7%
3	2,801	15.2%	14.5%
4	2,801	14.2%	15.3%
5	2,801	13.2%	12.5%
6	2,801	11.8%	9.6%
7	2,801	10.2%	8.1%
8	2,801	8.6%	6.5%
9	2,801	7.4%	6.6%
10	2,802	5.4%	6.7%
Total	28,011	12.2%	12.2%

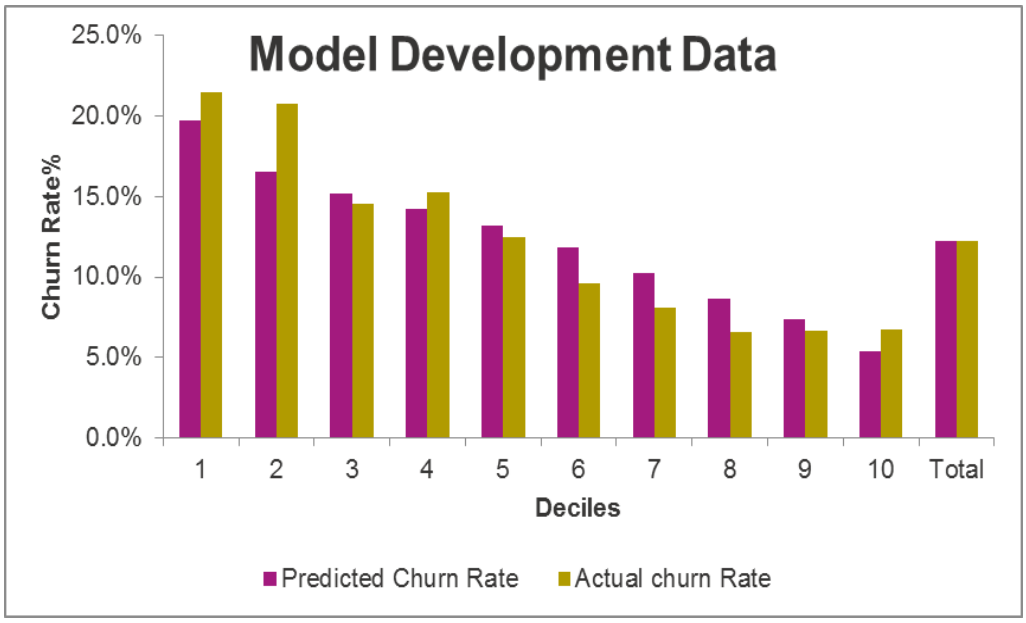


Figure 3.2 Development data predicted vs actual churn rates

According to table 3.12 and Figure 3.2 we see predicted churn rates and actual churn rates are very close to each other that shows the accuracy of the model.

Table 3.13 Validation data churn scores

VALIDATION DATA			
Decile	Candidates	Predicted Churn Rate	Actual churn Rate
1	1,188	19.8%	18.8%
2	1,189	16.5%	18.3%
3	1,189	15.2%	15.8%
4	1,189	14.2%	17.0%
5	1,189	13.1%	11.6%
6	1,189	11.7%	8.6%
7	1,189	10.1%	7.8%
8	1,189	8.5%	6.1%
9	1,189	7.3%	5.6%
10	1,190	5.4%	7.3%
Total	11,890	12.2%	11.7%

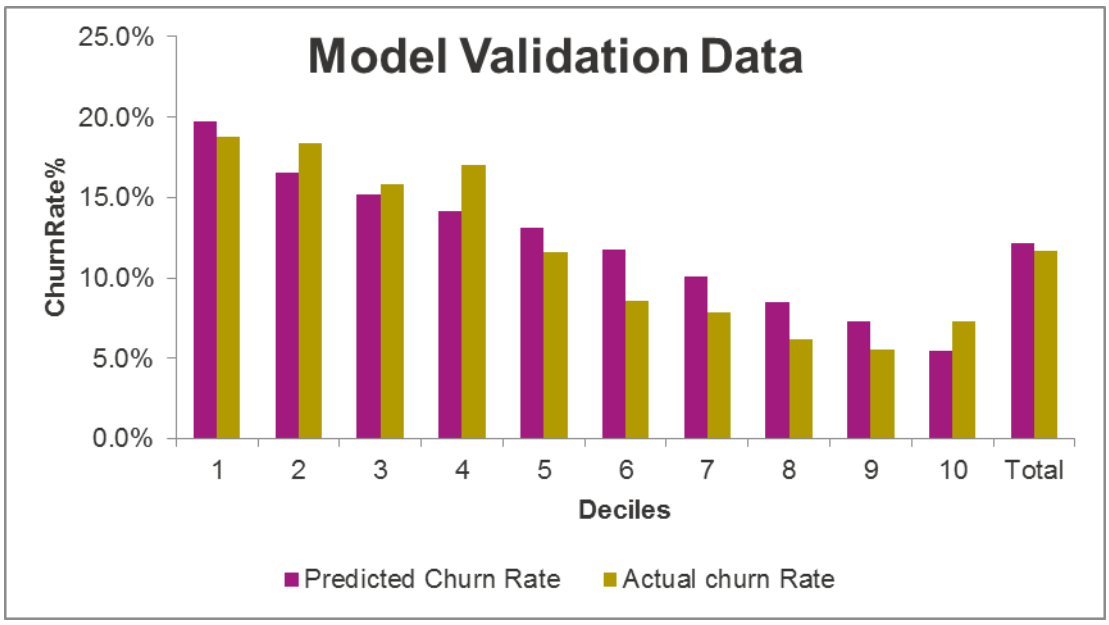


Figure 3.3 Validation data predicted vs actual churn rates

According to table 3.13 and Figure 3.3 we see predicted churn rates and actual churn rates are very close to each other in validation data like in development data that demonstrates the accuracy of the model.

Table 3.14 Logistic model information

The LOGISTIC Procedure

Model Information	
Data Set	WORK.PC
Response Variable	churn
Number of Response Levels	2
Weight Variable	splitwgt
Model	binary logit
Optimization Technique	Fisher's scoring

Number of Observations Read	39901
Number of Observations Used	28011
Sum of Weights Read	28011
Sum of Weights Used	28011

Response Profile			
Ordered Value	churn	Total Frequency	Total Weight
1	0	24590	24590.000
2	1	3421	3421.000

Probability modeled is churn=1.

11890 observations having nonpositive frequencies or weights were excluded since they do not contribute to the analysis.

Table 3.15 Model fit statistics

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	20794.490	20273.116
SC	20802.731	20297.837
-2 Log L	20792.490	20267.116

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	525.3747	2	<.0001
Score	492.7765	2	<.0001
Wald	457.9552	2	<.0001

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-2.0454	0.0195	10975.9392	<.0001
Factor1	1	0.1376	0.0176	61.4155	<.0001
Factor2	1	-0.4559	0.0239	362.7911	<.0001

From Table 3.15 we understand that Intercept, Factor1 and Factor2 parameters are statistically significant. Large Wald chi-square values also demonstrate this.

Table 3.16 Odd ratio estimates of factors

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Factor1	1.147	1.109	1.188
Factor2	0.634	0.605	0.664

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	63.2	Somers' D	0.278
Percent Discordant	35.4	Gamma	0.282
Percent Tied	1.4	Tau-a	0.060
Pairs	84122390	c	0.639

Table 3.17 Final data set including predictions

	dib_cust_id	churn	Factor1	Factor2	pred	mod_dec
1	3231419000022...	1	40.497421068	-11.55325206	0.9998480661	1
2	3231402000025...	1	12.720562346	-4.07172355	0.8264288153	1
3	3231338000032...	0	11.082748328	-2.81244524	0.6816103373	1
4	3231341000212...	0	6.9090301099	-3.398969138	0.6116932456	1
5	3231358000103...	0	12.760531667	-1.617892748	0.6100265069	1
6	3231342000263...	0	8.4634828831	-2.904046386	0.6088889386	1
7	3231345000071...	0	15.511424842	-0.579341681	0.5871975547	1
8	3231416000016...	1	6.6660946748	-2.99747835	0.5592194944	1
9	3231338000025...	0	10.824902006	-1.359336967	0.5158201484	1
10	3231402000020...	1	6.7685189501	-2.458136939	0.5015551304	1
11	3231394000019...	1	4.8988101432	-2.232289867	0.412431421	1
12	3231404000017...	1	3.6743783389	-2.389908863	0.3892422118	1
13	3231418000003...	0	13.538703731	0.6395898543	0.3835215348	1
14	3231339000485...	0	3.3140726982	-2.442422034	0.3831682328	1
15	3231350000032...	1	3.3680461801	-2.351814262	0.3751930173	1
16	3231338000029...	0	19.484007474	2.5155127851	0.3747311524	1
17	3231340000150...	0	4.5250224762	-1.927027616	0.3671422833	1
18	3231374000068...	1	3.4845610686	-2.172684442	0.359938109	1
19	3231339000289...	0	4.7568416619	-1.694894638	0.3501390734	1
20	3231343000233...	0	4.081691411	-1.77325623	0.3372491997	1

We have grouped customers into 10 deciles. 1st decile group includes highest churn risk customers while 10th decile group includes lowest churn risk customers.

3.2 Customer Churn Risk Scoring

In the second phase of the model we try to target loyal customers most likely to leave the market by scoring them based on the model built in the first phase described above.

We have selected 10,000 loyal customers from week 18 of 2013 and scored these customers based on the model which enables us calculate factor loadings of variables and then scored them again to calculate their churn risks based on the factor values. In the final dataset we have customer codes, predicted churn score and risk deciles (1=riskiest to 10=least risky). Churn Score source codes are given in Appendix B.

Table 3.18 Scored customers based on the model

	dib_cust_id	churn_score_13wks	mod_dec
1	3231403000030...	0.9128266118	1
2	3231340000282...	0.5895184822	1
3	3231404000040...	0.5884636692	1
4	3231339000146...	0.5653275402	1
5	3231405000003...	0.5499994394	1
6	3231355000049...	0.5096174142	1
7	3231342000299...	0.4738724836	1
8	3231341000623...	0.4724731992	1
9	3231365000057...	0.4539365387	1
10	3231410000032...	0.4331277822	1

This final table includes 2691 customers. Risk summary belonging to these customers are shown below :

Table 3.19 Risk deciles of scored customers

Risk Deciles	Average Churn Score
1	26%
2	22%
3	21%
4	20%
5	20%
6	18%
7	17%
8	15%
9	12%
10	8%
Grand Total	**18%**

According to this information, the priority of the retailer should be taking precautions to retent customers at risk group 1,2 and 3 since they are at least 21% likely to leave the retailer in next 13 weeks.

SECTION 4

RESULTS and DISCUSSION

Managing customer churn is of great concern to global retail companies and it is becoming a more serious problem as the market matures. The annual churn rate ranges from 20% to 40% in most of the global retail companies. Customer churn adversely affects these companies because they stand to lose a great deal of price premium, decreasing profit levels and a possible loss of referrals from continuing service customers. Furthermore, the cost of acquiring a new customer can substantially exceed the cost of retaining an existing customer.

In a highly competitive and retail market, a defensive marketing strategy is becoming more important. Instead of attempting to entice new customers or lure subscribers away from competitors, defensive marketing is concerned with reducing customer exit and brand switching. It is estimated that, with an increase in customer retention rates of just 5%, the average net present value of a customer increases by 35% for software companies and 95% for advertising agencies. Therefore, in order to be successful in the maturing market, the strategic focus of a company ought to shift from acquiring customers to retaining customers by reducing customer churn.

This study investigated factors leading to customer churn using a sample of 39,901 actual customer transactions data. In addition, the mediating effects of customer status between churn determinants and customer churn were analyzed. The following section summarizes the result, discusses implications and suggests areas for further study.

First, this study developed and tested a customer churn model based on a large number of transaction data. This actual data-based approach addressed the managerial problems that may arise from the discrepancy between customers' perception or intention and

their actual behavior in the market. For example previous research based on customer survey responses suggests that for membership card program subscribers, the negative effects of dissatisfaction with the service provider are adjusted, thus they remain loyal customers. However, this study using a company-internal database found that the membership card program subscribers, in fact, are more likely to churn. Customers known as loyal selected at the beginning and their shopping behaviours analysed for 8 weeks. Then their loyalty status was checked at the following 13 weeks over this observation period to understand whether these customers were actually churned or not. If they moved to a non-loyal segment then their churn status flagged as 1 otherwise 0. And in the final customer metrics table summarizes all the variables related to purchase behaviour over first 8 weeks of observation period and churn status after 13 weeks. These variables are customer ID, spend over 8 weeks, total quantity puchased over 8 weeks, visits over 8 weeks, number of distinct weeks shopped over 8 weeks, visits per week, spend per week, quantity per week and churn status.

Secondly, the variables explained above are reduced to 2 factors by using principal components method. Based on the principal components analysis spend, spend per week and quantity per week variables loaded on factor 1, while visits, distinct weeks and visits per week loaded on factor 2. By this way the last shape of the data model obtained before applying logistic regression model.

Thirdly, logistic regression model applied on the final dataset including 2 factors and 1 response variable which is churn flag. Likelihood estimates for Interception, Factor1 and Factor 2 was -2.0454 ,0.1376 and -0.4559 respectively. Then pred variable showing the probability of churn calculated by the following formula: pred=**1**/(**1**+exp(-churn). Afterwards customers are sorted according to these "pred" (i.e. prediction) values and grouped into 10 deciles for ease of interpretation.1st decile contained riskiest group and 10th decile contained the least riskiest group.

Fourthly, a loyal customer group selected in a closer time frame for scoring their churn risks in future based on the model establised. Customers in decile 1 had 26% of churn risk while customers in decile 10 had 8% of churn risk on average.

The study helps retailer targeting their loyal customers most likely to leave in close future. By this way retailer can take actions beforehand to retain these customers. An

important retention activity might be a targeted campaign activity for the customers in highest risk groups.

By using the customer churn model, a retention activity planned in the market to prevent these risky customers be churn in future. You can see the results below :

Table 4.1 Results of targeted campaign activity for retaining loyals at churn risk

Risk	churn rates 13 wks				Coupon offer	Spend over redemption period			SCR
Decile	PREDICTED	CTRL	TEST	DELTA	redemption	CTRL	TEST	DELTA	
1	10.6%	13.5%	12.9%	-0.6%	11.92%	523.12	539.87	16.75	6.7
2	8.3%	7.8%	8.3%	0.5%	14.39%	609.42	610.42	1.00	0.34
3	6.7%	5.5%	5.2%	-0.2%	16.97%	706.94	722.50	15.56	4.64
4	5.3%	3.8%	3.2%	-0.6%	20.00%	773.80	781.75	7.95	2.07
5	4.4%	2.6%	2.4%	-0.2%	21.38%	792.70	789.90	-2.79	-0.7
6	3.7%	2.5%	2.0%	-0.4%	22.57%	823.53	825.81	2.28	0.55

Customers most at-risk redeemed the offer at the lowest rates but still showed the highest spend uplift .Virtually all groups showed lift in churn reduction.

Two important learnings came out of this retention activity:

- Careful timing of the time-banded trade drivers are important as to not interfere with standard Statement offer (ie. don't make a trade driver valid at the same time statement lands as it's likely a wasted give-away.)
- Push the at-risk customer to spend to what a true Loyal would spend. Our trade-driver offer spend threshold should have been appropriate for the time-banded period and the targeted spend (ie. Consider a weekly spend target). It is thought that we could have generated significantly higher uplift with more stretch offers.

Despite this analysis, there are some areas that warrant further study. First, data for some variables, such as account tenure (also called customer duration) and each subscriber's age, were not available; and customers' perceived values on service satisfaction were not included in the data either. Therefore, a better model can be developed by including these variables. In particular, the account tenure will be a very important variable explaining customer churn.

REFERENCES

[1] Glady, N., Baesens, B., & Croux, C. (2009). Modeling churn using customer lifetime value. European Journal of Operational Research, 197, 402–411.

[2] Gustafsson, A., Johnson, M. D., & Roos, I. (2005). The effects of customer satisfaction,relationship commitment dimensions, and triggers on customer retention.Journal of Marketing, 69(4), 210–218

[3] Roberts, J. H. (2000). Developing new rules for new markets. Journal of the Academy of Marketing Science, 28(1), 31–44.

[4] Dixon, M. (1999). 39 Experts predict the future. America's Community Banker, 8(7),20–31.

[5] Reichheld, F. F., & Sasser, W. E. Jr., (1990). Zero defections: Quality comes to service.Harvard Business Review, 68(5), 105–111.

[6] Van den Poel, D., & Larivière, B. (2004). Customer attrition analysis for financial services using proportional hazard models. European Journal of Operational Research, 157(1), 196–217.

[7] Neslin, S. A., Gupta, S., Kamakura, W., Lu, J., & Mason, C. (2006). Defection detection: Improving predictive accuracy of customer churn models. Journal of Marketing Research, 43(2), 204–211.

[8] Pearson, K. (1901). "On Lines and Planes of Closest Fit to Systems of Points in Space" (PDF). Philosophical Magazine 2 (11): 559–572.

[9] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, 24, 417-441, and 498-520.

[10] Abdi. H., & Williams, L.J. (2010). "Principal component analysis.". Wiley Interdisciplinary Reviews: Computational Statistics, 2: 433–459

[11] Shaw P.J.A. (2003) Multivariate statistics for the Environmental Sciences, Hodder-Arnold

[12] Jolliffe I.T. Principal Component Analysis, Series: Springer Series in Statistics, 2nd ed., Springer, NY, 2002, XXIX, 487 p. 28 illus

[13] A. A. Miranda, Y. A. Le Borgne, and G. Bontempi. New Routes from Minimal Approximation Error to Principal Components, Volume 27, Number 3 / June, 2008, Neural Processing Letters, Springer

[14] Fukunaga, Keinosuke (1990). Introduction to Statistical Pattern Recognition

[15] Gharib Shirangi, M., History matching production data and uncertainty assessment with a truncated SVD parameterization algorithm, Journal of Petroleum Science and Engineering

[16] Jolliffe, I. T. (2002). Principal Component Analysis, second edition Springer-Verlag

[17] Jonathon Shlens, A Tutorial on Principal Component Analysis

[18] Geiger, Bernhard; Kubin, Gernot (Sep 2012). "Relative Information Loss in the PCA". Proc. IEEE Information Theory Workshop: 562–566.

[19] Linsker, Ralph (March 1988). "Self-organization in a perceptual network". IEEE Computer 21 (3): 105–117.

[20] Deco & Obradovic (1996). An Information-Theoretic Approach to Neural Computing. New York, NY: Springer.

[21] Plumbley, Mark (1991). Information theory and unsupervised neural networks.Tech Note

[22] Geiger, Bernhard; Kubin, Gernot (January 2013). "Signal Enhancement as Minimization of Relevant Information Loss"

[23] Matlab documentation

[24] MATLAB PCA-based Face recognition software

[25] Eigenvalues function Mathematica documentation

[26] Geladi, Paul; Kowalski, Bruce (1986). "Partial Least Squares Regression:A Tutorial". Analytica Chimica Acta 185: 1–17

[27] Kramer,R.,(1998) Chemometric Techniques for Quantitative Analysis (CRC Press, New York).

[28] M. Andrecut. Parallel GPU Implementation of Iterative PCA Algorithms. Journal of Computational Biology, 16(11), Nov. 2009

[29] H. Zha, C. Ding, M. Gu, X. He and H.D. Simon. "Spectral Relaxation for K-means Clustering", http://ranger.uta.edu/~chqding/papers/Zha-Kmeans.pdf, Neural Information Processing Systems vol.14 (NIPS 2001). pp. 1057–1064, Vancouver, Canada. Dec. 2001

[30] C. Ding and X. He. "K-means Clustering via Principal Component Analysis". Proc. of Int'l Conf. Machine Learning (ICML 2004), pp 225–232. July 2004

[31] Timothy A. Brown. Confirmatory Factor Analysis for Applied Research Methodology in the social sciences. Guilford Press, 2006

[32] Benzécri, J.-P. (1973). L'Analyse des Données. Volume II. L'Analyse des Correspondances. Paris, France: Dunod.

[33] Greenacre, Michael (1983). Theory and Applications of Correspondence Analysis. London: Academic Press

[34] Le Roux, Brigitte and Henry Rouanet (2004). Geometric Data Analysis, From Correspondence Analysis to Structured Data Analysis. Dordrecht: Kluwer

[35] Christopher M. Bishop (2006). Pattern Recognition and Machine Learning. Springer. p. 205. "In the terminology of statistics, this model is known as

logistic regression, although it should be emphasized that this is a model for classification rather than regression

[36] Clinical Research for Surgeons. Mohit Bhandari,Anders Joensson. page 293

[37] Boyd, C. R.; Tolson, M. A.; Copes, W. S. (1987). "Evaluating trauma care: The TRISS method. Trauma Score and the Injury Severity Score". The Journal of trauma 27 (4): 370–378

[38] Harrell, Frank E. (2001). Regression Modeling Strategies. Springer-Verlag. ISBN 0-387-95232-2

[39] M. Strano; B.M. Colosimo (2006). "Logistic regression analysis for experimental determination of forming limit diagrams". International Journal of Machine Tools and Manufacture 46 (6)

[40] Palei, S. K.; Das, S. K. (2009). "Logistic regression model for prediction of roof fall risks in bord and pillar workings in coal mines: An approach". Safety Science 47: 88

[41] Hosmer, David W.; Lemeshow, Stanley (2000). Applied Logistic Regression (2nd ed.). Wiley

[42] Menard, Scott W. (2002). Applied Logistic Regression (2nd ed.)

[43] Jonathan Mark and Michael A. Goldberg (2001). Multiple Regression Analysis and Mass Assessment: A Review of the Issues. The Appraisal Journal, Jan. pp. 89–109

[44] Wikipedia , Principal component analysis, http://en.wikipedia.org/wiki/Principal_component_analysis, 07 February 2014

[45] Wikipedia , Logistic Regression, http://en.wikipedia.org/wiki/Logistic_regression, 07 February 2014

APPENDIX A

CUSTOMER RETENTION MODEL SOURCE CODE

```
/* this code attempts to build a simple customer retention model from basic shopping
behavior over 8 wks */


**********************************************;
***** Required Inputs for Macro Variables *****;
**********************************************;

%put
NOTE:
PROGRAM LAST RUN ON: %sysfunc(date(),worddate.)
AT: %sysfunc(time(),time8.).;
footnote;
options mprint = on
nonotes nomprint nosource;
%let
country   = turkey ;
%include
"/dhcommon/DHLIB/profiles/ti_&country._lev1_autoexec.sas" / nosource2;
%include
"/ti_&country./Lev1/analysis/tesco/activity_modules/auto_process/clubcard_macros/inc
lude_all_clubcard_macros.sas";

* Set your root directory (you will save your final model paramters here);
libname loc
"/ti_turkey/Lev1/analysis/tesco/activity_modules/adhoc/customer_retention/sasdata/";

* Define model development dataset (Loyals from end_week will be used used to
develop model parameters);
%let st_week=201230;  * first week of 8-wk observation window;
%let end_week=201237; * last week of 8-wk observation window;

%let churn_week=201250;* week to capture churn (13-wks post end_week);

**********************************************;
*************** End of Inputs ****************;
```

```
*************************************************;


*************************************************;
** Part 1: bring in necessary customer metrics;
*************************************************;

* Take all your Loyals into the model (no reason to sample.  process does not take much
longer. );
data loc.loyals_&end_week. (keep=dib_cust_code dib_cust_id);
set dib_pdl.dib_customer_metrics_&end_week.;
where dib_shabits in ('VL','PR');
run;


* defining time loop (8 wks of data);
data time_loop (keep=dib_time_code);
set dib_pdl.dib_time;
where dib_time_code>="&st_week." and dib_time_code<="&end_week.";
run;

data time_loop;
set time_loop;
if _n_=1 then n=1;
else n+1;
run;

/* setting max loop var */

proc sql;
select max(N) into: max_loop from time_loop;
quit;

/*  Starting loop */

%MACRO time_loop;
%DO i=1 %TO %eval(&max_loop);

proc sql;
select dib_time_code into: week_num
from time_loop
where N=&i;
quit;

proc sql;
create table customer_metrics_&i as
select *,
&week_num as tesco_week
from dib_pdl.dib_cust_spend_&week_num.
where dib_cust_id in (select dib_cust_id from loc.loyals_&end_week.);
```

```
quit;

%end;
%MEND;
%time_loop;

* stacking;

%MACRO stacker;
data customer_metrics;
set
%DO i=1 %TO %eval(&max_loop);
customer_metrics_&i
%END;
;
run;
%MEND;
%stacker;

* creating 8 week customer metrics;
proc sql;
create table customer_metrics as
select dib_cust_id,
dib_cust_code,
sum(dib_cust_spend) as spend,
sum(dib_cust_quantity) as quantity,
sum(dib_cust_visits) as visits,
count(distinct tesco_week) as weeks
from customer_metrics
group by 1,2;
quit;

* creating additional variables;
data customer_metrics;
set customer_metrics;
VPW=visits/weeks;
SPW=spend/weeks;
QPW=quantity/weeks;
run;


* flagging for churn;
proc sql;
create table customer_metrics as
select A.*,
case when b.dib_shabits in ('GO','LP') then 1 else 0 end as churn
from customer_metrics a, dib_pdl.dib_customer_metrics_&churn_week. b
where a.dib_cust_id=b.dib_cust_id;
quit;
```

```
* building principal components;
PROC FACTOR DATA=customer_metrics
SIMPLE
METHOD=PRIN
PRIORS=ONE
NFACT=2
ROTATE=VARIMAX
score
SCREE
ROUND
FLAG=0.4
OUT=PC
OUTSTAT=loc.PC_&end_week.;
VAR spend visits weeks
SPW
QPW
VPW;
run;


*****************************************************;
***** BUILDING MODEL AND VALIDATING MODEL ****;
*****************************************************;
* creating validation dataset;
data PC;
set PC;
digit=substr(left(reverse(dib_cust_code)),1,1);
if digit in ('0','1','2') then splitwgt=.;
else splitwgt=1;
run;


* need record var for steps later ;
data PC;
set PC;
record=1;
run;



* LIST variables;
%LET var_list =
factor1
factor2
;


/*  Fitting Logistic Regression Model */
```

```
proc logistic data=PC;
WEIGHT SPLITWGT;
model churn (event='1') =
&var_list
/ maxiter=200;
output out=churn_output pred=pred;
run;

proc sort data=churn_output;
by DESCENDING pred;
run;

* decile analysis for DEV and VAL samples;

%MACRO dev_and_val(wgt, ds);

proc univariate data=churn_output
(where=(splitwgt=&wgt.)) noprint;
var pred churn;
output out=preddata sumwgt=sumwgt;
run;

data churn_output_&ds.;
set churn_output (where=(splitwgt =&wgt.));
if (_n_=1) then set preddata;
retain sumwgt;
number+1;
if number < .10*sumwgt then mod_dec=1;
else if number < .20*sumwgt then mod_dec=2;
else if number < .30*sumwgt then mod_dec=3;
else if number < .40*sumwgt then mod_dec=4;
else if number < .50*sumwgt then mod_dec=5;
else if number < .60*sumwgt then mod_dec=6;
else if number < .70*sumwgt then mod_dec=7;
else if number < .80*sumwgt then mod_dec=8;
else if number < .90*sumwgt then mod_dec=9;
else mod_dec=10;
run;

title1 "DECILE ANALYSIS";
title2 "&ds - Score Selection";

proc tabulate data=churn_output_&ds;
class mod_dec;
var record pred churn;
table mod_dec='Decile' all='Total',
record='Candidates'*sum=' '*f=comma10.
pred='Predicted Churn Rate'*(mean=' '*f=11.5)
churn='Actual churn Rate'*(mean=' '*f=11.5)
/ rts=9 row=float;
```

```
run;

%MEND;

%dev_and_val(1 , DEV);
%dev_and_val(. , VAL);

* creating the final parameters;
proc logistic data=PC outest=loc.RegOut_&end_week.;
WEIGHT SPLITWGT;
model churn (event='1') =
&var_list
/ maxiter=200;
/*output out=churn_output pred=pred; */

run;
```

APPENDIX B

CHURN SCORING MODEL SOURCE CODE

```
/* this code attempts to build a simple customer retention model from basic shopping
behavior over 8 wks */


**********************************************;
***** Required Inputs for Macro Variables *****;
**********************************************;

%put
NOTE:
PROGRAM LAST RUN ON: %sysfunc(date(),worddate.)
AT: %sysfunc(time(),time8.).;
footnote;
options mprint = on
nonotes nomprint nosource;
%let
country   = turkey ;
%include
"/dhcommon/DHLIB/profiles/ti_&country._lev1_autoexec.sas" / nosource2;
%include
"/ti_&country./Lev1/analysis/tesco/activity_modules/auto_process/clubcard_macros/inc
lude_all_clubcard_macros.sas";

* set working directory (same as 1_Model_Builder.sas code).  It contains your model
parameters;
libname loc
"/ti_turkey/Lev1/analysis/tesco/activity_modules/adhoc/customer_retention/sasdata/";

* set week of model parameters (corresponds to week used in 1_Model_Builder.sas
code);
%LET parameters=201237;
```

```
* set week of Loyals scored;
%LET week=201318;

* Select a group of loyals for scoring;
/*

Bring in the Loyal customer list ('PR' and 'VL') you want scored (dib_cust_id and
dib_cust_code) - must be from "week" macro var above.
Call this dataset "LOYALS".




*/

* example;
proc surveyselect data=dib_pdl.dib_customer_metrics_&week. sampsize=10000
method=srs out=LOYALS;
quit;
****************************************;
******** PART 1: data pull loop ********;
****************************************;



%MACRO trans_loop;

data week_loop;
set dib_pdl.dib_time;
where dib_time_code<="&week.";
run;

proc sort data=week_loop;
by descending dib_time_code;
run;

data week_loop;
set week_loop;
if _n_<=8;
n+1;
run;

%DO q=1 %TO 8;

proc sql;
select dib_time_code into: week_num
from week_loop
where n=&q;
quit;
```

```
proc sql;
create table customer_metrics_&q as
select *,
&week_num as tesco_week
from dib_pdl.dib_cust_spend_&week_num.
where dib_cust_id in (select dib_cust_id from LOYALS);
quit;

%end;

%MEND;
%trans_loop;

* stacking;

%MACRO stacker;

data customer_metrics;
set
%DO t=1 %TO 8;
customer_metrics_&t
%END;
;
run;
%MEND;
%stacker;

* creating 8 week customer metrics;
proc sql;
create table customer_metrics as
select dib_cust_id,
dib_cust_code,
sum(dib_cust_spend) as spend,
sum(dib_cust_quantity) as quantity,
sum(dib_cust_visits) as visits,
count(distinct tesco_week) as weeks
from customer_metrics
group by 1,2;
quit;

* creating additional variables;
data customer_metrics;
set customer_metrics;
VPW=visits/weeks;
SPW=spend/weeks;
QPW=quantity/weeks;
run;

*****************************************;
```

```
******** PART 2: Scoring ***************;
****************************************;

* building principal components from 1_Model_Builder param estimates;
proc score data=customer_metrics score=loc.PC_&parameters. out=PC;
var spend visits weeks SPW QPW VPW;
run;

* scoring with parameters from 1_Model_Builder Logistic Reg;
proc score data=PC score=loc.RegOut_&parameters. out=PC_val_scr type=parms;
  var factor1 factor2;
run;

* calculating probabilities;
data PC_val_scr;
set PC_val_scr;
pred=1/(1+exp(-churn));
run;

* creating deciles;
proc sort data=PC_val_scr;
by DESCENDING pred;
run;


proc univariate data=PC_val_scr noprint;
var pred churn;
output out=preddata sumwgt=sumwgt;
run;

data PC_val_scr;
set PC_val_scr;
if (_n_=1) then set preddata;
retain sumwgt;
number+1;
if number < .10*sumwgt then mod_dec=1;
else if number < .20*sumwgt then mod_dec=2;
else if number < .30*sumwgt then mod_dec=3;
else if number < .40*sumwgt then mod_dec=4;
else if number < .50*sumwgt then mod_dec=5;
else if number < .60*sumwgt then mod_dec=6;
else if number < .70*sumwgt then mod_dec=7;
else if number < .80*sumwgt then mod_dec=8;
else if number < .90*sumwgt then mod_dec=9;
else mod_dec=10;
run;



****************************************;
```

```
****** Step 3: finalize your data ********
****************************************;


data YOUR_CUSTOMERS;
set PC_val_scr (keep=dib_cust_code dib_cust_id pred mod_dec);
rename pred=churn_score_13wks  mode_dec=risk_decile;
run;

* NOTE: In this dataset, you will have customer codes, predicted churn score, risk
decile (1=riskiest to 10=least risky);
```

RESUME

PERSONAL INFORMATION

Name Surname : Cagdas Kanar

Date and Place of Birth : 11.11.1985

Foreign Language : English

E-mail : cagdaskanar@hotmail.com

EDUCATION

Degree	Field	College/University	Graduate Year
Graduate	Statistics	Yıldız Teknik Uni.	2014 (expected)
Undergraduate	Industrial Engineering	Orta Dogu Teknik Uni.	2009
Highschool	Maths	Adile Mermerci ALS	2003

WORK EXPERIENCE

Year	Company	Position
2012-now	Dunnhumby	Sr. Insight Analyst
2009-2012	Turkish Airlines	Loyalty Analyst