Response Modeling in Direct Marketing
A Data Mining Based Approach for Target Selection

Sadaf Hossein Javaheri
Response Modeling in Direct Marketing – a Data Mining Based Approach for Target Selection

Supervisors:
Dr. Moez Limayem
Dr. Mohammad Mehdi Sepehri

Referee:
Dr. Amir Albadvi
Dr. Kamal Charsoghi

Prepared by:
Sadaf Hossein Javaheri

Tarbiat Modares University Faculty of Engineering
Department Industrial Engineering

Lulea University of Technology
Department of Business Administration and Social Sciences
Division of Industrial Marketing and E-Commerce

MSc PROGRAM IN MARKETING AND ELECTRONIC COMMERCE Joint

2007
Abstract

Identifying customers who are more likely to respond to a product offering is an important issue in direct marketing. In direct marketing, data mining has been used extensively to identify potential customers for a new product (target selection). Using historical purchase data, a predictive response model with data mining techniques is developed to predict a probability that a customer is going to respond to a promotion or an offer. The purpose of this thesis is to identify the Parsian bank customers who are more likely to respond positively to a new product offering. To reach this purpose a predictive response model using customer historical purchase data is built with data mining techniques. Response modeling procedure consists of several steps. In building a response model one has to deal with some issues, such as: constructing all purchase behavior variables (RFM variables), determining the inputs to the model (feature selection) and class imbalance problem. The purpose of this study is to deal with all these issues and steps of modeling. Thus various data mining techniques and algorithms are used to implement each step of modeling and alleviate related difficulties.

For modeling purpose customers' data (30,000 customers) were gathered from Parsian bank. Based on literature and domain knowledge 85 RFM features and their two-way interactions were constructed from collected data. Since irrelevant or redundant features result in bad model performance thus feature selection was performed in order to determine the inputs to the model. Feature selection was done in three steps using F-score (filter method) and backward elimination on Random Forest (wrapper method). The data was highly unbalanced. We used under-sampling for solving class imbalance problem. Finally SVM was used as a classifier for classification purpose. The result is presented in detail which indicates that Parsian bank can reach three times as many respondents as if they use no model (random sampling) for target selection and this could be very beneficial for the bank: can maximize customers' response to a product offering, minimize the overall marketing cost and improve customer relationship management.

Keywords: Data mining, Direct Marketing, Target Selection, Response Model, Classification
Acknowledgments

This master thesis was written during the summer of 2006 and winter 2007 at Tarbiat Modares University (TMU) in fulfillment of the master program in marketing and e-commerce, jointly held with Lulea University of Technology. I have learned a lot and really enjoyed while working on this master thesis. I would like to thank sincerely all those who helped me with their valuable support during the entire process of this thesis.

I would especially like to express my deep gratitude to my supervisors Prof. Moez Limayem from Lulea University and Dr. Mohammad Mehdi Sepehri from Trabiat Moders University, for their valuable guidance, strong support, encouragement and their helpful comments throughout the progress of this thesis.

Additionally I would like to thank Dr. Amir Albadvi for his useful assistance by introducing me to Parsian bank. I would also like to show my sincere appreciation to Mr. Abdollah Talebi the president of Parsian bank and Mr. Amir Taheri the manager of R&D division in Parsian bank for their kind support, cooperation and for making available the customers data to me.

I would like to extend my thanks and regards to Mr. Teymourpor for his great helps and supports. He kindly helped me through all the steps of this research from the beginning to the end.

Finally, special thanks to my great parents, lovely sister and friends for their consideration and kind support during this thesis. I would like to dedicate the whole thesis to my dear mother and father, to express my deep appreciation towards them, for their never-ending support that they have extended me in every step of my life.

Sadaf Hossein Javaheri
March 2007
Table of Content

Abstract .......................................................................................................................... 1

Acknowledgments ......................................................................................................... 2

List of Tables .................................................................................................................. 6

List of Figures ................................................................................................................. 7

Chapter 1 ..................................................................................................................... 8

Introduction................................................................................................................. 8

1. Introduction ............................................................................................................... 8

1.1 Background of the Study .................................................................................... 9

1.2 Problem Discussion ........................................................................................... 11

1.2.1 Problem Definition ..................................................................................... 12

1.3 Research Objectives and Questions .................................................................. 13

1.4 Subject Importance ........................................................................................... 14

1.5 Outline of the Thesis ......................................................................................... 16

Chapter 2 ................................................................................................................... 17

2. Literature Review ................................................................................................. 17

2.1 Data Mining and Knowledge Discovery in Databases ..................................... 18

2.1.1 What is Data Mining and Knowledge Discovery? .................................... 18

2.1.2 Data Mining Process .................................................................................. 20

2.1.3 Data Mining Tasks and Functionalities ..................................................... 22

2.1.4 Application of Data Mining in Marketing ................................................. 25

2.2 Direct Marketing ............................................................................................... 27

2.2.1 What is Direct Marketing? ......................................................................... 27

2.2.2 Direct Marketing vs. Mass Marketing ....................................................... 29

2.2.3 What is Target Selection in Direct Marketing? ......................................... 29

2.3 Response Modeling in Direct Marketing.......................................................... 31

2.3.1 Response Models ....................................................................................... 31

2.3.2 Recency, Frequency and Monetary Variables (RFM) ............................... 32

2.3.3 Related Work on Response Modeling ....................................................... 34

Chapter 3 ................................................................................................................... 44

Technical Review of Modeling................................................................................. 44

3. Technical Review of Modeling ............................................................................. 44

3.1 Data Preprocessing ............................................................................................ 45

3.1.1 Data Cleaning ............................................................................................. 46

3.1.2 Data Integration ......................................................................................... 46

3.1.3 Data Normalization/Scaling ....................................................................... 47

3.1.4 Feature Construction .................................................................................. 48

3.2 Feature Selection ............................................................................................... 50

3.2.1 What is Feature Selection and why it's important? .................................... 50

3.2.2 Feature Selection Algorithms .................................................................... 51

3.2.2.1 Filter Model ........................................................................................ 52
### Chapter 6

Conclusions and Further Research

- 6. Conclusions and Further Research ................................................................. 125
- 6.1 Conclusion .................................................................................................... 125
- 6.2 Research Limitations .................................................................................. 129
- 6.3 Managerial Implications ............................................................................ 130
- 6.4 Suggestions for Further Research ............................................................... 131

References ............................................................................................................. 133

Appendix 1 ............................................................................................................. 145

Appendix 2 ............................................................................................................. 148
List of Tables

Table 2.1: Literature Review on Response Modeling in Direct Marketing .......... 34
Table 3.1: Confusion Matrix ............................................................................. 70
Table 4.1: Collected Data from Parsian Bank .................................................... 84
Table 4.2: Two Time Horizon for Constructing RFM Variables ......................... 89
Table 5.1: Packages ......................................................................................... 101
Table 5.2: Constructed Features: Dependent Variable and Predictor Variables .... 104
Table 5.3: Selected Interaction Features using F-score ..................................... 108
Table 5.4: Fifty Features Selected Using F-score ............................................... 109
Table 5.5: Random Forest Result ...................................................................... 112
Table 5.6: Selected Features using Backwards Elimination on Random Forest ..... 113
Table 5.7: Class Balancing in Training Set ....................................................... 114
Table 5.8: SVM Model Summary ...................................................................... 117
Table 5.9: Confusion Matrix of Model .............................................................. 118
Table 5.10: SVM Model Performance on Test Set .......................................... 119
Table 5.11: Evaluation Criteria ......................................................................... 120
List of Figures

Figure 1.1: Outline of the Thesis ................................................................. 16
Figure 2.1: The KDD Process .................................................................. 20
Figure 2.2: Tasks in KDD Process ............................................................ 22
Figure 2.3: Data Mining Functionalities .................................................... 23
Figure 2.4: Application of Data Mining for Marketing ....................... 26
Figure 2.5: Response Modeling Based on Bagging Neural Network (Ha and Cho 2005) ................................................................. 37
Figure 2.6: Gains Chart for Four Models: BLR, UBLR, SMLP, and BMLP .......... 38
Figure 2.7: Neural Network for Target Selection (Portharst et al., 2001) ...... 39
Figure 2.8: Response Modeling using LS-SVM based Input Selection Wrapper .... 40
Figure 2.9: Response Modeling with SVM (Shin and Cho, 2006) ......... 40
Figure 2.10: Average Accuracies for the seven response models .......... 43
Figure 3.1: The Filter Approach ............................................................... 52
Figure 3.2: The wrapper Approach ......................................................... 54
Figure 3.3: The Optimum Separation Hyperplane (OSH) .................... 66
Figure 3.4: Kernel Mapping From Input Space to Feature Space ............. 68
Figure 3.5: Example gain charts for typical, ideal and random selection models .. 74
Figure 4.1: Research Design of the Study ............................................. 76
Figure 4.2: General Overvie of Response Model ................................. 79
Figure 4.3: Research Process - Overall Procedure ................................. 80
Figure 4.4: Research Process - Detailed Procedure ............................... 81
Figure 4.5: Split Data into Train and Test Set ......................................... 95
Figure 4.6: Parameter Tunning with Cross-Validation ......................... 99
Figure 5.1: Feature Selection Process ................................................... 107
Figure 5.2: Result of Parameter Tunning for SVM Model .................... 116
Figure 5.3: Lift Chart of the Model ....................................................... 122
Figure 5.4: Features Importance ......................................................... 124
Chapter 1

Introduction

1. Introduction

In this chapter an introduction of the thesis is introduced and provides the reader with an insight to the research area. The chapter begins with a background concerning the research area, followed by the problem discussion, which guides the reader to the research questions and research objective. Also a brief presentation of Parsian Bank is provided in problem discussion. In the end of this chapter the study's importance and the structure of the thesis are also presented.
1.1 Background of the Study

Traditional Large-scale sales pattern is the most familiar sales pattern for companies. Based on this pattern, companies usually aim at their produces, products and then give all the customers same sales promotion. However, this kind of sales promotions neglects the differences among customers. In most cases, these promotions cost a lot, but only get few real profits from customers. That means many promotions are waste. A new business culture is developing today. Within it, the economics of customer relationships are changing in fundamental ways, and companies are facing the need to implement new solutions and strategies that address these changes. The concepts of mass production and mass marketing, first created during the Industrial Revolution, are being supplanted by new ideas in which customer relationships are the central business issue. The traditional process of mass-marketing is being challenged by the new approach of one-to-one marketing. In the traditional process, the marketing goal is to reach more customers and expand the customer base. But given the high cost of acquiring new customers, it makes better sense to conduct business with current customers. In so doing, the marketing focus shifts away from the breadth of customer base to the depth of each customer’s needs. Businesses do not just deal with customers in order to make transactions; they turn the opportunity to sell products into a service experience and endeavor to establish a long-term relationship with each customer (Rygielski et al., 2002). Actually in direct marketing, companies or organizations try to establish and maintain a direct relationship with their customers in order to target them individually for specific product offers or fund raising. Nowadays this type of marketing is being used by growing number of companies, especially financial services, banks and insurance as their main strategy for interacting with their customers. But direct marketers in a wide range of industries from banking and financial services to consumer electronics to computers to office supplies to consumer retail to catalogers are faced with the challenge of continually rising printing and postage costs, on the one hand, and decreasing response rates on the other. Meanwhile, consumer response to campaigns remains extremely low. To combat rising costs and declining response rates, direct marketers are advised to shift from intuitive selection of their audience or the profiling method to more scientific approaches such as predictive modeling and analyzing the customers' data (demographic and historical purchase data) and select
those customers who are most likely respond to a promotion. Identifying customers who are more likely to respond to a product offering is an important issue in direct marketing (Deichmann et al., 2002).

Data mining can solve this problem. Nowadays, a huge amount of information on customers is kept in databases. Thus data mining can be very effective for direct marketing (Ling and Li, 1998). Data mining can be used to improve targeting by selecting which people to contact. Data mining is the process of exploration and analysis of large quantities of data in order to discover meaningful patterns and rules. Simply stated, data mining refers to extracting or “mining” knowledge from large amounts of data. Due to the wide availability of large amounts of data and to the imminent need for turning such data into useful information and knowledge, recently data mining has attracted a great deal of attention in the information industry (Han and Kamber, 2006). Since the first half of the nineties direct marketing has become an important application field for data mining. Identifying customers for marketing purposes is one of the most common applications of data mining. Actually target selection is an important data mining problem from the world of direct marketing. It aims at identifying customers who are most likely to respond positively to a new product. Large databases of customer and market data are maintained for this purpose. The customers or clients to be targeted in a specific campaign are selected from the database given different types of information such as demographic information and information on the customers' personal characteristics like profession, age and purchase history. The main task is to determine the potential customers for a new product from a client database by identifying profiles of customers that are known to have shown interest in a product in the past. That is, the generation of customer models for a given product by analyzing customers’ data obtained from similar previous marketing campaigns. Response model is a well known technique commonly used by direct marketing analysts and it is a profitable tool in fine-tuning direct marketing strategies (Potharst et al., 2002). A response model predicts a probability that a customer is going to respond to a promotion or offer. Response models are typically built from historical purchase data. Using the model, one can identify a subset of customers who are more likely to respond than others. Actually the purpose of these models is the selection of those customers who will be most interested in a particular product offer, so that a percentage as large as
possible of the targeted customers responds to the product offer. A more accurate response model will have more respondents and fewer non-respondents in the subset. By doing so, one can significantly reduce the overall marketing cost without sacrificing opportunities (Shin and Cho, 2006).

Response modeling is usually formulated as a binary classification problem. The customers are divided into two classes, respondents and non-respondents. Various classification methods (classifiers) have been used for response modeling such as statistical and machine learning methods. Neural networks (Bentz and Merunkay, 2000; Bounds and Ross, 1997; Ha et al., 2005; Moutinho et al., 1994; Kim and Street, 2004; Zahavi and Levin, 1997a), decision trees (Haughton and Oulabi, 1997) and support vector machines (Shin and Cho, 2006; Yu and Cho, 2006). As can be seen recently, support vector machine (SVM) has drawn much attention and a few researchers have implemented them for response modeling. SVMs are attracting increasing attention because they rely on a solid statistical foundation and appear to perform quite effectively in many different applications (Lecun et al., 1995; Osuna et al., 1997). From a modeling point of view, response modeling procedure consists of several steps such as: data collection, data preprocessing, feature construction, feature selection, class balancing, classification and evaluation. Various data mining techniques and algorithms have been applied for implementing each step. According to literature review on response modeling very few articles deal with all these steps. Most of them focus only on two or three steps of modeling and for the rest used the results of previous works. Response models can be very beneficial for companies. By using this model companies can identify a subset of customers who are more likely to respond than others. As a result they can maximize the profits of selling the product and minimize the cost of the marketing campaign. Actually they can improve Return on Investment (ROI), also improve customer relationships and retention (Shin and Cho, 2006).

1.2 Problem Discussion

In general financial services and banks in Iran use mass marketing as their strategy for offering and promoting a new product or service to their customers. In this strategy, a single communication message is broadcast to all customers through media such as print, radio or television. In this approach companies do not establish a
direct relationship with their customers for offering new products. However, this kind of sales promotions neglects the differences among customers. Such an approach always implies a high waste: only a small proportion of the customers will actually buy the product. In today's world where products are overwhelming and the market is highly competitive, mass marketing has become less effective. The response rate, the percent of people who actually buy the products after seeing the promotion, is often low (Ling and Li, 1998). So in this competitive environment there is a need for doing direct marketing, which is an effective and efficient way of communicating with customers. As a result, many of banks and financial services in Iran are trying to move away from traditional aggregate-level mass marketing programs and using direct marketing as their main strategy for interacting with their customers. Direct marketing is done by sending product offers and information directly to the customers. This can be done by sending e-mails or SMS to the customers, or by making phone calls, or by addressing the customers by post. By doing this, companies are faced with two important issues: the costs of addressing all the customers and the customers' annoyance due to undesirable mail, SMS or phone calls. In the first issue it is imperative to reduce costs because the costs of such a full-scale mailing campaign soon become too large, and rise above the expected returns, since the product may only be interesting to a subset of the total customer base, while in the second issue it is important to have in mind that the customers’ annoyance may lead to loss of market share because sending many uninteresting product offers to customers leads to irritation as such mail is often considered “junk” (Setnes and Kaymak 2001). So it is important to select those who are more likely to be potential buyers of the new product or service. As a result companies in order to address these problems are searching for accurate methods of identifying their most promising customers to focus specifically on those individuals.

1.2.1 Problem Definition

Parsina Bank with 1,000,000 customers is the second private bank in Iran after the Islamic revolution and started its banking operations on March 2002 (Parsian Bank, 2006). Parsian bank offers different products and services to its customers and use mass marketing as it strategy for offering and promoting a new product or service to its customers. This bank is faced with the challenges of
increasing competition, continually rising marketing costs, decreasing response rates and also lack of direct relationship with its customers. As the market is highly competitive and products are overwhelming, Parsian bank in order to retain its customers try to move away from traditional aggregate-level mass marketing programs and use direct marketing as their strategy for interacting with its customers, because of two important issues in direct marketing: high marketing cost and customer annoyance they have to select subset of customers. To combat these problems, Parsian bank is searching for accurate method to identify the potential customers for a new product offering in order to focus specifically on those individuals. Parsian bank wants to predict a probability that a customer is going to respond to a next marketing campaign and select those customers who are most likely to be potential buyers of the new product or service. In simple word Parsian bank wants to select the “customers” that should be contacted in the next marketing campaigns. The goal of the Parsian bank is to obtain a percentage as higher as possible of positive responses and minimize the marketing cost. This research attempts to address Parsian banks’ issues.

1.3 Research Objectives and Questions

The purpose of this study is to identify those customers who are most likely to respond positively to a new product offering and by a new product or service, based on information provided by the purchase behavior variables. We try to predict whether an existing customer will purchase on the next marketing campaign or not. In order to reach this purpose, we have to develop a predictive response model for Parsian bank using customers' historical purchase data with data mining techniques to select the customers that should be targeted. The purpose of this model is the selections of those customers who will be most interested in a particular product offer (potential customers for a new product), so that percentages as large as possible of the targeted customers respond to the product offer. This can be very beneficial for Parsian bank. By using this model Parsian bank can identify a subset of customers who are more likely to respond to a new product offering than others and establish direct relationship with them. Actually Parsian bank can improve customer relationship and retention, can maximize the profits of selling the product and minimize the cost of the marketing campaign (improve return on investment (ROI)).
The overall procedure of response modeling consists of several steps including data preprocessing, feature construction, feature selection, class balancing, classification and model evaluation. Various data mining techniques and algorithms have been used for implementing each step of modeling. Most of articles focus only on two or three steps of modeling and for the rest used the results of previous works. The purpose of this study is deals with all these steps. To build this model, various data mining techniques and algorithm have to be used. In order to implement these algorithms and techniques an extensive programming is required. Algorithms related to each step of modeling have to be programmed and implemented. All the programming are then combined together to make the whole prediction package.

To fulfill this purpose the following research questions shall be addressed:

Who is more likely to respond to a product offer in the next marketing campaign based on the history of previous marketing campaigns?

1.4 Subject Importance

Terabytes of data are generated everyday in many organizations. To extract hidden predictive information from large volumes of data, data mining techniques are needed. Organization are starting to realize the importance of data mining in their strategic planning and successful application of data mining techniques can be an enormous payoff for the organizations (Lee and Siau, 2001). With the rapid growing marketing business, data mining technology is playing a more and more important role in the demands of analyzing and utilizing the large scale information gathered from customers. Digital marketing is revolutionizing the practice of marketing by bringing the customer to the main point of the marketing process. Data mining plays an important role in this new marketing era by allowing the marketer to harness data about customers and prospects to manage relationships between customers and increase marketing efficiency. Traditional database marketing considers customers as “targets”. The objective is to approach each customer with products/services that are keen only to him/her. The emerging new interactive media, which enables a two-way interaction between the customer and the seller, allows making the customer a real partner in the marketing process, thus making one-to-one marketing a true reality (Zahavi and Levin, 1998). Direct marketing is an effective and efficient way of communicating with customers. In 2002, the Direct Marketing Association (DMA)
reported a positive expansion of the direct marketing industry for the sixth consecutive quarter. Several reasons can be found to account for this continuing development. Most authors ascribe the progress to the constant reduction of data storage costs, the available amount of computing power and the rising number of software packages. These trends enable companies to collect more and more individual (detailed) customer data, so more well-founded decisions can be taken. In addition, and maybe even more important, companies are now more aware that by implementing these facilities and using innovative modeling techniques to improve customer relationships, profitability and sales increase (The Direct Marketing Association, 2006). Also Service providers are finding themselves in the mature markets in which they have to switch their marketing efforts from acquiring new customers towards retention of existing customers (Grant and Schlesinger, 1995; Stone and Woodcock, 1996). This growing awareness stimulates further research into better procedures and techniques. Direct marketing has received the level of attention in customer relationship management (CRM) literature over the last decades.

In current competitive market firms in order to retain their customers must develop good relationship with current and new customers and because of high marketing cost and customer annoyance they have to select subset of customers. Availability of large amount of data to companies about their customers is caused that companies use this data to establish and maintain a direct relationship with the customers in order to target them individually for specific product. Actually data mining can be very effective for direct marketing. Data mining can be used to improve targeting by selecting which people to contact (Ling and Li, 1998). This can be done by generating a response model by analyzing customer historical purchase data. Response models can be very beneficial for companies. By using this model companies can identify a subset of customers who are more likely to respond than others and establish a direct relationship with them. They can maximize response rate and minimize the cost of the marketing campaign and also they can improve customer relationship and retention. As a result the need for direct marketing, data mining and predictive model will continue and there is more need for further research in to better procedures and techniques for target selection in direct marketing.
1.5 Outline of the Thesis

This thesis consists of six chapters. In Figure 1.1 outline of the thesis is shown. In the first chapter the background of the selected research area is presented followed by a problem area discussion and research objectives that end with the research question. The literature review in chapter two will give the reader an overall review on theories relevant to research area and also previews studies related to the topic. This chapter provides an introduction to data mining, direct marketing and also response modeling. In chapter three the technical review of response modeling will be presented. This chapter consists of detail description of each step in response modeling and all the techniques and algorithms can be used for each step, also previews studies related to each step. In chapter four, research methodology, the design and process of this research will be discussed. Also all the methods and techniques were used for each step of modeling will be mentioned in this chapter. Result returned by each step of modeling: data collection, data preprocessing, feature construction, feature selection, prediction result, model evaluation and analysis will be presented in chapter five. Conclusion and managerial implications of the study will be discussed in chapter six and also limitations of the study and suggestions for further research will be provided in this chapter.

Figure 1.1: Outline of the Thesis
2. Literature Review

This chapter is based on the previous introduction and the problem are presented in chapter one. The aim of this chapter is to provide the reader with a literature review concerning the research area. It will provide literature review on data mining and knowledge discovery, direct marketing, target selection in direct marketing and response modeling. Also the previous work on response modeling will be presented in this chapter.
2.1 Data Mining and Knowledge Discovery in Databases

2.1.1 What is Data Mining and Knowledge Discovery?

The digital revolution has made digitized information easy to capture, process, store, distribute, and transmit (Fayyad et al., 1996). With significant progress in computing and related technologies and their ever-expanding usage in different walks of life, huge amount of data of diverse characteristics continue to be collected and stored in databases. The rate at which such data are stored is growing phenomenally. We can draw an analogy between the popular Moore's law and the way data are increasing with the growth of information in this world of data processing applications. The advancement of data processing and the emergence of newer applications were possible, partially because of the growth of the semiconductor and subsequently the computer industry. According to Moore's law, the number of transistors in a single microchip is doubled every 18 months, and the growth of the semiconductor industry has so far followed the prediction. We can correlate this with a similar observation from the data and information domain. If the amount of information in the world doubles every 20 months, the size and number of databases probably increases at a similar pace. Discovery of knowledge from this huge volume of data is a challenge indeed. Data mining is an attempt to make sense of the information explosion embedded in this huge volume of data (Frawley et al., 1991).

There exist several domains where large volumes of data are stored in centralized or distributed databases. Such as: Manufacturing and Production, Business and Marketing, Finance and Investment, Telecommunication network, Digital library, Image archive, Medical imagery, Health care, scientific domain, Internet and etc. Raw data are rarely of direct benefit. Its true value is predicated on (a) the ability to extract information useful for decision support or exploration and (b) understanding the phenomenon governing the data source. In most domains, analyst process data manually and, with the help of statistical techniques, provide summaries and generate reports. However, such an approach rapidly broke down as the size of data grew and the number of dimensions increased. Nowadays databases contain number of data on the order of $10^9$ or above and dimension on the order of $10^3$ are becoming increasingly common. When the scale of data manipulation and
exploration goes beyond human capacities, people need the aid of computing technologies for automating the process.

All these have prompted the need for intelligent data analysis methodologies, which could discover useful knowledge from data. Data Mining is the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data (Fayyad et al., 1996). There are other definitions:

- Data Mining is the process of exploration and analysis of large quantities of data in order to discover meaningful patterns and rules from large amounts of data (Berry and Linoff, 2004).
- Data Mining is the process of discovering interesting knowledge from large amounts of data stored either in databases, data warehouses, or other information repositories. Simply stated, data mining refers to extracting or "mining" knowledge from large amounts of data (Han and Kamber, 2006).

Actually the major reason that data mining has attracted a great deal of attention in the information industry and in society as a whole in recent years is due to the wide availability of huge amounts of data and the imminent need for turning such data into useful information and knowledge (Han and Kamber, 2006). The data mining process is sometimes referred to as knowledge discovery or KDD (knowledge discovery in databases). The term “KDD” (Knowledge Discovery in Databases) refer to the overall process of discovering useful knowledge from data. There is a difference in understanding the terms “knowledge discovery” and “data mining” between people from different areas contributing to this new field. Knowledge discovery in databases is the process of identifying valid, novel, potentially useful, and ultimately understandable patterns/models in data. Data mining is a step in the knowledge discovery process consisting of particular data mining algorithms that, under some acceptable computational efficiency limitations, finds patterns or models in data (Ho, nd).

Shapiro in 1989 invented the KDD term. He said that the term "Knowledge Discovery in Databases" (KDD) became popular in the AI and Machine Learning community. However, the database researchers were on better speaking terms with the business folks and the press, and the term "data mining" became much more
popular in the business press. Data mining term is older than KDD term which was invented in Statistics Data Analysis Association (Shapiro, 2000). Data Mining is actually a misnomer term. Mining of gold from rocks or sand is referred to as gold mining rather than rock or sand mining. Thus, data mining should have been more appropriately named "knowledge mining from data," which is unfortunately somewhat long. "Knowledge mining," a shorter term may not reflect the emphasis on mining from large amounts of data. Nevertheless, mining is a vivid term characterizing the process that finds a small set of precious nuggets from a great deal of raw material. Thus, such a misnomer that carries both "data" and "mining" became a popular choice (Han and Kamber, 2006). Although Knowledge Discovery term is more appropriate but is assumed that Data mining is equal to Knowledge Discovery.

### 2.1.2 Data Mining Process

Generally KDD is an iterative and interactive process involving several steps. This KDD process was chosen (Figure 2.1) according to Unesco definition because of its simplicity and comprehensiveness.

![Figure 2.1: The KDD Process](source: (Ho, nd))

1- **Problem Identification and Definition:**

The first step is to understand the application domain and to formulate the problem. This step is clearly a prerequisite for extracting useful knowledge and for
choosing appropriate data mining methods in the third step according to the application target and the nature of data.

2- **Obtaining and Preprocessing Data:**

The second step is to collect and preprocess the data. Today's real-world databases are highly susceptible to noisy, missing, and inconsistent data due to their typically huge size (often several gigabytes or more), and their likely origin from multiple, heterogeneous sources. Low quality data will lead to low quality mining results. Data preprocessing is an essential step for knowledge discovery and data mining. Data preprocessing include the data integration, removal of noise or outliers, the treatment of missing data, data transformation and reduction of data, etc. This step usually takes the most time needed for the whole KDD process. Detail description of data preprocessing methods and techniques are discussed in the chapter 3.

3- **Data Mining / Knowledge Discovery in Databases:**

The third step is data mining that extracts patterns and/or models hidden in data. This is an essential process where intelligent methods are applied in order to extract data patterns. In this step we have to select first data mining tasks and then data mining method. The major classes of data mining methods are predictive modeling such as classification and regression; segmentation (clustering) and association rules which are explained in detail in section 2.1.3.

4- **Result Interpretation and Evaluation:**

The fourth step is to interpret (post-process) discovered knowledge, especially the interpretation in terms of description and prediction which is the two primary goals of discovery systems in practice. Experiments show that discovered patterns or models from data are not always of interest or direct use, and the KDD process is necessarily iterative with the judgment of discovered knowledge. One standard way to evaluate induced rules is to divide the data into two sets, training on the first set and testing on the second. One can repeat this process a number of times with different splits, and then average the results to estimate the rules performance.

5- **Using Discovered Knowledge:**

The final step is to put discovered knowledge in practical use. Putting the results into practical use is certainly the ultimate goal of knowledge discovery. The information achieved by data mining can be used later to explain current or historical phenomenon, predict the future, and help decision-makers make policy from the existed facts (Ho, nd).
Tasks in the KDD process in more details are shown in Figure 2.2.

2.1.3 Data Mining Tasks and Functionalities

Several data mining problem types or analysis tasks are typically encountered during a data mining project. Depending on the desired outcome, several data analysis techniques with different goals may be applied successively to achieve a desired result. In general, data mining tasks can be classified into two categories: descriptive and predictive. Descriptive mining tasks characterize the general properties of the data in the database. Predictive mining tasks perform inference on the current data in order to make predictions (Han and Kamber, 2006).

Based on the different mining tasks, we can categorize date mining functionalities (methods) as classification, clustering, regression, association rules, sequence discovery, prediction, and so on (Dunham, 2002). Data mining functionalities are used to specify the kind of patterns to be found in data mining tasks (Han and Kamber, 2006). Data mining functionalities is shown in Figure 2.3.
According to Berry and Linoff (2004) basic data mining functionalities are: Classification, Estimation, Prediction, Affinity grouping or associating rules, Clustering, Description and visualization. The first three are all examples of directed data mining, where the goal is to find the value of a particular target variable. Affinity grouping and clustering are undirected tasks where the goal is to uncover structure in data without respect to a particular target variable. Profiling is a descriptive task that may be either directed or undirected.

**Classification (Supervised learning):** Classification maps data into predefined group or classes. Because the classes are determined before examining the data, classification is often considered as supervised learning. Classification algorithms require that the classes be defined based on data attribute values. They often describe these classes by looking at the characteristics of data which are already known to belong to the classes. Classification techniques are: Decision Tree: CART, C4.5, Bayesian Classification: Consists of two type, Naive Bayesian Classification and Bayesian Belief Networks, Neural Network, Support Vector Machines, Associative Classification, Lazy Learners (or Learning from Your Neighbors): k-Nearest Neighbor Classifiers, Case-Based Reasoning. Other Classification Methods: Genetic Algorithms, Rough Set Approach and Fuzzy Set Approach (Berry and Linoff, 2004).
• **Estimation**: Estimation deals with continuously valued outcomes. Given some input data, estimation is used to assign a value for some unknown continuous variable such as income, height, credit balance, or donation amount. Often, classification and estimation are used together, as when data mining is used to predict who is likely to respond to the fund raising campaigns of a charity organization and also to estimate the amount of money donated by each supporter (Berry and Linoff, 2004).

• **Prediction**: Based on past and current data, many real-world data mining applications can be considered as predicting future data states. Prediction is viewed as a type of classification. The difference is that prediction is predicting a future state rather than a current state. Actually the difference is on the emphasis, since in predictive tasks the records are classified according to some predicted future behavior or estimated future value. With prediction, the only way to check the accuracy of the classification or the estimation is to apply the model and then evaluate if its performance was the desired. That is, if the predictive task was to predict the customers who will respond to the next marketing campaign and buy the new product, the only effective way to evaluate the performance of the model is to wait until after the campaign results and count how many of the target customers did actually buy the product. Prediction applications include flooding, speech recognition, machine learning, and pattern recognition (Berry and Linoff, 2004).

• **Affinity grouping or associating rules**: Association rules alternatively referred to as affinity analysis. An association rule is a model that identifies specific types of data associations. They are usually used in the retail sales community to identify items which are often purchased together. The task of affinity grouping is to determine which things go together (e.g. what usually goes together at a shopping cart at the supermarket). Affinity grouping can also be used to identify cross-selling opportunities and to design attractive packages or groupings of products and services (Berry and Linoff, 2004).

• **Clustering (Unsupervised learning)**: Clustering is the task of segmenting a diverse group into a number of more similar subgroups or clusters. What distinguishes clustering from classification is that clustering does not rely on predefined classes, examples, or target concepts. Clustering analyzes data objects without consulting a known class label. In general, the class labels are not present in the training data simply because they are not known to begin with. Clustering can be
used to generate such labels. The objects are clustered or grouped based on the principle of maximizing the intra-class similarity and minimizing the interclass similarity. That is, clusters of objects are formed so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters. Each cluster that is formed can be viewed as a class of objects, from which rules can be derived. Clustering is often done as a prelude to some other form of data mining or modeling. Clustering techniques are: Partitioning: K-means and K-medians, Hierarchical, Density based, and Model based (Berry and Linoff, 2004).

- **Description and visualization:** Sometimes the purpose of data mining is simply to describe what is going on in a complex database, in a way that increases our understanding of the people, the products, or the processes that produced the data in the first place. A good enough description of a behavior will often suggest an explanation for it as well, or at least where to start looking for it (Berry and Linoff, 2004).

### 2.1.4 Application of Data Mining in Marketing

In data mining the information and knowledge gained can be used for applications ranging from Marketing, customer profiling and retention, identifying potential customers, market segmentation, Fraud detection, Text and web mining, e-commerce to production control, manufacturing and science exploration.

Data mining technology in the marketing is a relatively universal application. Such applications are referred to a Boundary Science, because it sets a variety of scientific theories in all. First, two basic disciplines: Information Technology and Marketing. Another very important basis is Statistics. In addition, it relates to the psychology and sociology as well. The charm of this area is just about the wide scope of disciplines study. Generally speaking, through the collection, processing and disposal of the large amount of information involving consumer behavior, identify the interest of specific consumer groups or individual, consumption habits, consumer preferences and demand, moreover infer corresponding consumption group and the next group or individual consumption behavior, then based on them sale produces to the identification consumer groups for a specific content-oriented marketing. This is the basic idea. As automation is popular in all the industry operate processes, enterprises have a lot of operational data. The data are not collected for
the purpose of analysis, but come from commercial operation. Analysis of these data does not aim at studying it, but for giving business decision-maker the real valued information, in order to get profits. Commercial information comes from the market through various channels. For example, purchasing process by credit card, we can collect the customer’s consumption data, such as time, place, interesting goods or services interested, willing price and the level of reception capacity; when buying a brand of cosmetics or filling in a member form can collect customer purchase trends and frequency. In addition, enterprises can also buy a variety of customer information from other consulting firms.

Marketing based on data mining usually can give the customer sales promotion according to his previous purchase records. It should be emphasized data mining is application-oriented. There are several typical applications in banking, insurance, traffic-system, retail and such kind of commercial field. Generally speaking, the problems that can be solved by data mining technologies include: analysis of market, such as Database Marketing, Customer Segmentation & classification, Profile Analysis and Cross-selling. And they are also used for Churn Analysis, Credit Scoring and Fraud Detection (Dunham, 2002). Figure 2.4 shows us the relation between application and data mining techniques clearly and completely.

![Figure 2.4: Application of Data Mining for Marketing](Source: Dunham (2002))
Many of the most successful applications of data mining are in marketing, especially in the area known as database marketing. The objective is to drive targeted and therefore effective marketing and promotional campaigns through the analysis of databases. Nowadays, a huge amount of information on customers is kept in databases. Thus data mining can be very effective for direct marketing. Data mining can be used to improve targeting by selecting which people to contact. It is a win-win game for both the consumers and marketers: Consumers perceive greater value in the (reduced) number of advertising messages, and marketers save by limiting their distribution costs and getting an improved response to the campaign (Cabena et al., 1999).

2.2 Direct Marketing

2.2.1 What is Direct Marketing?

Large amounts of data are nowadays available to companies about their customers. These data can be used to establish and maintain a direct relationship with the customers in order to target them individually for specific product offers and services from the company. Large databases of customer and market data are maintained for this purpose. The customers to be targeted in a specific campaign are selected from the database given different types of information such as demographic information and information on the customer’s personal characteristics like profession, age and purchase history. Usually, the selected customers are contacted directly by mail promoting the new products or services. This type of marketing is called direct marketing (Potharst et al., 2002). In direct marketing, companies or organizations try to establish and maintain a direct relationship with their customers in order to target them individually for specific product offers or fund raising. Nowadays this type of marketing is being used by growing number of companies, especially financial services, banks and insurance as their main strategy for interacting with their customers.

Prior to the late 1960s, what is today called “direct marketing” was known as “mail order” or “direct mail” or “the catalog business.” Although “direct marketing” has today become synonymous to some people with “direct mail,” the term was actually developed in the late 1960s to encompass the newer concepts of targeting
and long-term value. It also was designed to lend credibility to what had become a somewhat disreputable field (Petrison et al., 1997).

For many years marketers have recognized direct marketing as an effective and efficient way of communicating with customers. Direct Marketing is a modern business activity with an aim to maximize the profit generated from marketing to a selected group of customers and makes it possible to offer goods and services or transmit message to a specific, targeted segment of the population by mail, telephone, email or other direct means (Wang et al., 2005). The definition of direct marketing has evolved over time. Terms like directed marketing, relationship marketing, interactive marketing, database marketing and integrated marketing, etc. are now used to describe direct marketing activities (Scovotti and Spiller, 2005). There is no consensus in the literature on the definition of direct marketing. Direct marketing textbook authors, most of them practitioners, present a more unified opinion about the definition. Half of the textbooks use the official definition given by the Direct Marketing Association:

- Direct Marketing is an interactive system of marketing which uses one or more advertising media to affect a measurable response and/or transaction at any location.

Based on the synthesis of academic and practitioner perspectives, the conceptual definition of direct marketing is also proposed:

- Direct marketing is a database-driven process of directly communicating with targeted customers or prospects using any medium to obtain a measurable response or transaction via one or multiple channels.

According to Jonker et.al (2002) Direct Marketing is as a form of marketing that is aimed at obtaining and maintaining direct relations between individual suppliers and buyers within one or more product/market combinations. Also Shin and Cho (2006) said that Direct Marketing is concerned with identifying likely buyers of certain products or services and promoting them to the potential buyers through various channels.
2.2.2 Direct Marketing vs. Mass Marketing

In marketing, there are two opposed approaches to communication: mass marketing and direct marketing. In mass marketing, a single communication message is broadcast to all potential customers through media such as print, radio or television. Such an approach always implies a high waste: only a small proportion of the customers will actually buy the product (Putten, 1999). In today's world where products are overwhelming and the market is highly competitive, mass marketing has become less effective. The response rate, the percent of people who actually buy the products after seeing the promotion, is often low. So in that situation marketers are moving away from traditional aggregate-level mass marketing programs and are searching for accurate methods of identifying their most promising customers in order to focus specifically on those individuals. Against mass marketing is direct marketing. Instead of promoting to customers indiscriminatively, direct marketing studies customers' characteristics and needs, and select certain customers as the target for promotion. The hope is that the response rate for selected customers can be much improved. The objective of direct marketing is to obtain a long-term relationship with customers. Actually the ultimate goal of direct marketing is cost-effective, two-way, one-to-one communication with individual customers (Ling and Li, 1998).

2.2.3 What is Target Selection in Direct Marketing?

Direct marketing has become an important application field for data mining. As explained above in direct marketing, companies or organizations try to establish and maintain a direct relationship with their customers in order to target them individually for specific product offers or fund raising. In fact, nowadays more and more companies are using the information about their customers’ preference and behavior, which is provided by their databases, to do this kind of marketing. Moreover, many companies are using this type of relationship as their main strategy for interacting with their customers (Potharst et al., 2002).

Direct marketing is done by sending product offers and information directly to the customers. This can be done by sending e-mails to the customers, in the case of Internet campaigns, by making phone calls, or by addressing the customers by
post. In the previous cases two main aspects have to be considered: the costs of addressing all the customers and the customers’ annoyance due to undesirable mail or phone calls. In the first case it is imperative to reduce costs because the costs of such a full-scale mailing campaign soon become too large, and rise above the expected returns, since the product may only be interesting to a subset of the total customer base, while in the second case it is important to have in mind that the customers’ annoyance may lead to loss of market share because sending many uninteresting product offers to customers leads to irritation as such mail is often considered “junk” (Setnes and Kaymak, 2001). Assuming that the company has already decided that its marketing strategy is not addressing all the customers, it is important to select those who are more likely to be potential buyers of the new product. This can be done by analyzing data from previous campaigns or by organizing test mail campaigns from which models can be generated to select the customers who will be targeted. Usually data from previous campaigns is used. Large databases of customer and market data are maintained for this purpose. In a specific campaign, the customers or supporters to be targeted are selected from the database, given different types of information such as demographic information or personal characteristics like profession, age and purchase history.

Target selection is an important data mining problem from the world of direct marketing. In fact, the goal of target selection is to determine the potential customers for a new product by identifying profiles of customers that are known to have shown interest in a product in the past; that is, the generation of customer models for a given product by analyzing customers’ data obtained from similar previous marketing campaigns (Kaymak, 2001). The purpose of these models, response models, is the selection of those customers who will be most interested in a particular product offer, so that percentages as large as possible of the targeted customers respond to the product offer. The key to target selection is to maximize the profits of selling the product and minimize the cost of the marketing campaign.

Target selection is done by generating a response model, by analyzing customer historical purchase data with data mining techniques. A response model is discussed in more detail in following section.
2.3 Response Modeling in Direct Marketing

2.3.1 Response Models

As part of relationship marketing programs, marketing executives are taking advantages of vast quantities of customer data newly available. Models commonly used in the direct marketing arena to predict response to mailings and other forms of direct marketing promotions are increasingly being used to up-sell or cross-sell customers who contact companies through call centers, for example. The models can be used to decide which of various possible products or services to offer the customer based on a predicted probability of accepting an offer that is estimated on the fly from data already available on the customer or obtained with a couple questions. A class of such models is called response models, in which the dependent variable is a simple response or not (Ha et al., 2005). Response modeling for database marketing is concerned with the task of modeling the customers’ purchasing behavior. The information at the level of the individual consumer is typically used to construct a response score. A response model predicts a probability that a customer is going to respond to a promotion or offer. Response models compute each customer’s likelihood or propensity to respond to a particular offer of a product or a service. These likelihood values or scores are then used to sort the customers in a descending order. Those who have the highest probability to respond are selected to receive a mailing or a catalog.

Response modeling is a well known technique commonly used by direct marketing analysts (Desarbo and Ramaswamy, 1994). It has proven to be a profitable tool in fine-tuning direct marketing strategies since even small improvements attributed to modeling can create great financial gains (Elsner et al., 2004). The substantive relevance of response modeling comes from the fact that an increase in response of only one percentage point can result in substantial profit increases (Viaene et al., 2002). Given a tendency of rising mailing costs and increasing competition, the importance of response modeling increased. Improving the targeting of the offers may indeed counter these challenges by lowering non response. Using the model, one can identify a subset of customers who are more likely to respond than others. A more accurate response model will have more respondents and fewer non-respondents in the subset. By doing so, one can significantly reduce the overall
marketing cost without sacrificing opportunities. The main purpose of response modeling is to improve future campaign return on investment (Shin and Cho, 2006). Various classification methods (classifiers) have been used for response modeling such as statistical and machine learning methods. Neural networks, decision trees and support vector machines. Related works on response modeling are explained in more details in section 2.3.3.

Response models are typically built from historical purchase data. For data mining purposes, the purchase history can often be translated into features based on measures of Recency, Frequency and Monetary values (Van den Poel, 2003). These variables are described in following section.

2.3.2 Recency, Frequency and Monetary Variables (RFM)

Just like in any modeling, selecting the features that will be used as the explanatory variables in the response model, is an important step. Different types of databases are available to a direct marketing company to be used in target selection. The quality of the database used for selection purposes is considered to be the most important aspect of a successful direct mailing campaign.

A large variety of data is available to the marketing professionals from the databases or external to their organization. The external databases contain information from independent marketing surveys or from demographic studies. There are companies, which are specialized in collecting and maintaining data for sale to direct marketing companies in need of such data. One special category is the databases with demographic information. The demographic information provides additional group-level information, which may be useful in determining targets with similar properties. Contrary to the external databases, the internal customer database of the company provides the most reliable and relevant information regarding the customer behavior. Important information such as purchase history is typically stored in the internal databases (Kaymak, 2001). For data mining purposes, the purchase history can often be translated into features based on measures of Recency (e.g. How recent is the last purchase?, customers who have recently made a purchase are likely to purchase again), Frequency (e.g. How often does a customer buy a product?, customers who make frequent purchases are likely to purchase again) and Monetary
value (e.g. How much money does the customer spend per order?, customers who have spent a lot of money in the past are likely to spend more money now). Cullinan (1977) is generally credited for identifying the three (RFM) variables most often used in database marketing modeling: recency, frequency and monetary value (Bauer, 1988). Since then, the literature has accumulated so many uses of these three variables, that there is overwhelming evidence from existing studies that the RFM variables are the most important set of predictors for modeling mail-order purchasing.

Recency is the number of months since the last purchase and first purchase. It is typically the most powerful of the three characteristics for predicting response to a subsequent offer. This seems quite logical. It says that if you've recently purchased something from a company, you are more likely to make another purchase than someone who did not recently make a purchase. Frequency is the number of purchases. It can be the total of purchases within a specific time frame or include all purchases. This characteristic is second to recency in predictive power for response. Again, it is quite intuitive as to why it relates to future purchases. Monetary value is the total dollar amount. Similar to frequency, it can be within a specific time frame or include all purchases. Of the three, this characteristic is the least powerful when it comes to predicting response. But when used in combination, it can add another dimension of understanding. These three characteristics can be used alone or in combination with other characteristics to assist in CRM efforts (Rud, 2001).

The advantage of the so-called RFM-variables, also called RFM-features, is that the customers’ behavior can be captured by using a relatively small number of features, which improves the transparency of the target selection models derived. It is often assumed in marketing literature, that the RFM-variables are appropriate for capturing the specifics of the customers’ purchase behavior (Bauer, 1988). From a modeling point-of-view, the RFM-variables have the advantage of summarizing the purchase behavior of the customers, using a relatively small number of variables. Contrary to the demographic variables, which can be about 100, the RFM-variables can be 10 or less for a given data set. Hence, the feature reduction is less of a problem when these types of variables are used (Kaymak, 2001). Furthermore, the RFM variables obtained from an internal database represent customer behavior at the
individual level, as opposed to the aggregate level information in the demographic databases. Data about the individual purchase behavior is more informative for targeting individual customers.

### 2.3.3 Related Work on Response Modeling

The main objective of this research is to predict a probability that a customer is going to respond to a product offering. This can be done by generating a response model with data mining techniques. Response modeling is usually formulated as a binary classification problem. The customers are divided into two classes, respondents and non-respondents. Various classification methods (classifiers) have been used for response modeling in direct marketing domain such as statistical and machine learning methods. Literature Review on response modeling is illustrated in Table 2.1.

**Table 2.1: Literature Review on Response Modeling in Direct Marketing**

<table>
<thead>
<tr>
<th>Author, Year</th>
<th>Title</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moutinho, Curry, Davies &amp; Rita, 1994</td>
<td>Neural network in marketing</td>
<td>Neural Network</td>
</tr>
<tr>
<td>Haughton &amp; Oulabi, 1997</td>
<td>Direct Marketing Modeling with CART and CHAID</td>
<td>CART and CHAID</td>
</tr>
<tr>
<td>Bounds &amp; Ross, 1997</td>
<td>Forecasting Customer Response With Neural Networks</td>
<td>Neural Network</td>
</tr>
<tr>
<td>Zahavi &amp; Levin, 1997</td>
<td>Applying neural computing to target marketing</td>
<td>Back-propagation Neural Network</td>
</tr>
<tr>
<td>Ling &amp; Li, 1998</td>
<td>Data mining for direct marketing: Problems and solutions</td>
<td>Naïve Bayesian classifier and C4.5</td>
</tr>
<tr>
<td>Suh, Noh &amp; Suh, 1999</td>
<td>Customer list segmentation using the combined response model</td>
<td>ANN, Statistical Methods</td>
</tr>
<tr>
<td>Malthouse, 1999</td>
<td>Ridge regression and direct marketing scoring models</td>
<td>Ridge Regression</td>
</tr>
<tr>
<td>Colombo &amp; Jiang, 1999</td>
<td>A stochastic RFM model</td>
<td>Stochastic RFM Model</td>
</tr>
<tr>
<td>Coenen, Swinnen, Vanhoof &amp; Wets, 2000</td>
<td>The improvement of response modeling: Combining rule-induction and case-based reasoning</td>
<td>Rule-induction and Case-based Reasoning C5 Algorithm</td>
</tr>
<tr>
<td>Author</td>
<td>Year</td>
<td>Article</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Viaene, Baesens, Van den Poel, Dedene &amp; Vanthienen</td>
<td>2001</td>
<td>Wrapped Input Selection Using Multilayer Perceptrons for Repeat-Purchase Modeling in Direct Marketing</td>
</tr>
<tr>
<td>Viaene, Baesens, Van den Poel, Dedene &amp; Vanthienen</td>
<td>2001</td>
<td>Knowledge discovery in a direct marketing case using least squares support vector machines</td>
</tr>
<tr>
<td>Potharst, Kaymak &amp; Pijls</td>
<td>2002</td>
<td>Neural networks for target selection in direct marketing</td>
</tr>
<tr>
<td>Deichmann, Eshghi &amp; Teebagy Haughton, Sayek</td>
<td>2002</td>
<td>Application of multiple adaptive splines (MARS) in direct response modeling</td>
</tr>
<tr>
<td>Malthouse</td>
<td>2002</td>
<td>Performance-based variable selection for scoring models</td>
</tr>
<tr>
<td>Chiu</td>
<td>2002</td>
<td>A case-based customer classification approach for direct marketing</td>
</tr>
<tr>
<td>Cheung, Kwok, Law, and Tsui</td>
<td>2003</td>
<td>Mining customer product rating for personalized marketing</td>
</tr>
<tr>
<td>Ha, Cho &amp; MacLachlan</td>
<td>2005</td>
<td>Response model using bagging neural networks</td>
</tr>
<tr>
<td>Shin &amp; Cho</td>
<td>2006</td>
<td>Response modeling with support vector machines</td>
</tr>
<tr>
<td>Yu &amp; Cho</td>
<td>2006</td>
<td>Constructing response model using ensemble based on feature subset selection</td>
</tr>
</tbody>
</table>

As can be seen in above table different models with different techniques were built to select the targets in commercial applications. Traditionally, statistical methods, mostly regression techniques, have been applied to response modeling. Most textbooks cover logistic regression as the de facto method due to its simplicity, explainability and availability (Hosmer and Lemeshow, 1989; Sen and Srivastava, 1990). Malthouse (1999) compared ridge regression with stepwise regression on the Direct Marketing Educational Foundation data set 2 (DMEF2). In his study, both methods were used for determining the moderate number of variables in response modeling. Empirically, he showed that ridge regression is a more stable and less risky
method than dropping variables. In his recent report, a similar approach, which additively considered the dollars spent in response to an offer (Malthouse, 2002). Colombo and Jiang (1999) proposed a simple Recency–Frequency–Monetary (RFM) stochastic model for ranking (or scoring) customers. The RFM stochastic model derived from the response distribution of the past was used to estimate the likelihood of future responses. A customer mailing list obtained from a telemarketing company was used for comparing the performance of the stochastic model with that of regression and cross-tabulation model. They reported that the stochastic model provided a more insightful alternative to ranking customers.

Recently, machine-learning methods have been proposed. They include decision trees and neural networks, etc. Haughton and Oulabi (1997) compared the response lifts of two mostly common decision tree algorithms: Classification and Regression Tree (CART) and Chi-Square Automatic Interaction Detector (CHAID). Although the two models are different in their tree-generating mechanism, there was no significant difference in the response-lift perspective. Ling and Li (1998) compared a Naive Bayes response model and a C4.5 response model. Applying the ada-boost algorithm (Freund and Schapire, 1996) to each base model for better performance, they conducted experiments on three direct marketing problems such as loan product promotion, life insurance product campaign, and bonus program. All experiments were designed to discuss the difficulties, which can arise during the response modeling process, such as class imbalance and justifiability of evaluating measures. Coenen, Swinnen, Vanhoof, and Wets (2000) proposed to combine C5, a decision tree algorithm, and case-based reasoning (CBR). In this approach, the C5 based response modeling was conducted in the first step. Then, the respondents classified by the initial model were ranked by a CBR similarity measure. They improved the classification quality by accommodating a better ranking rather than the accuracy of the base response model itself. Chiu (2002) integrated genetic algorithm (GA) into a CBR based response model. For better case identification accuracy, the fittest weighting values on the cases were searched by GA. On the application of an insurance product purchase dataset, the base response model, CBR, achieved better classification accuracy. Deichmann, Eshghi, Haughton, Sayek, and Teebagy (2002) investigated the use of Multiple Adaptive Regression Splines (MARS) as a response model. MARS is an advanced decision tree technique enabling piecewise linear
regression. The MARS response model outperformed the logistic regression model on the DMEF2.

There have also been many reports on neural networks. Moutinho, Curry, Davies, and Rita (1994) predicted bank customers’ responses using neural networks, and Bounds and Ross (1997) showed that neural network based response models improved the response rate from 1 or 2% up to 95%. Zahavi and Levin (1997a) addressed unique merits and demerits of neural networks for response modeling. Viaene, Baesens, Van den Poel, Dedene, and Vanthienen (2001a) proposed to select relevant variables for neural network based response models. Ha, Cho, and MacLachlan (2005) proposed a response model using bagging neural networks. Modeling process is shown in Figure 2.5.

![Figure 2.5: Response Modeling Based on Bagging Neural Network (Ha and Cho 2005)](image)

DMEF4 dataset was used in this study. The data set is from an upscale gift business that mails general and specialized catalogs to its customers several times a year. There are two time periods, “base time period” from December 1971 through June 1992, and “later time period” from September 1992 through December 1992. Every customer in the “later time period” received at least one catalog in early autumn of 1992. They build a response model for period October 1992 through December 1992 time period. That data set consists of 101,532 customers or records and 91 predictor variables or columns. The response rate is 9.4%. 20% of customers who are “important” in terms of their recent purchase were selected. Recency was implemented using “weighted dollar amount spent”. The reduced data set has 20,300 “important” customers with 18.9% response rate. They randomly chose 90% of these data for training. The remaining 10% or 2,030 records were set aside for test. The response rate of the test data set was 17.73%. Such data sets are called unbalanced.
Neural networks do not work very well with unbalanced data sets. They used sub-sampling for solving the class imbalance problem. The training set was balanced by using the sub-sampling the non-respondents. Selection of input variables is a critical step in response modeling. They chose the 17 input variables that were used to predict the response in Malthouse (2001). In order to speed up neural network training, they performed scaling for each variable. The experiments over a publicly available DMEF4 dataset showed that bagging neural networks give more improved and stabilized prediction accuracies than single neural networks and logistic regression. Figure 2.6 shows the gain chart of four models was built in this study. In this study bagging neural network can improve response rate from 17% to 52% and the model accuracy was 84%.

![Figure 2.6: Gains Chart for Four Models: BLR, UBLR, SMLP, and BMLP](image)

**Source:** Ha et al., (2005)

Performance comparison of the methods has been one of the controversial issues in direct marketing domain. Suh, Noh, and Suh (1999) and Zahavi and Levin (1997a, 1997b) found that neural network did not outperform other statistical methods. They suggested combining the neural network response model and the statistical method. On the other hand, Bentz and Merunkay (2000) reported that neural networks outperformed multinomial logistic regression. Potharst, Kaymak, and Pijls (2002) applied neural networks to direct mailing campaigns of a large Dutch charity organization. As a result they used feed-forward network with one hidden layer, 4 hidden nodes, seven inputs (RFM feature) and one output (a measure for the likelihood of response). Model evaluation was done by means of gain chart.
According to their results, the performance of neural networks surpassed that of CHAID or logistic regression. The accuracy of this model with 4 hidden nodes is 74% which is comparable to the accuracy of standard classification methods for this data set. Stages of model building in this study are shown in Figure 2.7.

![Figure 2.7: Neural Network for Target Selection (Portharst et al., 2001)](image)

Although SVM is applied to a wide variety of application domains, there have been only a couple of SVM application reports in response modeling. Cheung, Kwok, Law, and Tsui (2003) used SVM for content-based recommender systems. Web retailers implement a content-based system to provide recommendations to a customer. The system automatically matches his/her interests with product contents through web pages, newsgroup messages, and new items. It is definitely a form of direct marketing that has emerged by virtue of recent advances in the World Wide Web, e-business, and on-line companies. They compared Naive Bayes, C4.5 and 1-nearest neighbor rule with SVM. The SVM yielded the best results among them. More specific, SVM application to response modeling was attempted by Viaene et al. (2001b). They proposed a Least Square SVM (LS-SVM) based wrapper approach. In their study, the input variable pool was composed of RFM and non-RFM variables from the customer dataset provided by a major Belgian mail-order company. All the RFM variables were constructed based on extensive literature and domain experts (see Appendix 1). Then, the wrapper approach was performed in a sequential backward fashion, guided by a best-first variable selection strategy. Their approach, a wrapper around the LS-SVM response model, could gain significant reduction of model complexity without degrading predictive performance with the accuracy of 77% on the training set. The overall procedure of this research is shown in Figure 2.8.
In order to overcome neural network difficulties Shin and Cho (2006) applied Support Vector Machine (SVM), a powerful classification model, to response modeling. In their study they introduced practical difficulties when applying it to response modeling in direct marketing: large training data, class imbalance and binary SVM output. For the intractability problem of SVM training, they presented a pattern selection algorithm (NPPS) that reduces the training set without accuracy loss. The algorithm selects only the patterns near the decision boundary based on neighborhood properties. For the remaining two problems, they used different costs for different classes and used distance from a pattern to the decision hyperplane in the feature space a score. In that experiment, they showed that the proposed solutions worked quite well. The following process was used in this research (Figure 2.9).

Dataset DMEF4, Direct Marketing Educational Foundation, (The Direct Marketing Association) was used in this research. It is concerned with an up-scale gift business that mails general and specialized catalogs to its customer base several times each year. The problem is to estimate how much each customer will spend during the test period, 09/1992– 12/1992, based on the training period, 12/1971–
There are 101,532 patterns in the dataset, each of which represents the purchase history information of a customer. Each customer is described by 91 input variables. A subset of 17 input variables, some original and some derived, were employed just as in (Malthouse, 2001). The dataset has two target variables, target mailing dollars (TARGDOL) and target mailing orders (TARGORD). The former indicates the purchase dollar amount during the test period, and the latter indicates the number of orders during the test period. The TARGDOL or the TARGORD could be directly estimated by building a regression model. Malthouse (2001) built a regression model to estimate the value of TARGDOL. But due to the problems of regression they formulated the problem into a classification one. A new target variable, RESPONSE, was defined as: 1 if TARGDOL (TARGORD)>0, 0 otherwise. Thus, all the customers were categorized into either a non-respondent (class 1) or a respondent (class 2). The response rate is 9.4%, which means the class distribution of the dataset is highly imbalanced. The training set reduced by NPPS, however, showed a different class ratio, m1:m2 65.5:34.5% (=5810:3061) on average. Even though NPPS improved the ratio of the smaller class from 9.4% up to 34.5%, the imbalance problem still remained. Thus, the different misclassification costs, C1 and C2 were set on every dataset as C1 = (m2/M)*C, C2 = (m1/M)*C. The RBF kernel was used with parameter γ set to 0.5, and the misclassification tolerance parameter C set to 10. These parameter settings were determined through a trial-error approach over the combination of C and s, (0.1, 1, 10, 100, 1000)*{0.25, 0.5, 1, 2, 3}, using ten fold cross-validation performance. Finally the performances of the SVM response model was compared in terms of three criteria: accuracies lift chart and computational efficiency.

In building a response model, determining the inputs to the model has been an important issue because of the complexities of the marketing problem and limitations of mental models for decision-making. It is common that the customers’ historical purchase data contains many irrelevant or redundant features thus result in bad model performance. Furthermore, single complex models based on feature subset selection may not always report improved performance largely because of overfitting and instability. Ensemble is a widely adopted mechanism for alleviating such problems. Yu and Cho (2006) proposed an ensemble creation method based on GA based wrapper feature subset selection mechanism. In particular, they proposed an ensemble
of classifiers trained with different subsets of features chosen by a GA based wrapper feature selection approach. The idea was to first create a population of ‘good’ classifiers each of which generalizes well with test data, second select a subset of diverse classifiers from them, and then finally use them in an ensemble to obtain response scores. They used DMEF4 data set from the Direct Marketing Educational Foundation. The problem was to model how much each customer will spend during the Fall-1992 season, i.e. from October 1992 through to December 1992, based on purchase history information. Where customers were categorized as either a respondent (class 1) or as non-respondent (class 0) depending upon whether they made orders during the ‘base time period’ or not, respectively. In DMEF4, there are 101,532 customers in the sample and 91 candidate predictor variables. The response rate was 9.4%. For practical reasons such as computational cost and imbalanced data distribution, only a subset of DMEF4 customers was selected and used for experiments based on ‘weighted dollar amount spent’ as was defined in (Ha et al., 2005). The customers or data samples with Weighted Dollar in top 20% were extracted for this study. The reduced data set now has 20,306 customers with 18.9% response rate. Although the degree of data imbalance is improved from 9.4 to 18.9%, it still shows a high level of skewness. Most classifiers, such as neural network and SVM, do not perform well on this kind of data set. They used the sub-sampling approach so that the training data contains an equal number of respondents and non-respondents.

In addition to the proposed FSE, they built six other models for comparison. These models were:

- MLP-FF: Single MLP with Full Feature Set,
- SVM-FF: Single SVM with Full Feature Set,
- EMLP: Ensemble MLP with Full Feature Set,
- FSVM-PCA: SVM with feature subsets selected by PCA,
- FSVM-DT: SVM with feature subsets selected by DT,
- SVM-FS: SVM with Feature Selection (by GA wrapper),
- FSE: SVM Ensemble based on Feature Selection.
After building these models, performance of models were evaluated in terms of Accuracy, balanced correct classification rate (BCR), ROC point and degree to perfection (DeP). Proposed method was better than other six competing methods in terms of both accuracy and stability. Figure 2.9 depicts the average accuracies of the seven response models built and tested in the experiment. In terms of accuracy, these models showed similar performance, but the proposed FSE performed best with an accuracy of 0.7268, and the worst performance was reported by single SVM model using the full feature set with an accuracy of 0.7109.

Figure 2.10: Average Accuracies for the seven Response Models

Chapter 3

Technical Review of Modeling

3. Technical Review of Modeling

In this chapter technical review of response modeling will be presented. Response modeling procedure consists of several steps such as: data preprocessing, feature construction, feature selection, class balancing, classification and model evaluation. This chapter consists of detail description of each step in response modeling and all the methods and techniques can be used for each step, also briefly review the each step related works.
3.1 Data Preprocessing

In the whole data mining process, data preparation is somehow a significant process. Some book says that if data mining is considered as a process then data preparation is at the heart of this process. However, nowadays databases are highly susceptible to noisy, missing, inconsistent data and due to their typically huge size (often several gigabytes or more), and their likely origin from multiple, heterogeneous sources. The database may contain fields that are obsolete or redundant, missing values, outliers, data in a form not suitable for data mining models and values not consistent with policy or common sense (Larose, 2005). Low quality data will lead to low quality mining results. So data preprocessing is an essential step for knowledge discovery and data mining and improve the efficiency and ease of the data mining process, this becomes an important problem. Several consulting firms have approved that data preparation costs 50%~ 80% resource of the whole data mining process (Han and Kamber, 2006). From this view, it really needs to pay attention to data preparation.

There are a number of data preprocessing techniques: data cleaning, data integration, data transformation and data reduction. Data cleaning can be applied to remove noise, supply missing values and correct inconsistencies in the data. Data integration merges data from multiple sources into a coherent data store, such as a data warehouse. Data transformations involve data normalization/scaling, aggregation and feature construction. In data transformation, the data are transformed or consolidated into forms appropriate for mining. Data reduction can reduce the data size by aggregating, eliminating redundant features (feature selection), or clustering, for instance. These techniques are not mutually exclusive; they may work together. Data processing techniques, when applied prior to mining, can substantially improve the overall quality of the patterns mined and/or the time required for the actual mining (Han and Kamber, 2006). Data preprocessing step usually takes the most time needed for the whole KDD process, Pyle (1999) estimates that data preparation alone accounts for 60% of all the time and effort expanded in the entire data mining process. In following sections data cleaning, data integration, data normalization/scaling and feature construction (data transformation) are explained in
more detail. Also in the next section feature selection (data reduction) is also explained.

### 3.1.1 Data Cleaning

Real-world data tend to be incomplete (lacking attribute values or certain attributes of interest, or containing only aggregate data), noisy (containing errors, or outlier values that deviate from the expected), and inconsistent (e.g., containing discrepancies in the department codes used to categorize items). Data cleaning (or data cleansing) routines attempt to fill in missing values (a record has no value for a particular field), smooth out noise while identifying outliers, and correct inconsistencies in the data. Actually data cleaning preprocessing consists of basic operations such as the removal of noise or outliers if appropriate, collecting the necessary information to model or account for noise, deciding on strategies for handling missing data fields, accounting for time sequence information and known changes (Han and Kamber, 2006).

According to Han and Kamber (2006) there are several strategies to deal with missing values such as: ignore the tuple, fill in the missing value manually, and to smooth out the data to remove the noise, use a global constant to fill in the missing value, use the attribute mean to fill in the missing value and use the most probable value to fill in the missing value.

Noise is a random error or variance in a measured variable. The solution is to smooth out the data and remove the noise using different techniques. Such techniques include binning, regression, and clustering (Han and Kamber, 2006).

### 3.1.2 Data Integration

It is likely that your data analysis task will involve data integration, which combines data from multiple sources into a coherent data store, as in data warehousing. These sources may include multiple databases, data cubes, or flat files. Several issues should be considered during data integration, such as schema integration, correlation analysis for detecting redundancy, and detection and resolution of data value conflicts. Some attributes representing a given concept may have different names in different databases, causing inconsistencies and
redundancies. For example, the attribute for customer identification may be referred to as customer id in one data store, and cust-id in another. Naming inconsistencies may also occur for attribute values. Metadata, correlation analysis, data conflict detection, and the resolution of semantic heterogeneity contribute towards smooth data integration. Careful integration of the data can help improve the accuracy and speed of the mining process (Han and Kamber, 2006).

### 3.1.3 Data Normalization/Scaling

Data Normalization involves scaling the attribute values to make them lie numerically in the same interval/scale, and thus have the same importance. An attribute is normalized by scaling its values so that they fall within a small specified range, such as 0.0 to 1.0. Normalization is particularly useful for classification algorithms involving neural networks, support vector machine, or distance measurements such as nearest neighbor classification and clustering. Such methods provide better results if the data to be analyzed have been normalized, that is, scaled to a specific range such as (0.0, 1.0). In classification normalizing the input values for each attribute measured in the training tuples will help speed up the learning phase. For distance-based methods, normalization helps prevent attributes with initially large ranges (e.g., income) from outweighing attributes with initially smaller ranges (e.g., binary attributes). The main advantage is to avoid attributes in greater numeric ranges dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation (Han and Kamber, 2006). There are many methods for data normalization:

- **Min-max normalization**: min-max normalization performs a linear transformation on the original data.
- **Z-score normalization** (or zero-mean normalization): in z-score normalization the values for an attribute, A, are normalized based on the mean and standard deviation of A.
- **Normalization by decimal scaling**: normalizes by moving the decimal point of values of attribute A. The number of decimal points moved depends on the maximum absolute value of A.
3.1.4 Feature Construction

Feature construction is an important part of supervised classification. Feature construction is a process that discovers missing information about the relationships between features and augments the space of features by inferring or creating additional features (Liu and Motoda, 1998). In attribute construction or feature construction), new attributes are constructed from the given attributes and added in order to help improve the accuracy and understanding of structure in high-dimensional data. By combining attributes, attribute construction can discover missing information about the relationships between data attributes that can be useful for knowledge discovery. Also it is helpful for understanding and accuracy (Han and Kamber, 2006).

Assuming there are \( n \) original features \( A_1, A_2, \ldots, A_n \), after feature construction, we may have the additional \( m \) features \( A_{n+1}, A_{n+2}, \ldots, A_{n+m} \). For example, a new feature \( A_k \ (n < k \leq n+m) \) could be constructed by performing a logical operation on \( Ai \) and \( Aj \) from the original set. All newly constructed features are defined in terms of original features; as such, no inherently new information is added through feature construction. Feature construction attempts to increase the expressive power of the original features. Usually, the dimensionality of the new feature set is expanded and is bigger than that of the original feature set. As its immediate result, it may not directly reduce the number of features. However, many features are rendered redundant after new features are constructed. Intuitively, there could be exponentially many combinations of original features in search for new constructed features, and not all combinations are necessary and useful. Constructing features manually has been shown to be difficult. Feature construction aims to automatically transform the original representation space to a new one that can better achieve data mining objectives: improved accuracy, easy comprehensibility, truthful clusters, revealing hidden patterns, and so forth (Ye, 2003).

The major research issues of feature construction are as follows (Ye, 2003):

1. **How to construct new features**: There are various approaches to feature construction. They can be categorized into four groups: data driven, hypothesis driven, knowledge based, and hybrid. The data driven approach is used to construct new features based on analysis of the available data by applying various operators.
The hypothesis driven approach is used to construct new features based on the hypotheses generated previously. Useful concepts in the induced hypotheses (e.g., rules) can be extracted and used to define new features. The knowledge based approach is used to construct new features applying existing knowledge, and domain knowledge, which is particularly helpful in determining the type of new compound features and choosing suitable operators. Using domain knowledge to construct new features is often recommended because one can thus quickly rule out many impossible combinations and determine an effective measure to evaluate new compound features.

2. How to choose and design operators for feature construction. There are exponentially many operators for combining features to form compound features. Data type plays an important role in determining each operator’s suitability. Conjunction, disjunction, and negation are commonly used constructive operators for nominal features. For numerical features, simple algebraic operators such as equivalence, inequality, addition, subtraction, multiplication, division, maximum, minimum, and average often are used to construct compound features (Zheng, 1998).

3. How to use operators to construct new features efficiently. It is clear that too many operators can be used in feature construction. It is impossible to explore every possible operator. The problem is further complicated by the fact that we could combine any number of features in search of compound features, which leads to combinatorial explosion. It is imperative to find intelligent methods that can avoid exhaustive search of all operators and heuristically try potentially useful operators. This line of research investigates the connections between data mining tasks, data characteristics, and operators that could be effective. Examples in Liu and Motoda (1998) show that one can employ greedy search of different constructed features for decision tree learning or exploit the association of fragmentary domain knowledge to feature construction.

4. How to measure and select useful new features. Not all constructed features are good ones. The danger of including all features (both original and constructed) is to artificially increase the dimensionality—the curse of dimensionality. One option is to handle the selection part by applying feature selection as discussed in the next section (section 3.2). Basically, the new features can be constructed first, unite them
with the original features, and then remove redundant and irrelevant features. When the number of features is very large, it is sensible to keep only those features that are potentially useful, making decisions while a new compound feature is generated to avoid too many features. In the following sections, feature selection and feature selection algorithms is explained in more details.

3.2 Feature Selection

3.2.1 What is Feature Selection and why it's important?

Machine learning algorithms automatically extract knowledge from machine readable information. Unfortunately, their success is usually dependant on the quality of the data that they operate on. If the data is inadequate, or contains extraneous and irrelevant information, machine learning algorithms may produce less accurate and less understandable results, or may fail to discover anything of use at all. Data mining is a process that consists of major steps such as preprocessing, mining and post-processing. Feature selection is frequently used as a preprocessing step to data mining. It is a process of choosing a subset of original features by removing irrelevant and/or redundant ones. Feature selection has been effective in removing irrelevant and redundant features, increasing efficiency in mining tasks, improving mining performance like predictive accuracy, and enhancing result comprehensibility (Dash and Liu, 1997; Blum and Langley, 1997). Actually feature subset selection is the process of identifying and removing as much of the irrelevant and redundant information as possible. This reduces the dimensionality of the data and allows learning algorithms to operate faster and more effectively (Hall and Smith, 1998).

Feature selection is an important part of any classification scheme. The success of a classification scheme largely depends on the features selected and the extent of their role in the model. The objective of performing feature selection is three fold: (a) improving the prediction performance of the predictors, (b) providing faster and more cost effective predictors and (c) providing a better understanding of the processes that generated the data (Guyon, 2003). There are many benefits of variable and feature selection: it facilitates data visualization and understanding, reduces the storage requirements, reduces training times and improves prediction performance. The discrimination power of the features used can be analyzed through
this process. The goal is to eliminate a feature if it gives us little or no additional information beyond that subsumed by the remaining features (Koller and Sahami, 1996). Only a few features may be useful or ‘optimal’ while most may contain irrelevant or redundant information that may result in the degradation of the classifier’s performance. Irrelevant and correlated attributes are detrimental because they contribute noise and can interact counter productively to a classifier induction algorithm (Chun et al., 2002). The information about the class that is inherent in the features determines the accuracy of the model (Koller and Sahami, 1996).

3.2.2 Feature Selection Algorithms

There are two major components in the feature selection algorithms: generation procedure and the evaluation function (Dash and Liu, 1997). Some feature selection methods use a measure to evaluate the goodness of individual features. For example, information measures, distance measures and dependence measures. Features are ranked according to their values on this measure. One can simply choose the first X features as the selected feature subset. X is decided according to some domain knowledge or a user-specified threshold value. Another group of feature selection methods evaluates the goodness of a group of features. The aim is try to find good or poor feature subsets, not good or poor features. Searching for good feature subsets requires extensive computation time. Therefore, some search strategies are applied to decrease the number of subsets to be evaluated, for example, best first search and beam search, forward/backward hill climbing search, and genetic search, etc.

According to the nature of the measures used to evaluate features, feature selection methods can be divided into ‘filter algorithms’ (John et al., 1994) and ‘wrapper algorithms’ (Kohavi and John, 1997). Wrapper algorithms evaluate features with the classification accuracy provided by a target classification algorithm. Filter algorithms, on the other hand, are independent of any learning algorithms and use a particular measure that reflects the characteristics of the data set to evaluate features. Filters select subsets of variables as a pre-processing step, independently of the chosen predictor (Liu and Schumann, 2005). Filter and Wrapper approach are described in more detail in following sections.
3.2.2.1 Filter Model

The earliest approaches to feature selection within machine learning were filter methods. All filter (John et al., 1994) methods use heuristics based on general characteristics of the data rather than a learning algorithm to evaluate the merit of feature subsets. In the filter model, feature selection is performed as a preprocessing step to induction. Thus the bias of the learning algorithm does not interact with the bias inherent in the feature selection algorithm (Koller and Sahami, 1996).

The filter operates independently of any induction algorithm - undesirable features are filtered out of the data before induction commences. Filter methods typically make use of all the training data when selecting a subset of features. Some look for consistency in the data - that is, they note when every combination of values for a feature subset is associated with a single class label (Allmullim and Deitterich, 1991). Another method (Koller and Sahami, 1996) eliminates features whose information content (concerning other features and the class) is subsumed by some number of the remaining features. Still other methods attempt to rank features according to a relevancy score (Kira and Rendell, 1992; Holmes and Nevill-Manning, 1995). The filter approach is shown in Figure 3.1.

![Figure 3.1: The Filter Approach](image)

There are four feature selection filtering methods: Bi-Normal Separation (BNS), correlation coefficients (CC), F-Score and a cross entropy based algorithm. It is found that all four filtering methods perform equally well. Filters have proven to be much faster than wrappers and hence can be applied to large data sets containing many features. The filter approach is generally computationally more efficient. However its, major drawback is that an optimal selection of features may not be independent of the inductive and representational biases of the learning algorithm that constructs the classifier. Since the filter model ignores the effect of the feature subset on the performance of the classifier induction algorithm, an alternative method of
feature selection called the wrapper model is proposed. Further analysis and application of filter methods can be found in Chen et al. (1999) and Salcedo-Sanz et al. (2002).

### 3.2.2.2 Wrapper Model

In the wrapper methodology, recently popularized by Kohavi and John (1997), the feature subset selection is done using the induction algorithm as a black box. The feature subset selection algorithm conducts a search for a good subset using the induction algorithm itself as part of the evaluation function. The accuracy of the induced classifier is estimated using accuracy estimation techniques (Kohavi and Sommerfield, 1995).

The wrapper methodology offers a simple and powerful way to address the problem of variable selection, regardless of the chosen learning machine. In fact, the learning machine is considered a perfect black box and the method lends itself to the use of off-the-shelf machine learning software packages. In its most general formulation, the wrapper methodology consists in using the prediction performance of a given learning machine to assess the relative usefulness of subsets of variables. Actually the basic idea behind a wrapper approach is that at each iteration the quality of the feature subset is evaluated by an inductive learning algorithm that is “wrapped” inside the feature selection procedure as a “black box” (Al-Shahib et al., 2005). In practice, one needs to define: (i) how to search the space of all possible variable subsets; (ii) how to assess the prediction performance of a learning machine to guide the search and halt it; and (iii) which predictor to use. An exhaustive search can conceivably be performed, if the number of variables is not too large. If the number of variables is not too large the search becomes quickly computationally intractable (Amaldi and Kann, 1998). A wide range of search strategies can be used, including, hill-climbing (greedy), best-first, branch-and-bound, simulated annealing and genetic algorithms (for a review see Kohavi and John, 1997). Popular predictors include decision trees, naive Bayes, least-square linear predictors, and support vector machines.

Wrappers are often criticized because they seem to be a “brute force” method requiring massive amounts of computation, but it is not necessarily so. Efficient
search strategies may be devised. Using such strategies does not necessarily mean sacrificing prediction performance. In fact, it appears to be the converse in some cases: coarse search strategies may alleviate the problem of overfitting (Reunanen, 2003). Greedy search strategies seem to be particularly computationally advantageous and robust against overfitting. They come in two flavors: forward selection and backward elimination. If the starting point is chosen with no features then the algorithm successively adds features from this starting point. This is known as forward selection. If however, a starting point with all features is chosen the algorithm successively removes what are considered irrelevant features; this is known as backward elimination (Al-Shahib et al., 2005).

The forward strategy does a better job at removing redundant attributes than the backward strategy. But the backward search strategies are generally more effective than forward search strategies in domains with feature interactions. Because backward search typically begins with all the features, the removal of a strongly interacting feature is usually detected by decreased accuracy during cross validation. Guyon and Elisseeff (2003) argued that forward selection is computationally more efficient than backward elimination to generate nested subsets of variables. However, the defenders of backward elimination argue that weaker subsets are found by forward selection because the importance of variables is not assessed in the context of other variables not included yet.

Source: Kohavi and John (1997)

Figure 3.2: The Wrapper Approach
The disadvantage of the wrapper model is that since a large number of training cycles is required to search for the best performing feature subset, it can be prohibitively expensive. For further consideration of wrapper methods, see Kohavi and John (1997), Vafaie and De Jong (1992) and Salcedo-Sanz et al. (2002).

### 3.2.2.3 Filter Model vs. Wrapper Model

Wrapper model tends to give superior performances it finds features better suited to the predetermined mining algorithm. Actually feature wrappers often achieve better results than filters due to the fact that they are tuned to the specific interaction between an induction algorithm and its training data, but it also tends to be more computationally expensive than the filter model (Blum and Langley, 1997). When the number of features becomes very large, the filter model is usually chosen due to its computational efficiency. Wrappers tend to be much slower than feature filters because they must repeatedly call the induction algorithm and must be re-run when a different induction algorithm is used. Actually the disadvantage of the wrapper model is that since a large number of training cycles is required to search for the best performing feature subset, it can be prohibitively expensive.

Guyon and Elisseeff (2003) explain some of the arguments about these two different approaches. They state that compared to wrappers, filters are faster. Another argument is that some filters provide a generic selection of variables, not tuned for/by a given learning machine. Another compelling justification is that filtering can be used as a preprocessing step to reduce space dimensionality and overcome over fitting. Their main drawback is that they totally ignore the effect of the selected feature subset on the performance of the classification algorithm (Hall and Smith, 1998).

### 3.2.2.4 Evaluation Function

After generating the subset of features, an evaluation function measures the goodness of the subset. Filtering methods use a measure to evaluate the goodness of individual features. For example, information measures, distance measures and dependence measures. Features are ranked according to their values on this measure. One can simply choose the first X features as the selected feature subset. X is decided according to some domain knowledge or a user-specified threshold value.
Wrapper method evaluates the goodness of a group of features. The aim is try to find good or poor feature subsets, not good or poor features. Searching for good feature subsets requires extensive computation time. Therefore, some search strategies are applied to decrease the number of subsets to be evaluated, for example, best first search and Beam search, forward/backward hill climbing search, and genetic search, etc. There are five main types of evaluation functions (Dash and Liu, 1997):

- **Distance** (Euclidean distance measure)
- **Information** (entropy, information gain, etc.)
- **Dependency** (correlation coefficient)
- **Consistency** (minimum features bias)
- **Classifier error rate** (based on a classification algorithm)

The first four are filter models while the last one comes under the wrapper model. Within the filter model, different feature selection algorithms can be further categorized into two groups, namely feature weighting algorithms and subset search algorithms depending on whether they evaluate the goodness of features individually or through feature subsets.

The **distance measure** calculates the physical distance (Dash and Liu, 1997), where the main assumption is that instances of the same class must be closer than those in different class.

Entropy is a measure of the uncertainty of a feature (Yu, 2003). The entropy of a variable (or feature) X is defined in Equation 3.1.

\[
H(X) = -\sum_{i} P(x_i) \log_2 (P(x_i)) \quad (3.1)
\]

and the entropy of a variable X after observing the value of another variable Y is defined by Equation 3.2.

\[
H(X \mid Y) = -\sum_{y} P(y) \sum_{i} P(x_i \mid y) \log_2 (P(x_i \mid y)) \quad (3.2)
\]

Where \( P(x_i) \) is the prior probabilities of all values of X, and \( P(x_i \mid y_j) \) is the posterior probability of X after observing the values of Y.

**Information gain** (Quinlan, 1993) gives the amount by which the entropy of X decreases and reflects the additional information about X provided by Y (Equation 3.3).
In Equation 3.3, a feature Y is regarded more correlated to feature X than to feature Z, if

\[ IG(X \mid Y) > IG(Z \mid Y) \]

Another feature weighting criteria is the correlation measure which measures the correlation between a feature and a class label. The Pearson’s correlation coefficient is given by Equation 3.4

\[ r_{X,Y} = \frac{\sum (X - \bar{X})(Y - \bar{Y})}{(n - 1)\sigma_X \sigma_Y} \]

A positive correlation implies a simultaneous increase in X and Y (Struble). A negative correlation indicates increase in one variable as other decreases. If the \( r_{X,Y} \) has a large magnitude, X and Y are strongly correlated and one of the attributes can be removed (Struble). On the other hand, variables that have a strong correlation with the outcome are retained in the model.

A limitation of all the methods listed above is that they may lead to the selection of a redundant subset of variables. Hence subset search methods are preferred over feature weighting methods. Isabelle et al (Guyon and Elisseeff, 2003) have shown that variables that are independently and identically distributed are not truly redundant. Noise reduction and better class separation can be obtained by adding variables that are presumably redundant. They have also shown that a variable that is completely useless by itself can provide a significant improvement in performance when taken with others. In other words, two variables that are useless by themselves can be useful together. Thus selecting subsets of variables could together have good predictive power, as opposed to ranking the variables according to their individual predictive power.

The wrapper methodology is based on using the prediction performance of a learning machine to assess the relative usefulness of subsets of variables. However, in practice it is necessary to decide on a search strategy that is computationally advantageous and robust against overfitting. Greedy search strategies like forward selection and backward elimination are the most popular search strategies while genetic algorithms, best-first and simulated annealing are among the others (Kohavi and John, 1997).
3.2.3 Related Works on Feature Selection

Machine learning algorithms automatically extract knowledge from machine readable information. Unfortunately, their success is usually dependant on the quality of the data that they operate on. If the data is inadequate, or contains extraneous and irrelevant information, machine learning algorithms may produce less accurate and less understandable results, or may fail to discover anything of use at all. Feature subset selectors are algorithms that attempt to identify and remove as much irrelevant and redundant information as possible prior to learning. Feature subset selection can result in enhanced performance, a reduced hypothesis search space, and, in some cases, reduced storage requirement (Hall and Smith, 1998). As a result feature selection has been studied for several years and several publications have reported performance improvements for such measure when feature selection algorithms are used. This section briefly reviews the related works on feature selection.

According to (Guyon and Elisseeff, 2003; John et al., 1994), feature selection approaches divide into filters and wrappers approaches. Most known approaches are filters which act as a preprocessing step independently of the final classifier (Hermes and Buhmann, 2000; Duda et al., 2000). In contrast, wrappers take the classifier into account as a black box (John et al., 1994; Weston et al., 2001). The pattern recognition and statistics literature offers many filter approaches for feature subset selection (Devijver and Kittler, 1982; Neter et al., 1990). Sequential backward elimination, sometimes called sequential backward selection, was introduced by Marill and Green (1963). Most machines learning induction algorithms do not obey the monotonic restrictions that underlie much of the early work in statistics and pattern recognition, and they are applied to database with a large number of features, so they require special heuristic methods.

Langley (1994) reviews feature subset selection methods in machine learning and contrasted the wrapper and filter approaches. A filter approach by Koller and Sahami (1996) based on cross-entropy seems to work well in practice. Since the introduction of the wrapper approach (John et al., 1994), several authors have experiment with it in various contexts. Langley and Sage (1994) used the wrapper approach to select features for Naïve-Bayes. Kohavi and John (1997) describe the wrapper with other search methods and as search using probabilistic estimates. They
studied the strengths and weaknesses of the wrapper approach and show a series of improved designs. Also they compared the wrapper approach to induction without feature subset selection and to a filter approach to feature subset selection. Significant improvement in accuracy was achieved for some dataset for the decision trees and Naïve-Bayes. Aha and Bankert (1996) showed that feature selection improves the performance based classifier on the spares dataset. Also showed that wrapper models outperform filter models and compared backward selection and forward selection algorithm. Smith and Hall (1998) presented an approach to feature selection for machine learning. The algorithm (CFS) uses features’ performances and inter-correlations to guide its search for a good subset of features. Experimental results were encouraging and show promise for CFS as a practical feature selector for common machine learning algorithms. The correlation-based evaluation heuristic employed by CFS appeared to choose feature subsets that are useful to the learning algorithms by improving their accuracy and making their results easier to understand.

In 2005 Uriarte and Andres used random forest (RF) for variable selection in gene expression data. Random forest is an algorithm for classification, also provides feature importance (Breiman, 2001). Random forest returns a measure of error rate based on the out-of-bag cases for each fitted tree, the OOB error. Using the OOB error as minimization criterion, carry out variable elimination from random forest, by successively eliminating the least important variables (with importance as returned from random forest). Random forests for Variable selection use backwards variable elimination.

Support Vector Machine (SVM) is an effective classification method, but it does not directly obtain the feature importance. Weston et al. (2001) introduced a method of feature selection for Support Vector Machines. The method was based upon finding those features which minimize bounds on the leave-one-out error. Also in 2005 Chen and Lin combined SVM with various feature selection strategies and investigated their performance. Some of them are filters: general feature selection methods independent of SVM. That is, these methods select important features first and then SVM is applied for classification. On the other hand, some are wrapper-type methods: modifications of SVM which choose important features as well as conduct training/testing. They used F-score, which is the filter method, for feature selection, combined SVM with F-score (F-score + SVM) and combined SVM with Random
Forest (RF), which is the wrapper method and F-score, which is the filter method (F-score + Random Forest + SVM). In practice, the wrapper methods cannot handle too many features. Thus, before using RF to select features, they obtained a subset of features using F-score selection first.

In building a response model, determining the inputs to the model has been an important issue because of the complexities of the marketing problem and limitations of mental models for decision-making. It is common that the customers’ historical purchase data contains many irrelevant or redundant features thus result in bad model performance. Various methods have been proposed to alleviate the irrelevant or redundant features in response modeling. Malthouse (2001) proposes a variable selection algorithm that optimizes the performances of scoring model (response model). This article proposed a forward selection algorithm that takes model performance into consideration when making decisions about which variables should enter the model. The empirical results using the DMEF1 and DMEF4 data sets suggested that on average, the overall model performance was improved by 3–4% over fit-based forward selection. Many researchers for building response models such as (Ha et al., 2005; Shin and Cho, 2006) used the features were selected in this article. Viaene et al., (2001b) applied an LS-SVM based input selection wrapper to a real-life direct marketing case involving the modeling of repeat-purchase modeling in direct marketing (whether or not a purchase is made from any product category offered by the direct mailing company) and they showed that by reducing the number of input features, both human understanding and computational performance can often be vastly enhanced. Actually the main objective of this paper involves the detection and qualification of the most relevant variables for repeat-purchase modeling in a direct marketing setting. In this paper binary (buyer versus non-buyer) classification problem was being tackled by using least squares support vector machine (LS-SVM) classifiers. The input selection procedure was based upon a (greedy) best-first heuristic, guiding a backward search mechanism through the input space. Also Viaene et al. (2001a) in another research applied feature selection for repeat purchase modeling. They implemented feature selection using a typical wrapper approach with a best-first search heuristic guiding the backward search procedure towards the optimal input set. Starting with the full set, all inputs are pruned sequentially, i.e. one by one. They used multilayer perceptron neural networks
as induction mechanism for their research. Results of this research indicated that elimination of redundant and/or irrelevant inputs by means of the applied input selection method allow to significantly reducing model complexity without degrading the predictive generalization ability. Recently Yu and Cho (2006) proposed the Feature Selection Ensemble (FSE) model based on a GA wrapper approach. Through experimental studies on DMEF4 data set, they found that the proposed method has, at least, two distinct advantages over other models: first, the ability to account for the important inputs to the response model; second, improved prediction accuracy and stability.

3.3 Class Balancing

3.3.1 Class Imbalance Problem

The class imbalance problem is one of the (relatively) new problems that emerged when machine learning matured from an embryonic science to an applied technology, sufficiently used in the worlds of business, industry and scientific research. Although practitioners might already have known about this problem early, it made its appearance in the machine learning/data mining research circles about a decade ago. Its importance grew as more and more researchers realized that their data sets were imbalanced and that this imbalance caused suboptimal classification performance (Chawla et al., 2004).

The class imbalance problem corresponds to domains for which one class is represented by a large number of examples while the other is represented by only a few. Actually the class imbalance problem typically occurs when, in a classification problem, there are many more instances of some classes than others. In such cases, standard classifiers tend to be overwhelmed by the large classes and ignore the small ones. In practical applications, the ratio of the small to the large classes can be drastic such as 1 to 100, 1 to 1,000, or 1 to 10,000. The class imbalance problem is of crucial importance since it is encountered by a large number of domains of great environmental, vital or commercial importance, and was shown, in certain cases, to cause a significant bottleneck in the performance attainable by standard learning methods which assume a balanced distribution of the classes (Japkowicz, 2000). This problem is prevalent in many applications, including: fraud detection, risk
management, direct marketing, text classification, and medical diagnosis/monitoring, but there are many others.

Classifier when faced with imbalanced datasets where the number of negative instances far outnumbers the positive instances, the performance drops significantly. With imbalanced data, the simplest hypothesis is often the one that classifies almost all instances as negative and the positive instances can be treated as noise and ignored completely by the classifier. However, the most commonly used classification algorithms do not work well for such problems because they aim to minimize the overall error rate, rather than paying special attention to the positive class (Akbani, 2004). A popular approach towards solving these problems is to bias the classifier so that it pays more attention to the positive instances. This can be done, for instance, by increasing the penalty associated with misclassifying the positive class relative to the negative class. Another approach is to preprocess the data by oversampling the majority class or undersampling the minority class in order to create a balanced dataset. These approaches are described in more detail in following section.

3.3.2 Strategies for Class Balancing

Regarding class imbalance, many researchers have recognized this problem and suggested several methods. A number of solutions to the class-imbalance problem were previously proposed. These solutions include over-sampling and under-sampling method. The over-sampling method consists of re-sampling the small class at random until it contains as many examples as the other class (Japkowicz, 2000). Actually the over-sampling increases the frequency of the minority class in the training set (Weiss and Provost, 2001; Drummond and Holte, 2003). The under-sampling method, closely related to the over-sampling method, consists of eliminating, at random, elements of the over-sized class until it matches the size of the other class (Japkowicz, 2000). Actually under-sampling decreases the frequency of the majority class in the training set. Most research has been focused on these approaches. In addition to under-sampling and over-sampling, another approach is to assign misclassification costs. In a general way, misclassified examples of the minority class are more costly than misclassified examples of the majority class (Bharatheesh and Iyengar, 2003).
Among these approaches the under-sampling method has been very popular. In literature many researcher under-sample the negative (majority) class for several reasons: First, as over-sampling makes exact copies of the minority class, it tends to result in overfitting of the data. Furthermore, over-sampling increases the size of the training set and hence the time to build a classifier. Also, using over-sampling is discouraged in parts of the machine learning literature, e.g. in Drummond and Holte (Drummond and Holte, 2003) because “under-sampling produces a reasonable sensitivity to changes in misclassification costs and class distribution. Over-sampling shows little sensitivity, there is often little difference in performance when misclassification costs are changed”.

### 3.3.3 Class Balancing Related Works

As explained above, approaches for addressing the imbalanced training data problem can be categorized into two main divisions: the data processing approach and the algorithmic approach.

The data processing approach can be further divided into two methods: under-sample the majority class, and over-sample the minority class. The one-sided selection proposed by Kubat (Kubat and Matwin, 1997) is a representative under-sampling approach which removes noisy, borderline, and redundant majority training instances. However, these steps typically can remove only a small fraction of the majority instances, so they might not be very helpful in a scenario with a majority-to-minority ratio of more than 100:1 (which is becoming common in many emerging applications). Multi-classifier training (Chan and Stolfo, 1998) and Bagging (Breiman, 1996) are two other under-sampling methods. These methods do not deal with noisy and borderline data directly, but use a large ensemble of sub-classifiers to reduce prediction variance. Over-sampling (Chawla et al., 2000; Weiss and Provost, 2001) is the opposite of the under-sampling approach. It duplicates or interpolates minority instances in the hope of reducing the imbalance. Ling and Li (1998) combined over-sampling with under-sampling, but this combination did not provide significant improvement in the “lift index” metric that they used. Drummond and Holte (2003) compared under-sampling and over-sampling and then showed that under-sampling produces a reasonable sensitivity to changes in misclassification costs and class distribution.
The algorithmic approach is orthogonal to the data processing approach. Nugroho (Nugroho et al., 2002) suggests combining a competitive learning network and a multilayer perceptron as a solution for the class imbalance problem. Kubat et al. (Kubat and Matwin, 1997; Drummond and Holte, 2003; Elkan, 2001; Ling and Li., 1998) modify the decision-tree generator to improve its learning performance on imbalanced datasets. For SVMs, few attempts (Karakoulas and Taylor, 1999; Lin et al., 2002; Veropoulos et al., 1999) have dealt with the imbalanced training-data problem. Veropoulos et al. (Lin et al., 2002; Veropoulos et al., 1999) use different penalty constants for different classes of data.

As discussed above various methods have been proposed to alleviate the class imbalance problem. A few of them have been applied to response modeling such as undesampling (Ha et al., 2005; Zhavi and Levin, 1997b), ensemble (Yu and Cho, 2006) and cost-modifying methods (Shin and Cho, 2006). Neural network and support vector machine do not work very well with unbalanced dataset. As Zahavi and Levin (1997b) have pointed out, class imbalance problem is an important issue in applying neural network to target marketing and modeling the response. They used under-sampling method to balance the training set. Also Ha et al., (2005) for building the response model with bagging neural network, under-sample the non-respondents class. The resulting training dataset had a 50% response rate. Shin and Cho (2006), proposed to employ different misclassification costs to different class errors in the objective function, which is naturally allowed in SVM. For SVM, the cost-modifying method can be applied: A small cost is assigned to the non-respondent class while a large cost to the respondent class. This is to assure that the minority class is not neglected. Among these approaches the under-sampling method has been very popular.

3.4 Classification Algorithm

Various classification methods (classifiers) have been used for response modeling such as statistical and machine learning methods. Neural networks (Bentz and Merunkay, 2000; Bounds and Ross, 1997; Ha et al., 2005; Moutinho et al., 1994; Kim and Street, 2004; Zahavi and Levin, 1997a), decision trees (Haughton and Oulabi, 1997) and support vector machines (Shin and Cho, 2006; Yu and Cho, 2006).
In this thesis for building a response model, Support Vector Machine (SVM) was used as a classifier for classification purpose. Support Vector Machines (SVMs) is the mathematical structure, or model, that underlies learning. It is machine learning technique that learn patterns based on training data, fit the models to this training data and predict or classify unseen (or future) data. The active development of SVMs started in 1980s. Currently SVMs demonstrate superior performance in various problems compared to other classifier. The applications of SVMs are expected to expand. SVMs are attracting increasing attention because they rely on a solid statistical foundation and appear to perform quite effectively in many different practical applications as diverse as face detection and recognition, handwritten character and digit recognition, text detection and categorization, etc. (Dumais, 1998; Heisele et al., 2000; Moghaddam and Yang, 2000; Osuna et al., 1997).

3.4.1 Support Vector Machines (SVMs)

Support vector machine (SVM) is a popular technique for classification. In recent years, the SVM (Chapelle and Vapnik, 1999; Pontil, 1998; Vapnik, 1995, 1998) has become an effective tool for pattern recognition, machine learning and data mining, because of its high generalization performance. The goal of SVM is to produce a model which predicts target value of data instances in the testing set which are given only the attributes (Hsu et al., 2003).

The foundations of SVM have been developed by Vapnik, and are gaining popularity due to many attractive features, and promising empirical performance. The formulation of SVM embodies the Structural Risk Minimization (SRM) principle, as opposed to Empirical Risk Minimization (ERM) commonly employed with other statistical methods. SRM minimizes the upper bound on the generalization error, as against ERM which minimizes the error on the training data. Thus, SVMs are known to generalize better. The SRM technique consists of finding the optimal separation surface between classes due to the identification of the most representative training samples called the support vectors. SVM attempts to position a decision boundary so that the margin between the two classes is maximized.

The idea of SRM is to find a hypothesis $h$ which reflects the well-known trade-off between the training error and the complexity of the space. SVM learns
from the training set to find a decision surface (classifier) in the vector space of data points, that best separates the data points into two classes (relevant and non-relevant). The decision surface by SVM for linearly separable space is a hyperplane which can be written as the Equation (3.5) where \( x \) is an arbitrary feature vector and \( w \) and \( b \) are learned from a training set of linearly separable data (Masuyama and Nakagawa, 2002).

\[
\mathbf{w} \cdot \mathbf{x} + b = 0 \quad (3.5)
\]

Figure 3.3 illustrates the optimum separation hyperplane. The solid line is the hyperplane and the two dashed lines parallel to the solid line indicate the boundaries in which one can move the solid line without any misclassification. SVM finds the solid line which maximizes the margin (distance) between those parallel dashed lines. The training dataset which lie on either of two dashed lines are called support vectors. The purpose of SVM is to find the hyperplane that best classifies the training datasets with the maximum accuracy rate and minimum error rate.

Figure 3.3: The Optimum Separation Hyperplane (OSH)

The aim of SVM is to maximize the distance between the two parallel lines which can be expressed as Equations (3.6) and (3.7). The distance between these two lines is equal to \( 2/\|w\| \).

\[
\mathbf{x} \cdot \mathbf{w} + b = 1 \quad (3.6)
\]

\[
\mathbf{x} \cdot \mathbf{w} + b = -1 \quad (3.7)
\]
In order to maximize \( M = \frac{2}{\|w\|} \), we should minimize \( \|w\| \) under the constraints (3.8) and (3.9) that can be summarized as Equation (3.10):

\[
\begin{align*}
\mathbf{x}_i \cdot \mathbf{w} + b &\geq +1 \quad \text{for } y_i = +1 \\
\mathbf{x}_i \cdot \mathbf{w} + b &\leq -1 \quad \text{for } y_i = -1
\end{align*}
\]  

\[ y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geq 0 \quad \text{(3.10)} \]

\( \mathbf{x}_i \) is a feature vector of the \( i \)th training document represented by an \( n \) dimensional vector and \( y_i \) is the class (positive (+1) or negative (-1)) label of the \( i \)th training data. All vectors lying on one side of the hyperplane are labeled as -1, and all vectors lying on the other side are labeled as 1. The training dataset which lie on either of two dashed lines are called support vectors.

To solve the problem, it should be switched to Lagrangian formulation. The reason for doing this is that the constraints will be replaced by constraints on the Lagrange multipliers themselves \( (\alpha_i) \), which will be much easier to handle. The Lagrangian formulation can be written as Equation (3.11):

\[
L_P \equiv \frac{1}{2} \|w\|^2 - \sum_{i=1}^{i} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{i} \alpha_i \quad \text{(3.11)}
\]

\( L_P \) should be minimized. This is a convex Quadratic Programming (QP) optimization problem which is very time-consuming. Sequential Minimal Optimization (SMO) is an algorithm for training the SVM where this large QP problem is broken down into a series of smallest possible QP problems which are solved analytically. SMO can handle very large training datasets and considerably speeds up training times. An equivalent dual problem (3.12) can be solved instead of \( L_P \). \( L_D \) is resulted by substituting the \( L_P \) with the equations (3.10)

\[
L_D = \sum_{i} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad \text{(3.12)}
\]

\[
\mathbf{w} = \sum_{i} \alpha_i y_i \mathbf{x}_i
\]

\[
\sum_{i} \alpha_i y_i = 0
\]
The major parameters of SVM are taken from the training patterns. If the training dataset is not linearly separable, a kernel method is used to simulate a non-linear projection of the data in a higher dimensional space, where the classes are linearly separable (Shin and Cho, 2006). A kernel \( k(x, y) \) is a similarity measure defined by an implicit mapping \( \phi \), from the original space to a vector space (feature space) such that: \( k(x, y) = \phi(x) \cdot \phi(y) \) (Figure 3.4). In the basic form, SVM learns a linear threshold function. Nevertheless, by a simple plug-in of an appropriate kernel function, they can be used to learn polynomial and radial basis function classifiers. The use of kernels makes it possible to map the data implicitly into a feature space and to train a linear machine in such a space, potentially side-stepping the computational problems inherent in evaluating the feature map (Cristianini and Shawe-Taylor, 2002).

### Figure 3.4: Kernel Mapping From Input Space to Feature Space

Three common families of kernels are Linear Kernel, Polynomial Kernel and Radial Basis Functions (RBF) Kernel. The choice of the kernel function is crucial to the efficiency of support vector machines and it is shown that RBF-kernel yields the best performance. Three common kernels are mathematically defined in following Equations (Hsu et al., 2003):

1. **Polynomial kernel.** \( K(x, y) = ((\gamma \cdot x^* \cdot y) + 1)^d \) \hspace{1cm} (3.13)
2. **RBF kernel.** \( k(x, y) = \exp(-\gamma \cdot |x - y|^2) \) \hspace{1cm} (3.14)
3. **Linear kernel.** \( k(x, y) = x^* \cdot y \) \hspace{1cm} (3.15)
3.4.2 Comparison of SVMs with other Classifiers

SVMs offer advantages over multivariate classifiers. They are free of optimization problems of Neural Networks (NNs) because they present a convex programming problem, and guarantee finding a global solution. They are much faster to evaluate than density estimators (like maximum likelihood classifiers), because they make use of only the relevant data points, rather than looping over each point regardless of its relevance to the decision boundary. Also SVMs can handle data simultaneously, without losing the degree of accuracy (Candade, 2004). Recent research has shown SVMs outperform NNs in classification tasks. SVMs show distinct advantages such as better generalization, increased speed of learning, ability to find a global optimum and ability to deal with linearly non-separable data. Thus, though NNs are more widely known and used, SVMs are expected to gain popularity in practical applications (Candade, 2004). Given a set of points that all belong to one of the two classes, an SVM can find the hyperplane that leaves the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyperplane. This optimal separating hyperplane can minimize the risk of misclassifying examples of the test set. On the other hand, Conventional neural networks tend to overfit the training dataset, resulting in poor generalization since parameter selection is based on Empirical Risk Minimization (ERM) principle, which minimizes the error on the training set (Shin and Cho, 2006). SVMs are attracting increasing attention because they rely on a solid statistical foundation and appear to perform quite effectively in many different applications (Lecun et al., 1995; Osuna et al., 1997). After training, the separating surface is expressed as a certain linear combination of a given kernel function centered at some of the data vectors (named support vectors). All the remaining vectors of the training set are effectively discarded and the classification of new vectors is obtained solely in terms of the support vectors.

3.5 Evaluation Measurements

After examining the data and applying automated methods for data mining, the quality of the end-product must be considered carefully. Predictive performance measurement often is considered the key evaluator of the success of an application. Accuracy and lift/gain charts are the methods for evaluating a model’s performance
in the marketing domain. These are not new, but they can offer insight into how
different models will perform for many application situations. Many classification
models can be modified so that the output is a probability of the given class, and
hence depending on the threshold or some other decision making parameter value,
one can get a family of models from one (Ye, 2003). The most basic performance
measures can be described by a confusion matrix. In following sections confusion
matrix, accuracy, weighted accuracy and lift chart are explained.

3.5.1 Accuracy, TPR, TNR and Weighted Accuracy

The most basic performance measures can be described by a confusion matrix
in. The confusion matrix (Kohavi, 1988) contains information about actual and
predicted classifications done by a classification system. The confusion matrix for a
binary classifier is shown in Table 3.1. The columns represent the predicted
classifications, and the rows represent the actual (true) classifications for each of all
records.

Table 3.1: Confusion Matrix

<table>
<thead>
<tr>
<th>True</th>
<th>Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Class 1 (Negative)</td>
</tr>
<tr>
<td>Class 1</td>
<td>TN</td>
</tr>
<tr>
<td>(Negative)</td>
<td></td>
</tr>
<tr>
<td>Class 2</td>
<td>FN</td>
</tr>
<tr>
<td>(Positive)</td>
<td></td>
</tr>
</tbody>
</table>

Source: (Yu and Cho, 2006)

Where
TP = positive examples correctly labeled as positives
FP = negative examples incorrectly labeled as positive
TN = negative examples correctly labeled as negative
FN = positive examples incorrectly labeled as negative
From such matrix it is possible to extract a number of metrics to measure the performance of learning systems. A common evaluation strategy is to consider classification accuracy or its complement error rate. Based on the above values, the accuracy and error rate can be defined as follow:

- **Accuracy (Acc)**: proportion of total number of predictions that were correct (Equation 3.16).

\[
Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.16}
\]

- **Error Rate (E)**: proportion of total number of predictions that were incorrect (Equation 3.17).

\[
Error \ Rate = \frac{FP + FN}{TP + FP + TN + FN} = 1 - Acc \tag{3.17}
\]

The accuracy (Acc) and Error rate (E) are widely used metric for measuring the performance of learning systems. However, when the prior probabilities of the classes are very different, such metric might be misleading. Accuracy alone is not an adequate measure of performance especially where the number of negative cases is much greater than the number of positive cases. A trivial classifier that predicts every case as the majority class can still achieve very high accuracy (Chen *et al.*, 2004). Suppose there are 100 cases, 95 of which are negative and 5 positive. If the system classified all the cases as negative, the accuracy would be 95%, even though the classifier missed all the positive cases. These are called as stupid classifiers, which are of no interest to the business community. In such cases other metrics such as true positive rate, false positive rate, true negative rate, false negative rate and weighted accuracy can be used to evaluate the performance of learning algorithms (Bharatheesh and Iyengar, 2004). Based on confusion matrix (Table 3.1) these performance metrics are defined as follow:

- **True Positive Rate (TPR)**: proportion of positive cases that were correctly identified, as calculated using the Equation 3.18.

\[
TPR = \frac{TP}{TP + FN} \tag{3.18}
\]
- **False Positive Rate (FPR):** proportion of negatives cases that were incorrectly classified as positives, as calculated using the Equation 3.19.

\[
FPR = \frac{FP}{FP + TN}
\]  

(3.19)

- **True Negative Rate (TNR):** proportion of negatives cases that were correctly classified correctly, as calculated using the Equation 3.20.

\[
TNR = \frac{TN}{TN + FP}
\]  

(3.20)

- **False Negative Rate (FNR):** proportion of positive cases that were incorrectly classified as negative, as calculated using the Equation 3.21.

\[
FNR = \frac{FN}{FN + TP}
\]  

(3.21)

For any classifier, there is always a trade off between true positive rate and true negative rate; and the same applies for false positive rate and false negative rate. In the case of learning extremely imbalanced data, quite often the rare class is of great interest. In many applications such as direct marketing, drug discovery and disease diagnosis, it is desirable to have a classifier that gives high prediction accuracy over the minority class (*True Positive Rate - Acc+*), while maintaining reasonable accuracy for the majority class (*True Negative Rate - Acc−*). Thus it is important to study the other parameters. Weighted Accuracy is often used in such situations (Equation 3.22). Weights can be adjusted to suit the application (Chen et al., 2004).

\[
\text{Weighted Accuracy} = \frac{\lambda (\text{True Positive}) + (\text{True Negative})}{\lambda (\text{True Positive} + \text{False negative}) + (\text{True Negative} + \text{False positive})}
\]

(3.22)

In such cases where the number of negative cases is much greater than the number of positive cases, also *lift chart* can be used. The lift chart gives a graphical representation of these parameters for various thresholds on the output and encapsulates all the information contained in the confusion matrix. Lift chart is explained in more detail in following section.
3.5.2 Lift/Gain Chart

A marketing agency is planning to send advertisements to selected households with the goal to boost sales of a product. The agency has a list of all households where each household is described by a set of attributes. Each advertisement sent costs a few pennies, but it is well paid off if the customer buys the product. Therefore an agency wants to minimize the number of advertisements sent, while at the same time maximize the number of sold products by reaching only the consumers that will actually buy the product. Therefore it develops a classifier that predicts the probability that a household is a potential customer. To fit this classifier and to express the dependency between the costs and the expected benefit the lift chart can be used (Vuk and Curk, 2006).

For many applications the aim of prediction is to identify some desired class members (e.g., customers) for whom some action (e.g., mailing advertisement circulars) is to be performed. It is more flexible than classification if the prediction is in the form of ranking based on the predicted class probability. Lift/gain chart is a graphical representation of the advantage of using a predictive model to choose which customers to contact. The gain charts (also called lift charts) show the gains to be expected from the utilization of a particular response model, over the gains usually obtained when no model is used and the clients are selected at random. Actually it shows that how much more likely we are to receive respondents than if we contact a random sample of customers. A steeply increasing gain-chart is desirable, i.e. the model should target as many respondents as possible while addressing as few clients as possible. For instance, if in a given data set 10% of the clients are respondents, an optimal model will target all these by approaching 10% of the clients, i.e. all clients addressed are respondents. The gains/lift charts can help us assess model performance; the greater the area between the lift curve and the baseline, the better the model (Ye, 2003).

Response models compute each customer’s likelihood or propensity to respond to a particular offer of a product or a service. Lift/gain chart is created by sorting all the prospects according to their likelihood of responding as predicted by the model. As the size of the mailing list increases, we reach farther and farther down the list (Berry and Linoff, 2004). The Y axis shows the percentage of responders that
is actually captured by the model, which is the true positive rate (TPR), given the percentage of customers targeted, (TP+FP/TP+FP+TN+FN), indicated on the X axis. Thus, the gain chart shows how many of the positive responders in the analyzed campaign (campaign used to generate the target selection model) would have been targeted by the model for any given campaign size (Vuk and Curk, 2006). Figure 3.5 shows an example of gain charts for typical, ideal and random selection models. The top line shows an ideal model. For instance, if in a given dataset 10% of the clients are respondents, an ideal model will target all these by approaching 10% of the clients, i.e., all clients addressed are respondents. The ideal response model would produce a gain chart that rises as steeply as possible to the 100% of responders, as the model should target as many respondents as possible while addressing as few clients as possible. Conversely, a straight line from lower left to upper right corresponds to a random selection; no model is used for target selection. A random selection of targets would produce a straight line from the point (0%, 0%) to the point (100%, 100%). The model in Figure 3.5 shows that with a campaign directed to 20% of the clients, 40% of the respondents are approached, as opposed to only 20% in a randomly targeted mailing of the same size.

![Gain Chart Example](source: Ye, 2003)

**Figure 3.5: Example gain charts for typical, ideal and random selection models**
Chapter 4

Research Methodology

4. Research Methodology

In this chapter the methodological approaches used in order to reach to the objective and answer the research question are described. It starts with research design and then presented the research process were used in this thesis. All the methods and techniques were used for each step of modeling: data collection, data preprocessing, feature construction, feature selection, class balancing, classification and model evaluation are also mentioned in this chapter.
4.1 Research Design

The research design is a framework for conducting marketing research (Malhotra, 1996). Consequently, it's a basic plan that guides the data collection and analysis phase of the research. It specifies information of the type of information to be collected, the sources of the data, and the data collection procedure. A good research design will ensure that the information collected will be consistent with the objectives of the study and that the procedures regarding data collection is accurate and efficient. The research design of this study is illustrated in Figure 4.1. Details descriptions are explained below.

![Figure 4.1: Research Design of the Study](image)

4.1.1 Research Purpose

Research can be classified in terms of their purpose. Accordingly, Saunders et al., (2000) mentioned that they are most often classified Exploratory, Descriptive and Explanatory (Causal). The different types are explained below:

Exploratory research aims to develop initial hunches or insights and provide direction for any further research needed. The primary purpose of exploratory research is to shed light on the nature of a situation and identify any specific
objectives or data needs to be addressed through additional research. Exploratory research is most useful when a decision maker wishes to better understand a situation and/or identify decision alternatives. Exploration is particularly useful when researchers lack a clear idea of the problems they will meet during the study. The object of descriptive studies is to describe market characteristics or functions (Malhotra, 1996). Describe is to make complicated things understandable by reducing them to their component parts. Descriptive research could be in direct connection to exploratory research, since researchers might have started off by wanting gain insight to a problem, and after having stated it their research becomes descriptive (Saunders et al., 2000). Explanatory studies establish causal relationship between variables. In these studies the emphasis is on studying a situation or a problem in order to explain the relationships between variables (Saunders et al., 2000).

According to data mining definition, the term data mining describes the process of discovering knowledge from databases stored in data warehouses. The purpose of data mining is to identify valid, novel, useful, and ultimately understandable patterns in data. Data mining is a useful tool, an approach that combines exploration and discovery with confirmatory analysis. Since the focus of this study is data mining, thus the purpose of this research is exploratory.

4.1.2 Research Approach

There are two main research approaches to choose from when conducting research in social science: qualitative and quantitative method (Yin, 1994). There is one significant difference between these two approaches. In the quantitative approach results are based on numbers and statistics that are presented in figures, whereas in qualitative approaches results lies on describing an event with the use of words.

Quantitative research seeks to quantify data and typically apply some form of statistical analysis. Quantitative research approach transform the information to numbers and amounts that later gets analyzed statistically. Quantitative studies tend to be more structured and formulized. It is also characterized by being study few variables on a large number of entities (Holme and Solvang, 1991).
Qualitative research is an unstructured, primarily design based on small samples which is intended to provide some insight and understanding. Qualitative research approach aims at reaching better understanding of the phenomenon being studied, they also tend to be relative flexible using this approach the researcher tries to separate the specific or add and strive to create a completer understanding of the situation (Yin, 1994). Characteristics of qualitative study are that they are based largely on the researcher's own description, emotions and reaction. It is also in closeness of respondents or to the source that data is being collected from (Holme and Slovang, 1991).

Exploration relies more heavily on qualitative techniques but in some cases, there exist sufficient data to allow data mining or exploration of relationships between individual measurements to take place. The concept of data mining allows decision-makers to be supported through exploratory quantitative research (Malhotra and Birks, 2003). The focus of this study is data mining, so the research approach of this research is to be considered a quantitative research.

4.1.3 Research Strategy

Research strategy will be a general plan of how you will go about answering the research question you have set (Sounders et al., 2000). A research strategy is a particular way the researcher wants to collect data. There are a number of approaches for a researcher to make when conducting empirical data collection. Depending on the character of research question, the researcher can choose between an experiment, a survey, history, secondary data analysis (an analysis of archival records) and case study (Yin, 1994).

There are two kinds of data normally used in researches: primary and secondary data. Primary data are originated by a researcher for the specific purpose of addressing the problem at hand. Secondary data are data that have already been collected for purposes other than the problem at hand. Secondary data include data generated within an organization, information made available by business and government sources, commercial marketing research firms and computerized databases. Secondary data can be classified as internal and external. The internal data are those generated within the organization for which the research is being conducted.
and the external data are those generated by sources outside the organization (Malhotra and Birks, 2003). Internal Secondary data were gathered from Parsian bank's database for this study; see the data collection section for more detail. Since the aim of this thesis is data mining and data were collected from bank's databases, the strategy, which suits for this study is a secondary data analysis. Within secondary data exploration, a researcher should start first with an organization's own data archive (Cooper and Schindler, 2003).

In conclusion, the purpose and approach of this study is an exploratory quantitative research and the research strategy is the analysis of secondary data.

### 4.2 Research Process

The objective of this research is to develop a response model in a direct marketing setting. The goal is to predict whether an existing customer will purchase on the next marketing using information provided by the purchase behavior variables. In Figure 4.2, a general overview of the proposed model for predicting a response to the next marketing campaign is presented. We try to predict, using independent variables (RFM variables and demographic), whether a customer will respond to the next marketing campaign or not.

![Figure 4.2: General Overview of Response Model](image)

Response model is a classification model. The task is to classify which customers will respond to a next marketing campaign based on information collected about the customers and divides the customers in to two class respondents and non-respondents. As can be seen in Figure 4.2 the input variables of response model are RFM variables, which show the customer purchase behavior, and some demographic information. Customers with these variables are induced to the model and then model classify them as respondents or non-respondents. Response modeling procedure
consists of several steps and different data mining techniques and algorithms have been used for implementing each step of modeling.

In chapter 2 different studies related to response modeling are reviewed. Based on literature, response modeling overall procedure consists of several steps such as: data preprocessing, feature construction, feature selection, class balancing, classification and model evaluation. As can be seen in related work on response modeling (section 2.3.3) very few articles deal with all these steps. Most of them focus only on two or three steps of modeling and for the rest used the result of previous works such as Ha, et al. (2005), Shin and Cho (2006) and Yu and Cho (2006). The purpose of this study is to focus on all these steps. In this thesis the research process presented in following has been followed. The following process was constructed based on previous methodology (work) on response modeling. Due to nature of this study different steps (components) of modeling were collected from previous work and with some changes and amendments integrated them in to unique process. The schematic of the entire procedure is shown in Figure 4.3. The overall procedure consists of multiple steps such as pre-processing, feature construction, feature selection, classification and evaluation.

![Figure 4.3: Research Process - Overall Procedure](Diagram)

Each Step of above procedure consists of many different courses of actions; in Figure 4.4 the overall procedure in detail is shown.
As can be seen in research process (Figure 4.4) response model procedure consists of several steps. In order to implement the overall research process, an extensive programming was required. The algorithms related to each step were programmed and coded in R Open Source Language Programming. After coding and running each step, they were combined together to make the prediction system.

In the following sections, the detailed of each step and the methods associated to each step are explained.

### 4.3 Data Collection and Description

After having determined the most suitable research strategy, it is necessary to decide on how the empirical data will be collected (Yin, 1994). According to Zikmund (2003) there are two kinds of data normally used in researches: secondary and primary data. Primary data is data that a person gathers on his/her own with a specific purpose in mind while secondary data is data that already has been gathered.
by other researchers with different purpose in mind. Secondary data include data generated within an organization, information made available by business and government sources, commercial marketing research firms and computerized databases. Secondary data can be classified as internal and external. The internal data are those generated within the organization for which the research is being conducted and the external data are those generated by sources outside the organization (Malhotra and Birks, 2003). Customer historical purchase data is collected from organization's database (internal databases) and some additional information such as average household size in a neighborhood, average income, percent of families with a given number of cars, etc are gathered from external databases.

As mentioned above, for data mining purpose secondary data are used. The information at the level of the individual consumer is typically used to construct a response model. Response models are typically built from historical purchase data. For data mining purposes, the purchase history can often be translated into features based on measures of Recency, Frequency and Monetary values (Van den Poel, 2003). This work was mainly based on secondary data. Internal secondary data were gathered from Parsian bank's database for this study. Actually customer's data were gathered from Parsina Bank's databases for building a model. Parsian Bank has about 1,000,000 customers in total and their information stored in database. From 1,000,000 customers about 30,000 customers were randomly selected through IT section and gave it for modeling. Each customer is described by following information. In Table 4.1 the information gathered from Parsian bank is shown. According to literature and close cooperation with domain expert, these features (Table 4.1) were chosen directly from the database tables. These features allowed us to drive and construct all necessary purchase behavior history/RFM variables (Recency, Frequently and Monetary). In addition to customer's data, campaigns data were also collected from marketing section of Parsian Bank.

For conducting this study three types of data were collected from Parsian bank's databases: Customer historical purchase data and demographic, Customer transaction data and Campaigns Data.
1. **Customer historical purchase data and demographic information**

For building a response model, three sets of variables, Recency- Frequency and Monetary, must be used. In order to build these variables, there is a need for customer historical purchase data. Thus customer historical purchase data were collected from Parsian Bank. This data includes such information as an unique ID for each customer, type of accounts (Long term investment deposit accounts, Short termed investment deposit account, Saving account, Current account) that the customer has in bank, number of account, the date that customer open an account, the date that customer close an account, type of services that the customer take (SMS, EMS, ISS, Telephone- Bank), number of services and start date of service. Besides RFM variables some researchers for building a response model use some demographic information. The only demographic data that Parsian bank provided was customer's date of birth.

2. **Customer transaction data**

For building the RFM variables, besides historical purchase data, customer transaction data is also need. Transaction data includes the number of transactions for each account, date of each transaction for each account, type of transaction (Debit or Credit) and amount of money that the customer put in bank, amount of money that the customer draw from bank.

3. **Campaigns Data**

For building target variable in addition to customer history we need campaign information. As I explained above Parsian Bank use mass marketing and send their message through newspaper, radio and television. Campaign data includes the date of the different campaigns (the date that bank send their communication message to the customers) and the content of each campaign.

Historical purchase data, transaction data and demographic information were in the ten text files. Each text file had a unique ID for each customer and required information for each customer. Following table (Table 4.1) shows the information was gathered from Parsian Bank.
Table 4.1: Collected Data from Parsian Bank

<table>
<thead>
<tr>
<th>Historical Purchase Data &amp; Demographic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account's type that the customer has</td>
</tr>
<tr>
<td>(Long term investment deposit account, Short term investment deposit account, Saving account, Current account)</td>
</tr>
<tr>
<td>No of account that the customer has</td>
</tr>
<tr>
<td>Open date of an account</td>
</tr>
<tr>
<td>Close date of an account</td>
</tr>
<tr>
<td>Type of services that the customer take (SMS, EMS, ISS, Telephone- Bank)</td>
</tr>
<tr>
<td>No of services that the customer take</td>
</tr>
<tr>
<td>Start date of service</td>
</tr>
<tr>
<td>Age</td>
</tr>
</tbody>
</table>

(a)

<table>
<thead>
<tr>
<th>Transaction Data</th>
<th>Campaign Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of transaction</td>
<td>Date of different campaigns</td>
</tr>
<tr>
<td>Type of each transaction (Credit, Debit)</td>
<td>Content of each campaign</td>
</tr>
<tr>
<td>Amount of each transaction</td>
<td></td>
</tr>
<tr>
<td>Date of each transaction</td>
<td></td>
</tr>
</tbody>
</table>

(b)

4.4 Data Preparation

Once the data required for the data mining process is collected, it must be in the appropriate format or distribution. Therefore it has to be integrated, cleaned and transformed to meet the requirements of the data mining algorithms. Data preprocessing is an essential step for knowledge discovery and data mining and it is a big issue for data mining. There are a number of data preprocessing techniques: data integration, data cleaning, data transformation and data reduction.

4.4.1 Data Integration and Cleaning

Data integration merges data from multiple sources into a coherent data store. These sources may include multiple databases, data cubes, or flat files. As explained above customer's data were collected from Parsian bank's database were in the ten text files. All the customers had a unique ID, which was the same in all text files. In this step all the ten text files were merged in to one table in database. In this study database was created in SQLite database and ten text files were integrated into one
table in SQLite database. After data was integrated into one table, data has to be cleaned and transformed to meet the requirements of the data mining algorithms.

Data cleaning routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data. As discussed in chapter 3, according to Han and Kamber (2006) there are several strategies to deal with missing values and noisy data.

For this data very little data cleaning was required. Only some variables had missing values. All records had missing values and also all the data which was only an overhead and was not helpful to us in any way was deleted from table. Data cleaning was done in R Languages Program; it can be done also in SQLite browser, as follow:

- Delete all records from table have missing value.
- Delete all those customers form table whose first open account date was after 30/10/1384 and they have no account before 30/10/1384. These customers are not useful for prediction because they have no purchase history in database.
- Delete all customers from table who closed their account.
- Delete all records (customers) from table have negative age.

As discussed above one of the feature was obtained from the bank was the customer's date of birth. Some of the date of birth was record in database erroneously. So when age of the customer was calculated from this wrong data, age become negative. Thus in this step all the customers has a negative age was deleted from table.

In this way, after data cleaning, dataset from 30,000 customers (records) was reduced to 22,427 customers (records).

4.4.2 Normalize/Scale Data

As discussed in chapter 3, data normalization involves scaling the attribute values to make them lie numerically in the same interval/scale, and thus have the same importance. Since Support Vector Machine produce better models when the data is normalized, so all data should be normalized or standardized before classification. As discussed above there are three normalization techniques: Z-score
Normalization, Min-Max Normalization and Normalization by decimal scaling. There is no difference between these three techniques.

For this study Z-score Normalization was used. The data was normalized using the mean and standard deviation. All data have a mean of zero and a standard deviation of 1. For each variable, this was done by subtracting the mean of the variable and dividing by the standard deviation, to arrive at the z-score. Scale function in R program was used for normalization task.

4.5 Feature Construction

After data integration, cleaning and normalization all the necessary variables, dependent and independent variable, for modeling should be constructed.

Parsian bank started its banking operations on the year 1380 (March 2002) and offer different products and services to its customers through different campaigns. For building a model and constructing variables it is very important to decide which campaign should be used for modeling. One might consider constructing a model for each campaign, but according to Potharst et al., (2002) it is not good choice for several reasons: the models are not likely to be of the same quality, since the more recent a campaign, the more historical data are available, also the database has grown over time, information about more customers is available from the more recent campaigns. Therefore for this study based on the campaigns data collected from Parsian bank the most recent campaign was used for construction variable and model. This campaign was from 1/11/1384 through 1/5/1385 and advertised new accounts profit rate and new services. Thus we want to build a response model for period between 1/11/1384 and 1/5/1385. Actually the goal is to predict whether an existing customer will respond to an offer in the observation period time between 1/11/1384 and 1/5/1385 or not, based on the purchase history information (predictor variables). As discussed, this is a classification problem; the dependent variable (target variable) is response (whether or not someone will respond to an offer). All the customers should be categorized as either a respondent (class 1) or as non-respondent (class 0) depending upon whether they made orders during the period between 1/11/1384 and 1/5/1385 or not, respectively. So for developing a response model first the target variable (dependent variable) was
created during the period between 1/11/1384 and 1/5/1385 and then the predictor variables (all necessary purchase behavior history/RFM variables) were calculated for the period between 1380 and 30/10/1384.

In following sections construction of dependent variable (target variable) and independent variables (predictor variables) are explained in more detail.

4.5.1 Target Variable Construction

Response model is a classification (supervised) model. The task is to classify which customers will respond to a next marketing campaign based on information collected about the customers (see Figure 4.2). The first step in the predictive modeling process is to determine the objective variable to be modeled. When building models for targeting direct marketing campaigns, the objective variable is usually based on the customer's historical response data to similar direct marketing campaigns (Cabena et al., 1999). Parsian bank use mass marketing as its strategy for offering a new product or service to customers and they did not have direct marketing campaign and customer's historical response data to direct marketing campaign. So in this thesis for building the target variable customer's historical response data to mass marketing campaigns was used.

The first and critical step in creating the target (dependent) or objective variable is to select the time period in consideration. Setting the objective variable correctly is critical. The size of the time window selected is driven by the time horizon of the desired prediction. For instance, if the marketing campaign to be executed has a six-month window for the customer to respond, the objective variable should be defined with a six-month window (Cabena et al., 1999). Actually it is very important to decide which campaign (time window) should be used for modeling.

As explained above amongst the various mass marketing campaigns that the Parsian had, the most recent campaign (period between 1/11/1384 and 1/5/1385) was selected for building a model and target variable. Purchase information from the period between 1/11/1384 and 1/5/1385 was used as the dependent variable. All the customers should be categorized as either a respondent (class 1) or as non-respondent (class 0) depending upon whether they made purchase during the period between
1/11/1384 and 1/5/1385 or not, respectively. All those customers whose open account date lie between 1/11/1384 and 1/5/1385 were declared as responder and rest of them were non-responder. The target variable in table was assigned a value 1 for responders (if a customer had opened an account during period of 1/11/1384 through 1/5/1385) and value 0 for non-responder.

4.5.2 Predictor Variables/Features Construction

In this research RFM variable and demographic information were used as independent variables. Cullinan (1977) is generally credited for identifying the three sets of variables most often used in response modeling: recency, frequency, and monetary (RFM) (Bauer, 1988; Kestnbaum, 1992). Since then, the literature has provided so many uses of these three variable categories, that there is overwhelming evidence both from academically reviewed studies as well as from practitioners’ experience that the RFM variables are an important set of predictors for response modeling (see section 2.3.2).

As explained in feature construction section (chapter 3) there are various approaches to feature construction. In this study all RFM variables were constructed based on knowledge based approach and literature. Using domain knowledge to construct new features is often recommended because one can thus quickly rule out many impossible combinations and determine an effective measure to evaluate new compound features. Some information on customer historical purchase data was obtained from Parsian bank. This allowed us, in close cooperation with domain experts and guided by the extensive literature, to derive and construct all the necessary purchase behavior variables/RFM variables for a total sample size of 22427 customers.

According to literature various RFM variables were constructed for response modeling. In Appendix 1 various RFM variables were used in response modeling is shown. Based on previous constructed features (Appendix 1) and domain knowledge, Recency, Frequency and Monetary variables for this study were defined as follow: Two time horizons (Hist Horizon and Year Horizon) for constructing all RFM variables were used (Table 4.2). All the variables were measured for this two time period. The Hist horizon refers to the fact that the variable is measured for the period
between 1380 and 30/10/1384. The Year horizon refers to the fact that the variable is measured over the last year (time period between 30/10/1383 and 30/10/1384). Including both time horizons allows us to check whether more recent data are more relevant than historical data.

<table>
<thead>
<tr>
<th>Time Horizon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hist Horizon</td>
<td>Variable is measured for the period between 1380 and 30/10/1384</td>
</tr>
<tr>
<td>Year Horizon</td>
<td>Variable is measured over the last year</td>
</tr>
</tbody>
</table>

For the recency variable, many operationalizations have already been suggested. In this study, the recency variables were defined as:
- number of days since the last open account
- number of days since the first open account
- number of days since the first transaction
- number of days since the last transaction
- number of days since the largest transaction
- number of days since the first service started
- number of days since the last service started

Recency has been found to be inversely related to the probability of the next purchase, i.e., the longer the time delay since the last purchase the lower the probability of a next purchase within the specific period (Cullinan, 1997).

In the context of direct marketing, it has generally been observed that multi-buyers (buyers who already purchased several times) are more likely to repurchase than buyers who only purchased once (Bauer, 1988; Stone, 1984). Although no detailed results are reported because of the proprietary nature of most studies, the frequency variable is generally considered to be the most important of the RFM variable (Nash, 1994). Bauer (1988) suggests operationalizing the frequency variable as the number of purchases divided by the time on the customer list since the first purchase. In this study, the frequency variables were defined as:
- the number of account that the customer has in certain time period (Hist versus Year)
the number of services that the customer take in certain time period (Hist versus Year)

the number of transaction in certain time period (Hist versus Year)

In the direct marketing literature, the general convention is that the more money a person has spent with a company, the higher his/her likelihood of purchasing the next offering (Zahavi and Levin, 1996). Nash (1994) suggests operationalizing monetary value as the highest transaction sale or as the average order size. Zahavi and Levin (1996) propose to use the average amount of money per purchase. For this study the monetary variables were modeled as:

- total money that the customer put in bank (for each account) during a certain time period (Hist versus Year)
- total money that the customer draw from bank (for each account) during a certain time period (Hist versus Year)
- maximum amount of money that the customer put in bank (for each account) during a certain time period (Hist versus Year)
- minimum amount of money that the customer put in bank (for each account) during a certain time period (Hist versus Year)
- average amount of money that the customer put in bank (for each account) during a certain time period (Hist versus Year)

Apart from the RFM variables, demographic information was used as independent variables. The only demographic information used in this research was the age of each customer, which calculated from the given date of birth.

In addition to study the effects of individual independent variables (predictors), also it is important to study their interactions. An interaction effect is a change in the simple main effect of one variable over levels of the second (Candade, 2004). After construction individual independent variables the second step is to construct interactions features. First interactions between the predictor variables should be identified with an appropriate method or model which searches for two-way interactions between candidate predictors, and then after identifying which variables interacted, the pool of candidate predictors is augmented by adding products between the interacting variables. In this study because the appropriate method and
tool for identifying interactions between predictors variable couldn't be found, all the
85 constructed features (Table 5.1) two by two were multiplied and then by means of
F-score selection, in feature selection step, appropriate ones were selected. RFM
variables and their interactions were coded and constructed in R Language Program
and then added to table in SQLite database. In the next section, the feature selection
step and the techniques were used for this step is described.

4.6 Feature Selection

Feature selection is a critical step in response modeling. No matter how
powerful a model is non-relevant input variables lead to poor accuracy. Because
customer related dataset usually contains hundreds of features or variables, many of
which are irrelevant and heavily correlated with others. Without prescreening or
feature selection, they tend to deteriorate performance of the model, as well as
increase the model training time. Feature subset selection can be formulated as an
optimization problem which involves searching the space of possible features to
identify a subset that is optimum or near-optimal with respect to performance
measures such as accuracy (Yang and Honavar, 1998). As stated in feature selection
(section 3.2) various ways exist to perform feature subset selection. Feature selection
can then either be performed as a preprocessing step, independent of the induction
algorithm, or explicitly make use of it. The former approach is termed filter, the latter
wrapper (John et al., 1994). Filter methods operate independently of the target
learning algorithm. Undesirable inputs are filtered out of the data before induction
commences. Wrapper methods make use of the actual target learning algorithm to
evaluate the usefulness of inputs. Typically the input evaluation heuristic that is used
is based upon inspection of the trained parameters and/or comparison of predictive
performance under different input subset configurations. Input selection is then often
performed in a sequential fashion. The backward selection scheme starts from a full
input set and step-wise prunes input variables that are undesirable. The forward
selection scheme starts from the empty input set and step-wise adds input variables
that are desirable.

Feature wrappers often achieve better results than filters due to the fact that
they are tuned to the specific interaction between an induction algorithm and its
training data, but it also tends to be more computationally expensive than the filter
model (Blum and Langley, 1997). When the number of features becomes very large, the filter model is usually chosen due to its computational efficiency. As explained in feature selection (section 3.2) Chen and Lin (2005) combined SVM with various feature selection strategies, filter type method and wrapper type method.

In this study, because the number of constructed features (individual and interaction features) were large, only the wrapper method couldn't be used. So feature selection according to Chen and Lin (2005) was performed by using both the wrapper and filter method. Filter method (F-score) was used as a pre-processing step and then wrapper method (Random Forests – backward elimination) was done for selecting final features as input for model. Thus feature selection was implemented in three steps: 1. the most important interaction features were selected by using F-Score selection, 2. selected interaction features were added to 85 individual features and then fifty important features were selected by using F-score selection, 3. the best subset of features were selected as input for modeling by using Random Forests. In practice, the Random Forests cannot handle too many features, also as mentioned above when the number of features are large wrapper approach tends to be more computationally expensive than the filter model. Thus, before using Random Forests (wrapper approach) to select features, a subset of features was obtained using F-score selection first. So this is a reason why in this study feature selection was implemented in three steps. Following sections describe Random Forests and F-score techniques were used for feature selection step.

### 4.6.1 F-Score

F-score is a simple technique which measures the discrimination of two sets of real numbers. Given training vectors x_k, k = 1, ..., m, if the numbers of positive and negative instances are n+ and n−, respectively, then the F-score of the ith feature is defined as:

\[ F(i) \equiv \frac{\left( \overline{x}_i^{(+)} - \overline{x}_i \right)^2 + \left( \overline{x}_i^{(-)} - \overline{x}_i \right)^2}{\frac{1}{n_{+}-1} \sum_{k=1}^{n_{+}} (x_{k,i}^{(+)}) - \overline{x}_i^{(+)} \right)^2 + \frac{1}{n_{-}-1} \sum_{k=1}^{n_{-}} (x_{k,i}^{(-)} - \overline{x}_i^{(-)} \right)^2} \]
where $\bar{x}_i$, $\bar{x}_i^{(+)}$, $\bar{x}_i^{(-)}$ are the average of the $i$th feature of the whole, positive, and negative data sets, respectively; $x_{k,i}^{(+)}$ is the $i$th feature of the $k$th positive instance, and $x_{k,i}^{(-)}$ is the $i$th feature of the $k$th negative instance. The numerator indicates the discrimination between the positive and negative sets, and the denominator (the sum of variances of the positive and negative sets) indicates the one within each of the two sets. The larger the F-score is, the more likely this feature is more discriminative. Therefore, we use this score as a feature selection criterion (Chen and Lin, 2005).

A disadvantage of F-score is that it does not reveal mutual information among features. Despite this disadvantage, F-score is simple and generally quite effective. In this study F-score was used as a preprocessing step and features with high F-scores were selected and then Random Forests was applied to select final features as input for modeling. F-score was programmed and implemented in R language program.

### 4.6.2 Random Forests

Random forest is an algorithm for classification developed by Leo Breiman (Breiman, 2001) that uses an ensemble of classification trees (Breiman et al., 1984; Hastie et al., 2001). Each of the classification trees is built using a bootstrap sample of the data, and at each split the candidate set of variables is a random subset of the variables. Thus, random forest uses both bagging (bootstrap aggregation), a successful approach for combining unstable learners (Breiman, 1996; Hastie et al., 2001), and random variable selection for tree building. Each tree is unpruned (grown fully), so as to obtain low-bias trees; at the same time, bagging and random variable selection result in low correlation of the individual trees. The algorithm yields an ensemble that can achieve both low bias and low variance (from averaging over a large ensemble of low-bias, high-variance but low correlation trees). Random Forest also provides feature importance (Breiman, 2001). This algorithm returns several measures of variable importance. The most reliable measure is based on the decrease of classification accuracy when values of a variable in a node of a tree are permuted randomly (Breiman, 2001), and this is the measure of variable importance that will be used in this thesis.
The random forest algorithm estimates the importance of a variable by looking at how much prediction error increases (OOB) when data for that variable is permuted while all others are left unchanged. It returns a measure of error rate based on the out-of-bag cases for each fitted tree, the OOB error. Random forest for feature selection uses backward variable elimination approach, which is the wrapper approach, and OOB error as a minimization criterion. The backward elimination begins with all the features in the model and at each step eliminates the feature that contributes least to the discriminatory power of the model. The process stops when all the remaining features meet the criterion to stay in the model (Diaz-Uriarte and Alvarez de Andres, 2005).

In this study Random forest was also used for feature selection. varSelRF and Random Forests packages in R language program were used for variable selection from Random Forest. varSelRF use the OOB error as minimization criterion, carry out variable elimination from random forest, by successively eliminating the least important variables (with importance as returned from random forest). There is little need to fine-tune parameters for varSelRF to achieve excellent performance. There is little need to fine-tune parameters for varSelRF function to achieve excellent performance. The parameters are $c.sd$ (the factor that multiplies the sd. to decide on stopping the iterations or choosing the final solution), $mtryFactor$ (the number of input variables tried at each split), $ntree$ (the number of trees to grow for each forest), $ntreeIterat$ (the number of trees to use for all additional forests) and $vars.drop.frac$ (the fraction of variables, from those in the previous forest, to exclude at each iteration). It has been reported that the default value of these parameters is often a good choice (Liaw and Wiener, 2002). According to Uriarte and Alvarez de Andres (2005) changes in these parameters have in most cases negligible effects and suggested that the default values are often good options for these variables. Thus in this study the default value of these parameters were examined.

### 4.7 Data Sampling for Training and Test

In this step data should be partitioned into training and test sets. It is important to create both a training data set, which is used to build the model, and a test or hold-back data set, which is used to test the model. Because if rely solely on training data, it is very easy to get exceptional results. However, these results would likely not
generalize to new examples, which would be missing from the stored table. Researchers have developed training techniques that reduce the likelihood of “overfitting”, fitting the training data too precisely—usually leads to poor results on new data, to the training data. Still, they are subject to potential problems, whether they specialize too much and overfit the training data or attempt to ensure generality by “underfitting” and not using the training data to its full potential. A strong and effective way to evaluate results is to hide some data and then do a fair comparison of training results to unseen test results. Of course, one could wait until new data arrives during application of the solution, but it is wise to test performance prior to actual application. It prevents unexpected poor results and gives the developers time to extract the best performance from the application system (Ye, 2003). Actually researchers build a classifier using the train set and evaluate it using the test set (see Figure 4.5). They randomly split data into training and test sets. Training is typically done on a large proportion of the total data available, whereas testing is done on some small percentage of the data that has been held out exclusively for this purpose (usually 2/3 for train, 1/3 for test).

![Figure 4.5: Split Data into Train and Test Set](image)

In this thesis for avoiding overfitting and model evaluation, the dataset was partitioned into training and test sets for performance evaluation. A 2/3 of customers were randomly assigned to the training set, which is used to build the model, while the other 1/3 to the test set, which is used to test the model.
4.8 Class Balancing

Many practical classification problems are imbalanced. The class imbalance problem typically occurs when, in a classification problem, there are many more instances of some classes than others. In such cases, standard classifiers tend to be overwhelmed by the large classes and ignore the small ones. Classifier when faced with imbalanced datasets where the number of negative instances far outnumbers the positive instances, the performance drops significantly (see section 3.3.1 for more information).

As explained above (section 3.3.2) various strategies have been proposed to deal with class imbalance problem, such as: increasing the penalty associated with misclassifying the positive class relative to the negative class, over-sampling the majority class and under-sampling the minority class. Among these approaches the under-sampling method has been very popular. As can be seen in section 3.3.3, in literature many researchers under-sample the negative (majority) class for several reasons.

In this study the dataset contains 22,427 customers each of whom is described by 85 individual features and their two way interactions. The response rate is 7% with 1712 respondents and 20715 non-respondents. Also when the dataset was partitioned into training and test set (2/3 for training and 1/3 for test) the training set had 13809 of non-responders and 1143 of responders, which means that the class distribution is highly imbalanced. According to Shin and Chin (2006) SVM do not work very well with unbalanced data sets, so the training set should be balanced. Under-sampling the none-responder class was chosen for this study. Non-respondents were randomly selected twice the numbers of respondents in training set.

The test data set should never be balanced. The test data set represents new data that the models have not yet seen. Certainly, the real world will not balance tomorrow’s data for our classification models; therefore, the test data set itself should not be balanced. Note that all model evaluation will take place using the test data set, so that the evaluative measures will all be applied to unbalanced (real-world-like) data (Larose, 2006).
4.9 Classification

As explained in related work on response modeling (section 2.3.3), various statistical and machine learning methods have been applied in response modeling. Most recent is Support Vector Machine (SVM) that has been spotlighted in the machine learning community also offer advantages over multivariate classifiers. SVMs show distinct advantages such as better generalization, increased speed of learning, ability to find a global optimum and ability to deal with linearly non-separable data.

In this study support vector machine (SVM) was used as a classifier for classification. SVM requires a certain amount of model selection. As explained above (section 3.4) there are only three common kernels: linear, polynomial and radial basis function (RBF). First, it is important to decide which one must be used for modeling. Then the penalty parameter C and kernel parameters are chosen.

4.9.1 RBF Kernel

A RBF kernel was used for SVM in this study. This was chosen based on previous work, also Hsu et al., (2003) suggested that in general Radial Basis Function (RBF) is a reasonable first choice. The RBF kernel nonlinearly maps samples into a higher dimensional space, so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear. Furthermore, the linear kernel is a special case of RBF as Keerthi and Lin (2003) showed that the linear kernel with a penalty parameter $\tilde{C}$ has the same performance as the RBF kernel with some parameters (cost & gamma). The second reason is the number of hyperparameters which influences the complexity of model selection. The polynomial kernel has more hyperparameters than the RBF kernel. Finally, the RBF kernel has less numerical difficulties than the other kernels.

After selection the RBF kernel for SVM, the best parameters have to be chosen. In this study the best parameters of RBF kernels were chosen by cross-validation procedure and grid search.
4.9.2 Cross-validation and Grid-search

There are two parameters while using RBF kernels: C (cost) and $\gamma$ (gamma). It is not known beforehand which C and $\gamma$ are the best for one problem; consequently some kind of model selection (parameter search) must be done. The goal is to identify good $(C, \gamma)$ so that the classifier can accurately predict unknown data (i.e., testing data). Note that it may not be useful to achieve high training accuracy (i.e., classifiers accurately predict training data whose class labels are indeed known). Therefore, a common way is to separate training data to two parts of which one is considered unknown in training the classifier. Then the prediction accuracy on this set can more precisely reflect the performance on classifying unknown data. An improved version of this procedure is cross-validation. In k-fold cross-validation, first the training set is divided into k subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining k-1 subsets. Thus, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly classified (see Figure 4.6). The cross-validation procedure can prevent the overfitting problem (Hsu et al., 2003).

It is recommend a “grid-search” on C and $\gamma$ using cross-validation. Basically pairs of $(C, \gamma)$ are tried and the one with the best cross-validation accuracy is picked. It is found that trying exponentially growing sequences of C and $\gamma$ is a practical method to identify good parameters. Users then use the best parameter to train the whole training set and generate the final model (Chang and Lin, 2006). Thus in this study 5 fold cross-validation via grid search was used for parameter selection $(C, \gamma)$. Different pairs of $(C, \gamma)$ were tried and the one with the best cross-validation accuracy was picked. For implementing SVM in R Language Program, SVM function in e1071 package was used.
4.10 Model Evaluation

The performance of a model do not evaluated on the training set, because it has had a hand in creating the model and so will overstate the model’s accuracy. The model’s accuracy always measure on a test set that is drawn from the same population as the training, but has not been used in any way to create the model. Thus in this study as explained above the dataset were divided in to train and test set (2/3 for train set and 1/3 for test set). Train set was used for model building and test set for measure the model's accuracy. In this study the performance of SVM response model was measured by using Accuracy, Error Rate, True Positive Rate, True Negative rate, Weighted Accuracy and Lift Chart (see section 3.5 for more information).

As explained in section 3.5 the most basic performance measures can be described by a confusion matrix. The accuracy of a classifier can be described by a confusion matrix. Accuracy alone is not an adequate measure of performance especially in this case where the number of negative cases (non-respondent class) is much greater than the number of positive cases (respondent class). Thus Weighted Accuracy with $\lambda = 9$, True Positive Rate, True Negative rate, and Lift Chart were also used for model evaluation (see section 3.5).
Chapter 5

Results and Analysis

5. Results and Analysis

The process analysis was performed with the R program (http://www.r-project.org; R Development Core Team, 2004). For implementation of each step of response modeling different packages were installed and used. This chapter begins with a brief introduction to R software and packages were used in each step of modeling, followed by results of each step of modeling, which consists of result of data cleaning and integration, feature construction, feature selection and balancing the class distribution. Then the prediction result is presented. Finally the model evaluation and analysis and features importance is also explained.
5.1 R Software and Packages

All the analysis was carried out with R (http://www.r-project.org; R Development Core Team, 2004). R is a language and environment for statistical computing and graphics. R provides a wide variety of statistical (linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, clustering ...) and graphical techniques, and is highly extensible. One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control. R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (Linux), Windows and MacOS. R can be extended (easily) via packages. There are about eight packages supplied with the R distribution and many more are available through the CRAN family of internet sites covering a very wide range of modern statistics.

In this study for implementation each step of modeling different packages were installed and used. A list of all packages were download from R website and used for each step of modeling with their applications is shown in table 5.1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSQLite (James, 2006)</td>
<td>Connecting to SQLite Database</td>
</tr>
<tr>
<td>Random Forests (Liaw, 2005)</td>
<td>Feature selection using Random Forest</td>
</tr>
<tr>
<td>varSelRF (Uriarte, 2005)</td>
<td>Feature selection using Random Forest</td>
</tr>
<tr>
<td>e1071 (Dimitriadou et al., 2006)</td>
<td>SVM Implementation</td>
</tr>
<tr>
<td>ROCR (Sing et al., 2005)</td>
<td>Plotting Lift Chart</td>
</tr>
</tbody>
</table>

5.1 Data Preprocessing

As explained in data collection (section 4.3) information about 30,000 customers from 1,000,000 customers was obtained from Parsian Bank. Customers were randomly selected from database. Three different types of information were
collected: customer historical purchase data, customer transaction data and campaign data (Table 4.1). This information allowed us, in close cooperation with domain expert and guided by the extensive literature to drive all the necessary purchase behavior variables (RFM variables) for a total sample size 30,000 customers. Gathered data was in the ten text files. For data analysis and modeling purpose all these ten text files were integrated in to one table in SQLite database. After integration, data has to be cleaned and normalized for data mining purpose. For this data very little data cleaning was required. Only some variables had missing values. All the data which was only an over head and was not helpful for modeling and also records with missing values were deleted from table. Data cleaning (reduction) was done in R Languages Program as follow:

- Delete all records have missing value.
- Delete all those customers whose first open account date was after 30/10/1384 and there have no account before 30/10/1384. Because these customers are not useful for prediction because they have no purchase history in data base.
- Delete all customers who closed their account.
- Delete all records (customers) have negative age.

At first there were 30000 customers in dataset. After running all those queries, all the data which was only an over head and was not helpful for modeling in any way was deleted from table. In this way, after data cleaning, dataset was reduced to 22,427 customers.

Also, the data should be normalized before presenting it to the SVM to ease mathematical calculations as well as reduce the effect of larger attributes. So the Z-score normalization was used for normalizing data. After Z-score normalization all data have a mean of zero and a standard deviation of 1. For each variable, this was done by subtracting the mean of the variable and dividing by the standard deviation Scale function in R was used for this purpose.

5.2 Results of Feature Construction

After data preprocessing (integrating, cleaning and normalizing), target variable (dependent variable) and all necessary purchase behavior/RFM variables (independent variables) for modeling, as explained in section 4.5, were constructed
from raw data (Table 4.1) gathered from Parsin Bank. Based on domain knowledge and literature (previous constructed features) 85 predictor features and target variable were constructed from gathered information. These features are listed in Table 5.2. A detailed description of feature construction was presented in research methodology chapter (see section 4.5).

As stated earlier, it is important to study both the effects of individual independents as well as their interactions. An interaction effect is a change in the simple main effect of one variable over levels of the second (Candade, 2004). After construction individual independent variables the second step is to construct interactions features. First interactions between the predictor variables should be identified with an appropriate method or model which searches for two-way interactions between candidate predictors, and then after identifying which variables interacted, the pool of candidate predictors is augmented by adding products between the interacting variables. In this study because the appropriate method for identifying interaction between predictors variable couldn't be found, all the 85 constructed features (Table 5.2) two by two were multiplied and then in the next step with the feature selection technique appropriate ones were selected. After selection all possible two way interactions were included to test for their significance. RFM variables and their interaction were coded and constructed in R Language Program and then added to table in SQLite database.

After construction all the necessary purchase behavior variable/RFM and their interaction, it is important to select the best subset of features for modeling. As explained in research methodology chapter (section 4.6), for this study feature selection was performed in three steps using both the wrapper and filter method. The following sections describe these steps and also show the result of each step.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Target</strong></td>
<td>Dependent variable</td>
</tr>
<tr>
<td><strong>Age</strong></td>
<td>Demographic Information</td>
</tr>
<tr>
<td><strong>Recency</strong></td>
<td></td>
</tr>
<tr>
<td>recencyfirstaccount</td>
<td>No. of days since first open account</td>
</tr>
<tr>
<td>recencylastaccount</td>
<td>No. of days since last open account</td>
</tr>
<tr>
<td>recencyfirstservice</td>
<td>No. of days since first start services</td>
</tr>
<tr>
<td>recencylastservice</td>
<td>No. of days since last start services</td>
</tr>
<tr>
<td>recencylasttran</td>
<td>No. of days since last transaction</td>
</tr>
<tr>
<td>recencymaxtrandate</td>
<td>No. of days since maximum transaction</td>
</tr>
<tr>
<td><strong>Frequency</strong></td>
<td></td>
</tr>
<tr>
<td>Current</td>
<td>Life to date No. of current account</td>
</tr>
<tr>
<td>saving</td>
<td>Life to date No. of saving account</td>
</tr>
<tr>
<td>long</td>
<td>Life to date No. of Long termed investment deposit account</td>
</tr>
<tr>
<td>short</td>
<td>Life to date No. of short termed investment deposit account</td>
</tr>
<tr>
<td>totalacc</td>
<td>Total accounts</td>
</tr>
<tr>
<td>telephonebank</td>
<td>Life to date No. of Telephonebank</td>
</tr>
<tr>
<td>sms</td>
<td>Life to date No. of SMS</td>
</tr>
<tr>
<td>ems</td>
<td>Life to date No. of EMS</td>
</tr>
<tr>
<td>ISS</td>
<td>Life to date No. of ISS</td>
</tr>
<tr>
<td>totalservice</td>
<td>Life to Date Total services</td>
</tr>
<tr>
<td>currentlastyear</td>
<td>No. of current account over last year</td>
</tr>
<tr>
<td>savinglastyear</td>
<td>No. of saving account over last year</td>
</tr>
<tr>
<td>longlastyear</td>
<td>No. of Long termed investment deposit account over last year</td>
</tr>
<tr>
<td>shortlastyear</td>
<td>No. of short termed investment deposit account over last year</td>
</tr>
<tr>
<td>totalacclastyear</td>
<td>Total account over last year</td>
</tr>
<tr>
<td>telephonbanklastyear</td>
<td>No. of telephonbank over last year</td>
</tr>
<tr>
<td>smslastyear</td>
<td>No. of sms over last year</td>
</tr>
<tr>
<td>emslastyear</td>
<td>No. of ems over last year</td>
</tr>
<tr>
<td>ISSlastyear</td>
<td>No. of ISS over last year</td>
</tr>
<tr>
<td>totalservicelastyear</td>
<td>Total service over last year</td>
</tr>
<tr>
<td>totalaccserv</td>
<td>Total service and account</td>
</tr>
<tr>
<td>totalaccservlastyear</td>
<td>Total service and account over last year</td>
</tr>
<tr>
<td>notrancurrent</td>
<td>No. of transaction for current account</td>
</tr>
<tr>
<td>notransaving</td>
<td>No. of transaction for saving account</td>
</tr>
<tr>
<td>notranlong</td>
<td>No. of transaction for Long termed investment deposit account</td>
</tr>
<tr>
<td>notranshort</td>
<td>No. of transaction for short termed investment deposit account</td>
</tr>
<tr>
<td>notrantotal</td>
<td>Life to Date total transaction</td>
</tr>
<tr>
<td>notrancurrentlast</td>
<td>No. of transaction over last year for current account</td>
</tr>
<tr>
<td>notransavinglast</td>
<td>No. of transaction over last year for saving account</td>
</tr>
<tr>
<td>notranlonglast</td>
<td>No. of transaction over last year for Long termed account</td>
</tr>
<tr>
<td>notranshortlast</td>
<td>No. of transaction over last year for short termed account</td>
</tr>
<tr>
<td>notrantotallast</td>
<td>Total transaction over last year</td>
</tr>
<tr>
<td><strong>Monetary</strong></td>
<td></td>
</tr>
<tr>
<td>totaltrancurrent</td>
<td>Total money that a customer put in current account</td>
</tr>
<tr>
<td>totaltransaving</td>
<td>Total money that a customer put in saving account</td>
</tr>
<tr>
<td>totaltranlong</td>
<td>Total money that a customer put in Long termed account</td>
</tr>
<tr>
<td>totaltranshort</td>
<td>Total money that a customer put in short termed account</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>totaltran</td>
<td>Total money that a customer put in all accounts - last year</td>
</tr>
<tr>
<td>trancurrentlast</td>
<td>Total money that a customer put in current account-last year</td>
</tr>
<tr>
<td>transavinglast</td>
<td>Total money that a customer put in saving account-last year</td>
</tr>
<tr>
<td>tranlonglast</td>
<td>Total money that a customer put in long termed account-last year</td>
</tr>
<tr>
<td>transhortlast</td>
<td>Total money that a customer put in short termed account-last year</td>
</tr>
<tr>
<td>mintrancurrent</td>
<td>Minimum money that a customer put in current account</td>
</tr>
<tr>
<td>mintransaving</td>
<td>Minimum money that a customer put in saving account</td>
</tr>
<tr>
<td>mintranlong</td>
<td>Minimum money that a customer put in long termed account</td>
</tr>
<tr>
<td>mintranshort</td>
<td>Minimum money that a customer put in short termed account</td>
</tr>
<tr>
<td>mintrancurrentlast</td>
<td>Minimum money that a customer put in current account-last year</td>
</tr>
<tr>
<td>mintransavinglast</td>
<td>Minimum money that a customer put in saving account-last year</td>
</tr>
<tr>
<td>mintranlonglast</td>
<td>Minimum money that a customer put in long termed account-last year</td>
</tr>
<tr>
<td>mintranshortlast</td>
<td>Minimum money that a customer put in short termed account-last year</td>
</tr>
<tr>
<td>maxtrancurrent</td>
<td>Maximum money that a customer put in current account</td>
</tr>
<tr>
<td>maxtransaving</td>
<td>Maximum money that a customer put in saving account</td>
</tr>
<tr>
<td>maxtranlong</td>
<td>Maximum money that a customer put in long termed account</td>
</tr>
<tr>
<td>maxtranshort</td>
<td>Maximum money that a customer put in short termed account</td>
</tr>
<tr>
<td>maxtrantotal</td>
<td>Maximum money that a customer put in all accounts</td>
</tr>
<tr>
<td>maxtrancurrentlast</td>
<td>Maximum money that a customer put in current account-last year</td>
</tr>
<tr>
<td>maxtransavinglast</td>
<td>Maximum money that a customer put in saving account-last year</td>
</tr>
<tr>
<td>maxtranlonglast</td>
<td>Maximum money that a customer put in long termed account-last year</td>
</tr>
<tr>
<td>maxtranshortlast</td>
<td>Maximum money that a customer put in short termed account-last year</td>
</tr>
<tr>
<td>avgrancurrent</td>
<td>Average money that a customer put in current account</td>
</tr>
<tr>
<td>avgransaving</td>
<td>Average money that a customer put in saving account</td>
</tr>
<tr>
<td>avgranlong</td>
<td>Average money that a customer put in long termed account</td>
</tr>
<tr>
<td>avgranshort</td>
<td>Average money that a customer put in short termed account</td>
</tr>
<tr>
<td>avgrancurrentlast</td>
<td>Average money that a customer put in current account-last year</td>
</tr>
<tr>
<td>avgransavinglast</td>
<td>Average money that a customer put in saving account-last year</td>
</tr>
<tr>
<td>avgranlonglast</td>
<td>Average money that a customer put in long termed account-last year</td>
</tr>
<tr>
<td>avgranshortlast</td>
<td>Average money that a customer put in short termed account-last year</td>
</tr>
<tr>
<td>totaldepositcurrent</td>
<td>Total money that a customer withdrawal from current account</td>
</tr>
<tr>
<td>totaldepositsaving</td>
<td>Total money that a customer withdrawal from saving account</td>
</tr>
<tr>
<td>totaldepositlong</td>
<td>Total money that a customer withdrawal from Long termed account</td>
</tr>
<tr>
<td>totaldepositshort</td>
<td>Total money that a customer withdrawal from short termed account</td>
</tr>
<tr>
<td>totaldepositsavinglast</td>
<td>Total money that a customer withdrawal from saving account-last year</td>
</tr>
<tr>
<td>totaldepositlonglast</td>
<td>Total money that a customer withdrawal from long termed account-last year</td>
</tr>
<tr>
<td>totaldepositshortlast</td>
<td>Total money that a customer withdrawal from short termed account-last year</td>
</tr>
<tr>
<td>totaldeposittotal</td>
<td>Total money that a customer withdrawal from all accounts</td>
</tr>
<tr>
<td>totaldepositsavecurrentlast</td>
<td>Total money that a customer withdrawal from current account-last year</td>
</tr>
<tr>
<td>totaldepositsavinglast</td>
<td>Total money that a customer withdrawal from saving account-last year</td>
</tr>
<tr>
<td>totaldepositlonglast</td>
<td>Total money that a customer withdrawal from long termed account-last year</td>
</tr>
<tr>
<td>totaldepositshortlast</td>
<td>Total money that a customer withdrawal from short termed account-last year</td>
</tr>
<tr>
<td>totaldepositalast</td>
<td>Total money that a customer withdrawal from all accounts-last year</td>
</tr>
</tbody>
</table>

Total transaction
5.3 Results of Feature Selection

SVM was selected as classifier for this study. Support Vector Machine (SVM) (Cortes and Vapnik 1995) is an effective classification method, but it does not directly obtain the feature importance. Also when the numbers of features become large the performance of SVM's model drops significantly also the model training time increase. Actually feature subset selection or determination of the predictor input variables is the main issue that affects the performance of SVM's model. As the numbers of constructed features (predictor variables and their interaction) for this study were large, so the best subset of features must be selected for SVM modeling.

In this study feature selection was performed in three steps using both wrapper and filter method, because when the number of features is large the wrapper approach can not handle too many features so the filter model is usually chosen as a preprocessing step due to its computational efficiency. F-score and Random Forest were chosen for this purpose, where F-score is a filter to select features and Random Forests is a wrapper. Before using Random Forests to select best subset of features, we obtain a subset of features using F-score selection first. These techniques are explained in chapter 4.

Feature selection was done in three steps as follow:

Step1. Selection of appropriate interaction features using F-score
Step2. Selection of appropriate features using F-score
Step3. Selection of best subset of features for modeling using Random Forests

The description and result of each step is presented in following sections.
5.3.1 Step1: Selection of interaction features using F-score

In the first step F-score, which is the filter method, was used to select the most significant interaction features. As explained above, in addition to construction all necessary purchase behavior variables (RFM variables) from raw data, interaction between these variables must be constructed for modeling. Because the appropriate tool or algorithm for finding interaction between predictor variables couldn't be found, all the 85 features two by two were multiplied. As the numbers of interaction features were large, the most appropriate ones were selected by means of F-score selection. While constructing two-way interactions of independents variables, F-score for each interaction were calculated. After calculation F-score for each interaction, features were ranked in descending order. Then the first 20 features from interaction features with high F-score were selected. These selected features with its F-score are listed in Table 5.3. In following table 20 selected features are ranked in descending order based on the F-score.

In order to perform F-score, F-score algorithm (formula) was programmed and implemented in R Language Program. F-score method is explained in detail in section 4.6.1.
5.3.2 Step 2: Selection of features using F-score

After selection of 20 interactions with high F-score, these selected features were added to the individual independent variables, so the numbers of predictor variables became 105 (constructed features = 85 and selected interaction features = 20). As the number of features (105) were large and the RF (Random Forests) code was used cannot handle too many features, thus before using RF to select best subset of features for modeling, F-score was used again to obtain a subset of features. Actually F-score was used as a preprocessing step. F-score for every feature (105 features) was calculated. After calculation F-score for each feature, features were sort in descending order, 50 from 105 features with high F-score were selected. This threshold was selected because in literature indicates that when the number of feature larger than 50 Random Forests can not end in appropriate time. These selected features are shown in Table 5.4. In following table 50 selected features are ranked in descending order based on the F-score.

<table>
<thead>
<tr>
<th>No.</th>
<th>Name</th>
<th>Formulation</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Inter 17-14</td>
<td>longlastyear * recencylastaccount</td>
<td>1.045098e-01</td>
</tr>
<tr>
<td>2</td>
<td>Inter 74-14</td>
<td>notranlonglast * recencylastaccount</td>
<td>9.091451e-02</td>
</tr>
<tr>
<td>3</td>
<td>Inter 58-14</td>
<td>notranlong * recencylastaccount</td>
<td>9.091451e-02</td>
</tr>
<tr>
<td>4</td>
<td>Inter 16-14</td>
<td>savinglastyear * recencylastaccount</td>
<td>7.156240e-02</td>
</tr>
<tr>
<td>5</td>
<td>Inter 55-17</td>
<td>recencylasttran * longlastyear</td>
<td>6.943128e-02</td>
</tr>
<tr>
<td>6</td>
<td>Inter 55-19</td>
<td>recencylasttran * totalacclastyear</td>
<td>6.845243e-02</td>
</tr>
<tr>
<td>7</td>
<td>Inter 55-6</td>
<td>recencylasttran * totalacc</td>
<td>6.591241e-02</td>
</tr>
<tr>
<td>8</td>
<td>Inter 14-4</td>
<td>recencylastaccount * long</td>
<td>6.553327e-02</td>
</tr>
<tr>
<td>9</td>
<td>Inter 55-4</td>
<td>recencylasttran * long</td>
<td>6.454754e-02</td>
</tr>
<tr>
<td>10</td>
<td>Inter 14-6</td>
<td>recencylastaccount * totalacc</td>
<td>6.262577e-02</td>
</tr>
<tr>
<td>11</td>
<td>Inter 74-55</td>
<td>notranlonglast * recencylasttran</td>
<td>5.539086e-02</td>
</tr>
<tr>
<td>12</td>
<td>Inter 58-55</td>
<td>notranlong * recencylasttran</td>
<td>5.467339e-02</td>
</tr>
<tr>
<td>13</td>
<td>Inter 27-14</td>
<td>totalacceserv * recencylastaccount</td>
<td>5.539086e-02</td>
</tr>
<tr>
<td>14</td>
<td>Inter 19-14</td>
<td>totalacclastyear * recencylastaccount</td>
<td>5.009501e-02</td>
</tr>
<tr>
<td>15</td>
<td>Inter 55-27</td>
<td>recencylasttran * totalacceserv</td>
<td>4.971132e-02</td>
</tr>
<tr>
<td>16</td>
<td>Inter 55-28</td>
<td>recencylasttran * totalacceservlastyear</td>
<td>4.926261e-02</td>
</tr>
<tr>
<td>17</td>
<td>Inter 28-14</td>
<td>totalacceservlastyear*recencylastaccount</td>
<td>4.394849e-02</td>
</tr>
<tr>
<td>18</td>
<td>Inter 87-55</td>
<td>recencymaxtrandate * recencylasttran</td>
<td>4.297667e-02</td>
</tr>
<tr>
<td>19</td>
<td>Inter 17-3</td>
<td>longlastyear * saving</td>
<td>4.084409e-02</td>
</tr>
<tr>
<td>20</td>
<td>Inter 19-16</td>
<td>totalacclastyear * saving</td>
<td>4.142787e-02</td>
</tr>
</tbody>
</table>
Table 5.4: Fifty Features Selected Using F-score

<table>
<thead>
<tr>
<th>No.</th>
<th>Name</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>longlastyear</td>
<td>2.071100e-01</td>
</tr>
<tr>
<td>2</td>
<td>Long</td>
<td>1.735767e-01</td>
</tr>
<tr>
<td>3</td>
<td>Totalacclastyear</td>
<td>1.726518e-01</td>
</tr>
<tr>
<td>4</td>
<td>notranlonglast</td>
<td>1.659363e-01</td>
</tr>
<tr>
<td>5</td>
<td>notranlong</td>
<td>1.659363e-01</td>
</tr>
<tr>
<td>6</td>
<td>totalacc</td>
<td>1.622260e-01</td>
</tr>
<tr>
<td>7</td>
<td>totalaccservlastyear</td>
<td>1.558499e-01</td>
</tr>
<tr>
<td>8</td>
<td>totalaccserv</td>
<td>1.494970e-01</td>
</tr>
<tr>
<td>9</td>
<td>inter17_14</td>
<td>1.478763e-01</td>
</tr>
<tr>
<td>10</td>
<td>saving</td>
<td>1.227522e-01</td>
</tr>
<tr>
<td>11</td>
<td>short</td>
<td>1.153240e-01</td>
</tr>
<tr>
<td>12</td>
<td>recencylasttran</td>
<td>1.086860e-01</td>
</tr>
<tr>
<td>13</td>
<td>inter87_55</td>
<td>8.372178e-02</td>
</tr>
<tr>
<td>14</td>
<td>inter14_4</td>
<td>7.989060e-02</td>
</tr>
<tr>
<td>15</td>
<td>inter74_14</td>
<td>7.823616e-02</td>
</tr>
<tr>
<td>16</td>
<td>inter58_14</td>
<td>7.823616e-02</td>
</tr>
<tr>
<td>17</td>
<td>inter55_6</td>
<td>7.665500e-02</td>
</tr>
<tr>
<td>18</td>
<td>recencylastaccount</td>
<td>7.522515e-02</td>
</tr>
<tr>
<td>19</td>
<td>notranshortlast</td>
<td>6.646031e-02</td>
</tr>
<tr>
<td>20</td>
<td>notranshort</td>
<td>6.644864e-02</td>
</tr>
<tr>
<td>21</td>
<td>inter55_27</td>
<td>6.583486e-02</td>
</tr>
<tr>
<td>22</td>
<td>inter16_14</td>
<td>6.351645e-02</td>
</tr>
<tr>
<td>23</td>
<td>savinglastyear</td>
<td>5.457700e-02</td>
</tr>
<tr>
<td>24</td>
<td>inter55_19</td>
<td>4.659479e-02</td>
</tr>
<tr>
<td>25</td>
<td>notrantotallast</td>
<td>4.326735e-02</td>
</tr>
<tr>
<td>26</td>
<td>notrantotal</td>
<td>4.326067e-02</td>
</tr>
<tr>
<td>27</td>
<td>shortlastyear</td>
<td>3.960566e-02</td>
</tr>
<tr>
<td>28</td>
<td>inter19_14</td>
<td>3.828196e-02</td>
</tr>
<tr>
<td>29</td>
<td>inter55_28</td>
<td>3.701948e-02</td>
</tr>
<tr>
<td>30</td>
<td>totaltranlong</td>
<td>3.369124e-02</td>
</tr>
<tr>
<td>31</td>
<td>totaltranlonglast</td>
<td>3.369124e-02</td>
</tr>
<tr>
<td>32</td>
<td>avgtranlonglast</td>
<td>3.221448e-02</td>
</tr>
<tr>
<td>33</td>
<td>avgtranlong</td>
<td>3.221448e-02</td>
</tr>
<tr>
<td>34</td>
<td>maxtranlonglast</td>
<td>3.124999e-02</td>
</tr>
<tr>
<td>35</td>
<td>maxtranlong</td>
<td>3.124999e-02</td>
</tr>
<tr>
<td>36</td>
<td>mintranlonglast</td>
<td>2.896750e-02</td>
</tr>
<tr>
<td>37</td>
<td>mintranlong</td>
<td>2.896750e-02</td>
</tr>
<tr>
<td>38</td>
<td>inter28_14</td>
<td>2.517619e-02</td>
</tr>
<tr>
<td>39</td>
<td>age</td>
<td>1.319582e-02</td>
</tr>
<tr>
<td>40</td>
<td>recencymaxtranrate</td>
<td>1.315932e-02</td>
</tr>
<tr>
<td>41</td>
<td>inter17_3</td>
<td>1.230620e-02</td>
</tr>
<tr>
<td>42</td>
<td>totaltranshort</td>
<td>1.022598e-02</td>
</tr>
<tr>
<td>43</td>
<td>totaltranshortlast</td>
<td>1.022598e-02</td>
</tr>
<tr>
<td>44</td>
<td>totaltran</td>
<td>8.445647e-03</td>
</tr>
<tr>
<td>45</td>
<td>totaltranlast</td>
<td>8.445647e-03</td>
</tr>
<tr>
<td>46</td>
<td>totaldepositshortlast</td>
<td>8.241015e-03</td>
</tr>
<tr>
<td>47</td>
<td>totaldepositshort</td>
<td>8.241015e-03</td>
</tr>
<tr>
<td>48</td>
<td>totalservicelastyear</td>
<td>7.947018e-03</td>
</tr>
<tr>
<td>49</td>
<td>totalservice</td>
<td>7.176597e-03</td>
</tr>
<tr>
<td>50</td>
<td>avgtranshortlast</td>
<td>7.085188e-03</td>
</tr>
</tbody>
</table>
5.3.3 Step3: Selection of best subset of features using Random Forests

After selection of 50 features from 105 features, Random forest (RF) was performed to select best subset features for modeling. Random forest (RF) is a classification method, but it also provides feature importance (Breiman, 2001). This algorithm returns several measures of variable importance. The most reliable measure is based on the decrease of classification accuracy when values of a variable in a node of a tree are permuted randomly (Breiman, 2001), and this is the measure of variable importance that was used in this thesis. See section 4.6.2 for more information.

The random forest algorithm estimates the importance of a variable by looking at how much prediction error increases when (OOB) data for that variable is permuted while all others are left unchanged. Random forest for feature selection use backward variable elimination approach, which is the wrapper approach. The backward elimination begins with all the features in the model and at each step eliminates the feature that contributes least to the discriminatory power of the model. The process stops when all the remaining features meet the criterion to stay in the model.

In the final step of feature selection, Random Forests was performed for selection of best subset of features for modeling with 50 features selected from previous step (step 2). varSelRF and Random Forests package in R were used for implementing Random Forest. varSelRF function in this package select variable from random forests using backwards variable elimination approach. Also use the OOB error as minimization criterion, carry out variable elimination from random forest, by successively eliminating the least important variables (with importance as returned from random forest).

There is little need to fine-tune parameters for varSelRF function to achieve excellent performance. The parameters are c.sd (the factor that multiplies the sd. to decide on stopping the iterations or choosing the final solution), mtryFactor (the number of input variables tried at each split), ntree (the number of trees to grow for each forest), ntreeIterat (the number of trees to use for all additional forests) and
vars.drop.frac (the fraction of variables, from those in the previous forest, to exclude at each iteration). It has been reported that the default value of these parameters is often a good choice (Liaw and Wiener, 2002). The default value of vars.drop.frac is 0.2 and c.sd is 0 and 1. Setting c.sd = 0 is selecting the set of variables that leads to the smallest error rate and setting c.sd = 1 can lead to solutions with fewer variables than selecting the solution with the smallest error rate, while achieving an error rate that is not different, within sampling error, from the “best solution”. Literature shows that there are no relevant differences in error rate whether we use the c.sd = 0 and c.sd=1 (Uriarte and Alvarez de Andres, 2005). Also the default values of mtryFactor, ntree and ntreeIterat parameter are 1, 2000 and 1000. According to Uriarte and Alvarez de Andres (2005) changes in these parameters have in most cases negligible effects and suggested that the default values are often good options for these variables. Thus in this study the default value of these parameters were examined.

varSelRF (Variable selection from random forests using OOB error) in R language program in R language program was performed with 50 features and with the default parameters (mtryFactor = 1, c.sd = 1, ntree = 2000, ntreeIterat= 1000 and vars.drop.frac = 0.2). With the default parameters, all forests were examined that result from eliminating, iteratively, a fraction, vars.drop.frac, of the least important variables used in the previous iteration. By default, vars.frac.drop = 0.2 which allows for relatively fast operation, is coherent with the idea of an “aggressive variable selection” approach, and increases the resolution as the number of variables considered becomes smaller. By default, we did not recalculate variable importances at each step (recompute.var.imp = FALSE) as Svetnik et al. 2004 mention severe overfitting resulting from recalculating variable importance. After fitting all forests, the OOB error rates from all the fitted random forests were examined. We choose the solution with the fewer variables than selecting the solution with the smallest error rate, while achieving an error rate that is not different, within sampling error, from the “best solution”. Actually c.sd = 1 was examined (tends to result in smaller sets of selected variables). The values of mtryFactor, ntreeIterat and ntree parameter were set to 1, 1000 and 2000. Value 2000 for ntree seems quite adequate, with further increases having negligible effects. Also default setting of mtryFactor = 1 was a good choice in terms of OOB error rate.
After performing varSelRF with 50 features and with these parameters: mtryFactor = 1, c.sd = 1, ntree = 2000, ntreeIterat= 1000 and vars.drop.frac = 0.2, the best subset of features with 26 important features was selected. The OOB estimate of error rate of this selected set of variables was 7.81%. The first forest (before any variable selection) was fitted with all 50 features and iteratively, a fraction of the least important variables used in the previous iteration was eliminated. The initial OOB error (the OOB estimate of error rate of first forest) was 7.65%. The numbers of variables were in the forest at that stage (iteration) and the OOB error rate of all fitted forest are shown in Table 5.5. Variables were selected at each iteration are illustrated in Appendix 2.

Table 5.5: Random Forest result

<table>
<thead>
<tr>
<th>Iteration</th>
<th>No. of Variables</th>
<th>OOB error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td>mean = 0.0777, sd = 0.0018</td>
</tr>
<tr>
<td>2</td>
<td>32</td>
<td>mean = 0.0772; sd = 0.0018</td>
</tr>
<tr>
<td>3</td>
<td>26</td>
<td>mean = 0.0781; sd = 0.0018</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
<td>mean = 0.0788; sd = 0.0018</td>
</tr>
</tbody>
</table>

Based on the "c.sd = 1 rule" was used, the random forest select the 26 features with 7.81% OOB error rate as the best subset of feature. These selected features are listed in Table 5.6. These 26 features were used as input variables for SVM modeling.
Table 5.6: Selected Features using Backwards Elimination on Random Forest
(Input Variables)
(Parameters used: mtry = 1, ntree = 2000, ntreeterat= 1000, vars.drop.frac = 0.2, c.sd=1)

<table>
<thead>
<tr>
<th>Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Frequency</strong></td>
<td></td>
</tr>
<tr>
<td>longlastyear</td>
<td>No. of Long termed investment deposit account over last year</td>
</tr>
<tr>
<td>notranshort</td>
<td>No. of transaction for short termed investment deposit account</td>
</tr>
<tr>
<td>notranshortlast</td>
<td>No. of transaction for short termed account over last year</td>
</tr>
<tr>
<td>notrantotal</td>
<td>LTD Total transaction</td>
</tr>
<tr>
<td>notrantotallast</td>
<td>Total transaction over last year</td>
</tr>
<tr>
<td><strong>Recency</strong></td>
<td></td>
</tr>
<tr>
<td>recencylastaccount</td>
<td>No. of days since last open account</td>
</tr>
<tr>
<td>recencylasttran</td>
<td>No. of days since last transaction</td>
</tr>
<tr>
<td>recencymaxtrandate</td>
<td>No. of days since maximum transaction</td>
</tr>
<tr>
<td><strong>Monetary</strong></td>
<td></td>
</tr>
<tr>
<td>totaldebitshort</td>
<td>LTD Total money that the customer draw from short termed account</td>
</tr>
<tr>
<td>totaldebitshortlast</td>
<td>Total money that the customer draw from short termed account-last year</td>
</tr>
<tr>
<td>totaltran</td>
<td>LTD total money that the customer put in all accounts</td>
</tr>
<tr>
<td>totaltranlast</td>
<td>total money that the customer put in all accounts over last year</td>
</tr>
<tr>
<td>totaltranlong</td>
<td>LTD total money that the customer put in Long termed account</td>
</tr>
<tr>
<td>totaltranlonglast</td>
<td>total money that the customer put in Long termed account-last year</td>
</tr>
<tr>
<td>totaltranshort</td>
<td>LTD total money that the customer put in short termed account</td>
</tr>
<tr>
<td>totaltranshortlast</td>
<td>Total money that the customer put in short termed account-last year</td>
</tr>
<tr>
<td>avgtranlong</td>
<td>Average LTD money that the customer put in long termed account</td>
</tr>
<tr>
<td>avgtranlonglast</td>
<td>Average money that the customer put in Long termed account-last year</td>
</tr>
<tr>
<td>avgtranshortlast</td>
<td>Average money that the customer put in short termed account-last</td>
</tr>
<tr>
<td><strong>Interaction</strong></td>
<td></td>
</tr>
<tr>
<td>inter55_27</td>
<td>recencylasttran * totalacceserv</td>
</tr>
<tr>
<td>inter14_4</td>
<td>recencylastaccount * long</td>
</tr>
<tr>
<td>inter17_14</td>
<td>longlastyear * recencylastaccount</td>
</tr>
<tr>
<td>inter28_14</td>
<td>totalacceservlastyear*recencylastaccount</td>
</tr>
<tr>
<td>inter55_6</td>
<td>recencylasttran * totalacc</td>
</tr>
<tr>
<td>inter87_55</td>
<td>recencymaxtrandate * recencylasttran</td>
</tr>
<tr>
<td>inter19_14</td>
<td>totalacclastyear * recencylastaccount</td>
</tr>
</tbody>
</table>

5.4 Split Data into Train and Test Set

After selection of best subset of features for modeling, the dataset contains 22427 customers was partitioned into training and test sets. It's important to create both a training data set, which is used to build the model, and a test or hold-back data set, which is used to test the model. A model should be tested against data that it has never seen before to ensure that there is no overfitting.
In this case we were trying to build a model that would predict a binary outcome (that is, the customer propensity to a particular product). Dataset for building this model contains 22427 customers with 20715 non-respondents and 1712 respondents. A 2/3 of customers (14952 records) were randomly assigned to the training set while the other 1/3 (7475 records) to the test set. Taking use of a training set SVM model was built, later the test set was used to evaluate the accuracy of model.

5.5 Balancing the Class Distribution

The dataset contains 22427 customers. The distribution of a positive class (respondents) was 1712 and the negative class was 20715 (non-respondents), so the response rate was 7%. Also when data was partitioned into training and test set, the response rate of training set was 7% with 1143 respondents and 13809 non-respondents, which means that the class distribution was moderately imbalanced. Various strategies have been proposed for class balancing (see section 3.3.3).

For balancing training set under-sampling the non-respondent class (majority class) was used for this study. Twice the numbers of respondents in training set, the non-respondents randomly were selected. After balancing the class distribution in training set (under-sample the non-respondents class) the number of non-respondents in training set were two times a number of respondents. So the response rate of training set was 33% with 1143 respondents and 2286 non-respondents, Table 5.7 shows the training set response rate before and after class balancing.

<table>
<thead>
<tr>
<th>Table 5.7: Class Balancing in Training Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Set No of Respondents</td>
</tr>
<tr>
<td>---------------------------------</td>
</tr>
<tr>
<td>Before Balancing</td>
</tr>
<tr>
<td>After Balancing (Under-Sampling)</td>
</tr>
</tbody>
</table>
5.6 Parameter Selection for Support Vector Machine

After completing the data preparation, feature construction and selecting the best subset of features for modeling the response, it is time to select the classification algorithm and then select the algorithm parameters. As explained above various classification methods (classifiers) have been used for response modeling such as statistical and machine learning methods. Neural networks, decision trees and support vector machines. In this study SVM was used as a classifier for classification. SVMs show distinct advantages such as better generalization, increased speed of learning; ability to find a global optimum and ability to deal with linearly non-separable data (see section 3.4.2).

For implementing SVM in R Language Program, svm function in e1071 package was used for classification task. As explained in section 4.9, SVM requires a certain amount of model selection (parameter search). The kernel parameter is one of the most important design choices for the SVM since it implicitly defines the structure of the high dimensional feature space where a maximal margin hyper-plane will be found. Thus the choice of the SVM kernel is crucial. As was recommended in the previous work Radial Basis Function (RBF) kernel was chosen for this study. There are two parameters while using RBF kernels: $C$ (cost = misclassification tolerance parameter) and $\gamma$ (gamma). In the R Language Program the default parameter of cost is 1 and default parameter of gamma is $1/\text{ncol (as.matrix(data))}$ which is 0.03571429 for this study. For gaining better SVM performance it is necessary to tune these parameters and chose the best ones.

As explained in chapter 4 (section 4.9) in order to fine tune these parameters and find the ones which generalize better, “grid-search” on $C$ and $\gamma$ using cross-validation (five fold cross-validation) was used. First, based on the default value of cost and gamma possible interval of $C$ and $\gamma$ with grid space were defined. Then, all grid points of $(C, \gamma)$ were tried to see which one gives the highest cross validation accuracy. Then, the best parameter was used to train the whole training set and generate the final model. Actually Parameter selection was performed using various values of $C$ and $\gamma$. 
Trying exponentially growing sequences of $C$ and $\gamma$ is a practical method to identify good parameters (for instance, $C = 10^{(-2:4)}$, $\gamma = 2^{(-8:0)}$). As explained, different combinations of cost and gamma using five fold cross-validations were examined. Pairs of $(C, \gamma)$ were tried and the one with the best cross-validation accuracy was picked. Among different combinations of cost and gamma were tried, combination of $C = 10^{(-2:4)} \{0.01, 0.1, 1, 10, 100, 1000, 10000\}$ and $\gamma = 2^{(-8:0)} \{0.00390625, 0.00781250, 0.01562500, 0.03125000, 0.06250000, 0.12500000, 0.25000000, 0.50000000, 1.00000000\}$ was the best one. The best $(C, \gamma)$ pair was $(10, 0.0078125)$ with the lowest error rate 0.2723793. As can be seen in Figure 5.2, the lowest error 0.27 belongs to the orange point in the whitest area which has a cost $= 10$ and gamma $= 0.0078125$. Thus these values for cost and gamma were used for modeling. Figure 5.2 visualizes the results of parameter tuning for SVM model.

Figure 5.2: Result of Parameter Tunning for SVM Model
After the best parameters of C and $\gamma$ (the best model) were chosen, SVM was trained with these parameters. Taking use of a training set, SVM model was built. Later the test set was used to evaluate the accuracy of model. The result of SVM classification (prediction) is shown in following section.

5.7 Prediction Result

After parameter tuning and selecting the best parameters for modeling, training set was used for building a model. Classification was performed using all the 26 features selected using random forests and the selected parameters. The RBF kernel was used for SVM model with parameter gamma ($\gamma$) set to 0.0078125, and the misclassification tolerance parameter C set to 10. These parameter settings were determined through a grid search approach over the combination of C and $\gamma$, \{0.01, 0.1, 1, 10, 100, 1000, 10000\} * \{0.00390625, 0.00781250, 0.01562500, 0.03125000, 0.06250000, 0.12500000, 0.25000000, 0.50000000, 1.00000000\}, using five fold cross-validation performance. The summary of SVM model is shown in Table 5.8

Table 5.8: SVM Model Summary

<table>
<thead>
<tr>
<th>Parameters:</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-Type: C-classification</td>
</tr>
<tr>
<td>SVM-Kernel: radial</td>
</tr>
<tr>
<td>cost: 10</td>
</tr>
<tr>
<td>gamma: 0.0078125</td>
</tr>
<tr>
<td>Number of Classes: 2</td>
</tr>
<tr>
<td>Levels: 0 1</td>
</tr>
</tbody>
</table>

After building a model on training set, to ensure that the model has not overfit the data, the test set was used to evaluate the model accuracy. The exciting model was executed to assess the model performance against the test data. Result returned when the model run on the test set and is shown by the confusion matrix. Basically the prediction result is shown by the confusion matrix. The confusion matrix contains information about actual and predicted classifications done by a classification system (model). Table 5.9, confusion matrix, shows the result returned when the model was
run on test set. The rows of the matrix are actual classes, and the columns are the predicted classes.

Table 5.9: Confusion Matrix of Model

<table>
<thead>
<tr>
<th></th>
<th>Predicted</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Class1 (Negative) Non-respondent (0)</td>
<td>Class2 (Positive) Respondent (1)</td>
</tr>
<tr>
<td>True Class1 (Negative) Non-respondent (0)</td>
<td>5727 (TN)</td>
<td>1179 (FP)</td>
</tr>
<tr>
<td>Class2 (Positive) Respondent (1)</td>
<td>245 (FN)</td>
<td>324 (TP)</td>
</tr>
<tr>
<td>Total</td>
<td><strong>5972</strong></td>
<td><strong>1503</strong></td>
</tr>
</tbody>
</table>

As can be seen in confusion matrix (Table 5.9), 5972 corresponds to the numbers of customers that the model classified as non-respondent and 1503 corresponds to the numbers of customers that the model classified as respondent. 5727 customers out of 6906 customers were predicted as non-respondents who were actually non-respondents (TN = True Negative), also 324 customers out of 569 customers were predicted as respondents who were actually respondents (TP = True Positive). 1179 out of 6906 customers were predicted as respondents while they were non-respondents (FP = False Positive). There were also 245 out of 569 customers predicted as non-respondents who were actually respondents (FN = False Negative).

The accuracy of a model and the other evaluation criteria are calculated by the confusion matrix. In this study the performance of SVM response model was measured by using Accuracy, True Positive Rate, True Negative Rate, Weighted Accuracy and Lift Chart (see section 3.5 for more information). Using confusion matrix (Table 5.9) these evaluation measurement were calculated also the lift/gain chart was drawn. Evaluation of the model and the analysis are described in more detail following.
5.8 Model Evaluation and Analysis

Once a data model has been created and tested, its performance is analyzed. In this study the performance of a model was measured by using Accuracy, True Positive Rate, True Negative Rate, Weighted Accuracy and Lift Chart.

The performance of a model do not evaluated on the training set, because it has had a hand in creating the model and so will overstate the model’s accuracy. The model’s accuracy always measure on a test set that is drawn from the same population as the training, but has not been used in any way to create the model (Berry and Linoff, 2004). Thus in this study as explained above the dataset were divided in to training and test set (2/3 for train set and 1/3 for test set). Train set was used for model building and test set for measure the model's accuracy. Result returned when the model run on the test set and is shown by the confusion matrix (Table 5.9). From confusion matrix the overall accuracy of the model was calculated. As can be seen in Table 5.10 the overall accuracy on the test set is 81%.

Table 5.10: SVM Model Performance on Test Set

<table>
<thead>
<tr>
<th>Predicted</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class1 (Negative)</td>
<td></td>
</tr>
<tr>
<td>Non-respondent (0)</td>
<td>5727</td>
</tr>
<tr>
<td>TNR=83%</td>
<td></td>
</tr>
<tr>
<td>Class2 (Positive)</td>
<td>1179</td>
</tr>
<tr>
<td>Respondent (1)</td>
<td></td>
</tr>
<tr>
<td>245</td>
<td>TPR=57%</td>
</tr>
<tr>
<td>324</td>
<td></td>
</tr>
<tr>
<td>81.75%</td>
<td></td>
</tr>
</tbody>
</table>

The accuracy (Acc) is widely used metric for measuring the performance of learning systems. But when the prior probabilities of the classes are very different, such metric might be misleading. A trivial classifier that predicts every case as the majority class can still achieve very high accuracy. In this study because the number of negative cases (non-respondents class) was much greater than the number of positive cases (respondent class) the overall classification accuracy was not an appropriate measure of performance. Thus, metrics such as True Negative Rate...
(TNR), True Positive Rate (TPR) and weighted accuracy were used to evaluate the performance of learning algorithms on imbalanced data. All the metrics are functions of the confusion matrix. Based on the confusion matrix (Table 5.9), the performance metrics were calculated as in Table 5.11.

<table>
<thead>
<tr>
<th>Evaluation Criteria</th>
<th>Formulation</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (Acc)</td>
<td>$\frac{TP + TN}{TP + FP + TN + FN}$</td>
<td>81%</td>
</tr>
<tr>
<td>Error Rate</td>
<td>$1 - Acc$</td>
<td>19%</td>
</tr>
<tr>
<td>True positive Rate (TPR)</td>
<td>$\frac{TP}{TP + FP}$</td>
<td>57%</td>
</tr>
<tr>
<td>(Acc+)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>True Negative Rate (TNR)</td>
<td>$\frac{TN}{TN + FP}$</td>
<td>83%</td>
</tr>
<tr>
<td>(Acc−)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weighted Accuracy $\lambda = 9$</td>
<td>$\frac{\lambda (TP) + TN}{\lambda (TP + FP) + (TN + FN)}$</td>
<td>72%</td>
</tr>
</tbody>
</table>

For any classifier, there is always a trade off between true positive rate and true negative rate. In the case of learning extremely imbalanced data, quite often the rare class is of great interest. In direct marketing application, it is desirable to have a classifier that gives high prediction accuracy over the minority class (True Positive Rate - Acc+), while maintaining reasonable accuracy for the majority class (True Negative Rate - Acc−). Actually accuracy for respondents' group is of greater importance to direct marketers, since their primary goal is to identify the respondents, not the non-respondents. Increasing the number of responders is the goal of response modeling. In such situations Weighted Accuracy is often used in such situations. For that reasons, in this study besides overall model accuracy True Positive Rate (Acc+), True Negative Rate (Acc−) and Weighted Accuracy with $\lambda = 9$ were calculated. Table 5.11 shows that the overall model accuracy was 81% with 57% true positive rate (Acc+) and 83% true negative rate (Acc-) also the weighted accuracy was 72%.

The data in the confusion matrix described above was plotted in a lift or gains chart to visually evaluate the results of the model performance and to compare the
performance of constructed response model within the results achieved by random chance. Lift/Gain chart is a graphical representation of the advantage of using a predictive response model to choose which customers to contact. A gain/lift chart shows the gain made by using a derived response model over random selection of targets. In this study lift chart was plotted in R Language Program. Prediction and Performance function in ROCR package were used for plotting lift chart.

The model computes each customer’s likelihood or propensity to respond to a particular offer of a product or a service. Lift chart was created by sorting all the prospects in descending order according to their likelihood of responding as predicted by the model and select instance subset starting from the one with the highest predicted probability. Figure 5.3 shows the lift/gain chart was created for the test set. The X axis shows the ratio (in percentage) of the size of mailed group of clients to the size of the total client base (TP+FP/TP+FP+TN+FN) and the Y axis shows the percentage of responders that is actually captured by the model, which is the true positive rate (TPR).

In Figure 5.3 the bottom line (straight line) is the random line and indicated that if the customers were randomly selected for marketing campaign. If no model were used, mailing to 20% of the population would reach 20% of the responders; mailing to 40% of the population would reach 40% of the responders. The other curve (curve above straight line) of chart shows what happens if the model is used to select customers for the campaign. If only top 20% of the customers are mailed the 57% of the respondents are approached as opposed to only 20% in a randomly targeted mailing of the same size. Actually by contacting only 20% of customers based on the predictive model Parsian bank will reach 3 times as many respondents as if Parsian bank use no model.
Figure 4.3: Lift Chart of the Model

According to following chart the model finds nearly 60% (57%) of the responders by mailing to only 20% of the population and finds 79% of the responders with a campaign directed to 40% of the customers.

Assume that Parsian bank strategy is just to contact the top 20% of the customers with the highest response score. By selecting top 20% customers nearly 60% of all respondents are found. It shows that by using this model Parsian bank can get three times as many responders for its marketing expenditure than it would have received by mailing to 20% of its one million prospects at random. So it can significantly reduce the overall marketing cost.

5.9 Features Importance

Response models are typically built from historical purchase data. For data mining purposes, the purchase history can often be translated into features based on measures of Recency, Frequency and Monetary values (Van den Poel, 2003). Cullinan (1977) is generally credited for identifying the three (RFM) variables most often used in database marketing modeling: recency, frequency and monetary value (Bauer, 1988). Since then, the literature has accumulated so many uses of these three
variables, that there is overwhelming evidence from existing studies that the RFM
tables are the most important set of predictors for response modeling. In this study
from raw data (collected information from Parsin bank) 85 features were constructed
also all the two way interaction between these variables were constructed. After
feature selection the best subset of feature which consists of 26 features was selected
for modeling. In this section we investigated that what feature (Recency, Frequency,
Monetary value or interaction) is the most important one for modeling the response
for Parsian bank. In the literature, we find some general statements such as ‘In
general the frequency variable is the most important of the set of three RFM
predictors’ (Nash, 1994), but due to the proprietary nature of most studies, no
detailed results are reported (Bass and Wind, 1995).

The importance of a variable or feature has to be evaluated given the presence
of other predictors in the model. Given the number of potentially important variables
(see Table 5.4) sequential procedure was followed. Constructed model was
investigated containing the following predictors: RF + interaction (total features -
monetary), RM + interaction (total features - frequency), FM + interaction (total
features - recency), RFM (total features - interaction). Figure 5.4 shows the result of
variable importance and reveals that frequency variables are more important than
other variables. The Y axis shows the features that the model was performed with
them and the X shows the true positive rate (TPR) of the model. In this study the
TPR was used as an evaluation metric for evaluating the goodness of features instead
of Accuracy (Acc), because in the direct marketing case the value of TPR is more
important than the value of Accuracy.

As can be seen in Figure 5.4, true positive rate of the model is 57% when it
was performed with all the 26 selected features (RFM and interaction). TPR is 48%
when the model was performed without Frequency features, it was performed with
RM and interaction features, TPR is 49% when the model was performed without
interaction features, TPR is 54% when the model was investigated without Monetary
features and also when it was performed without Recency variables the TPR is 56%.
The lowest TPR belongs to the condition that the model was investigated without
Frequency features. It shows that out of constructed features (RFM and interactions)
the frequency features are the most important one in modeling the response for
Parsian bank, followed by interaction features and monetary value. Finally the recency variables are the least important of the RFM variables and interactions. It means that, the larger the tendency of customers to engage frequently with bank seems to positively affect the probability of repeat purchase. It can be concluded multi-buyers (buyers who already purchased several times) and customers who have transaction with bank frequently are more likely to repurchase than buyers who only purchased once.

![Features Importance](image)

**Features**

*Figure 5.4: Features Importance*
Chapter 6

Conclusions and Further Research

6. Conclusions and Further Research

In this chapter conclusions are presented. Also the managerial implications and limitation of this research will be mentioned. At the end the further research on the area related to this study will be recommended.

6.1 Conclusion

In this thesis a response model for target selection in direct marketing with data mining techniques was constructed for Parsian bank. Parsian bank offers different products and services to its customers. This bank is faced with challenges of increasing competition, continually rising marketing costs, decreasing response rates
and also don't have a direct relationship with its customers. In order to combat these problems, Parsian bank wanted to select those customers who are most likely to be potential buyers of the new product or service and make a direct relationship with them. In simple word Parsian bank wants to select the “customers” that should be contacted in the next marketing campaigns. The purpose of this study was to predict whether an existing customer will purchase on the next marketing campaign or not, based on information provided by the purchase behavior variables. To reach this purpose, we have to develop a predictive response model with data mining techniques for Parsian bank to select the customers that should be targeted in order to obtain a percentage as high as possible of positive responses. By using this model, Parsian bank can obtain a percentage as large as possible of the targeted customers responds to the product offer and also minimize the marketing cost.

Response modeling is usually formulated as a binary classification problem. The customers are divided into two classes, respondents and non-respondents. Various classification methods (classifiers) have been used for response modeling such as statistical and machine learning methods. Neural networks, decision trees and support vector machines. Most recent is Support Vector Machine (SVM) that has been spotlighted in the machine learning community also offer advantages over multivariate classifiers. In this study support vector machine (SVM) was used as a classifier for modeling.

Response modeling procedure consists of different steps, such as: data collection, data preprocessing, feature construction, feature selection, class balancing, classification and model evaluation. Various data mining techniques and algorithm have been used for implementing each step. In this study the research process illustrated in Figure 4.4 was followed. This process was constructed based on previous methodology on response modeling. Due to nature of this study different steps (components) of modeling were collected from previous work and with some changes and amendments integrated them in to unique process. In order to select algorithm and technique for implementing each step of process, different studies related to each step were reviewed and evaluated. After reviewing various techniques and strategies for each step, the best and appropriate ones were selected. In order to implement this research process (building response model) an extensive
programming was required. Algorithms and techniques related to each step were programmed and run in R Language Program.

The first step of modeling was data collection. From Parsian Bank database, customers past purchase behavior data and campaigns data were obtained (Table 4.1). This allowed us, in close co-operation with domain experts and guided by the extensive literature to derive all the necessary purchase behavior variables (RFM variables) for a total sample size of 30,000 customers. Gathered data were in the ten text files. For data analysis and modeling purpose all these ten text files were combined and integrated into one table in SQLite database. As a form of preprocessing, records (customers) which had missing values and the data which were only an overhead and were not helpful to us in any way were deleted from table. After data cleaning dataset reduced to 22427 records.

In feature construction step, target variable and all the necessary purchase behavior variables (RFM) were constructed. From the raw data (collected from Parsian bank), for each customer, RFM variables were measured in the period between 1380 and 30/10/1383. The goal was to predict whether an existing customer will respond to an offer in the observation period time between 1/11/1384 and 1/5/1385 or not, based on the purchase history information, period between 1380 and 30/10/1384. All the customers were categorized as either a respondent (class 1) or as non-respondent (class 0) depending upon whether they made purchase during the period between 1/11/1384 and 1/5/1385 or not, respectively. In addition to construction individual RFM variables, all two way interactions between the 85 individual features were also constructed. Subsequent to feature construction, feature selection was performed. Feature selection was done in three steps using F-score and Random Forests. First twenty interaction features were selected using F-Score selection, which is the filter method, then selected interaction features were added to 85 individual features. In second step fifty important features were selected using F-score selection and finally in the third step the best subset of features were selected as input for modeling by using Random Forests, which is the wrapper method. The dataset was partitioned into training and test sets for performance evaluation. 2/3 of customers were randomly assigned to the training set while the other 1/3 to the test set. The data was highly unbalanced. The response rate was 7%. Thus for balancing
training set under-sampling the none-respondent class (majority class) was used. Also, the data was normalized and scaled before presenting it to the SVM to ease mathematical calculations as well as reduce the effect of larger attributes.

SVM was used for response modeling. SVM required a certain amount of model selection. The RBF kernel was used with parameter gamma ($\gamma$) set to 0.0078125, and the misclassification tolerance parameter C set to (10). These parameter settings were determined through a grid search over the different combination of C and $\gamma$ using five fold cross-validation performances. Five-fold cross validation was conducted on the training sets for model and the best parameter set was selected which resulted in the best model selection criterion. After selecting the best parameters for SVM, classification was performed with twenty six features selected from Random Forests. Training set was used to build the SVM model and then the test set was used to evaluate the accuracy of model. All the SVM analysis were done using e1071 package in R Language Program.

Performance of the SVM model was measured using Accuracy, True Positive Rate, True Negative Rate, Weighted Accuracy and Lift/Gain chart. The overall model accuracy on test set is 81% with 57 % true positives rate and 83% true negatives rate. Response model compute each customer's likelihood or propensity to respond to a particular offer of a product or a service. Lift chart was created by sorting all the prospects in descending order according to their likelihood of responding as predicted by the model. The lift chart shows that, for example, if only top 20% of the customers are mailed the 57% of the respondents are approached as opposed to only 20% in a randomly targeted mailing of the same size. Actually by contacting only 20% of customers based on the predictive model Parsian bank will reach 3 times as many respondents as if they use no model. In conclusion predictive response model help Parsian bank to identify a subset of customers who are more likely to respond than others and establish a direct relationship with them. By using this model companies not only can significantly reduce the overall marketing cost but also can prevent to irritate the customers and improve customer relationship management.
6.2 Research Limitations

This study, like all others, is not without its limitations. There were some limitations in this research work:

- Lack of good databases and data warehouses to store customers' information in Iranian companies. Actually one of the important limitations of this research was data collection. As you know for data mining purpose secondary data is used (customer data stored in company's databases). Finding a good database which store needed information for building a direct marketing response model was very hard. Also convincing managers to give us the customers' information was another problem. Actually doing data mining in Iran is very hard because there is no database management in the companies.

- For building a response model RFM variables and demographic customer information are generally used. Constructed response model included RFM variables and little demographic customer information. Only the customers’ age was incorporated. Since the Parsian bank that provided the data does not collect this type of information when customers register, no customer gender (female/male), average income and place of live were at our disposal for the predictive model.

- Lack of powerful and appropriate data mining software and tools. The only free and available software that we can use for implementing each step of modeling was R Language Program. In order to implement each step in R different packages were downloaded and installed. Also the algorithm related to each step should be programmed and coded; actually for building this model the extensive programming was required. Learning this language, finding an appropriate package and programming was very hard and time consuming.

- Target variable is usually build based on the customer's historical response data to similar direct marketing campaigns. One of the major limitation of this work was that Parsian bank and many companies in Iran use mass marketing as their strategy for offering a new product or service to customers and they did not have direct marketing campaign and customer's historical response data to direct marketing campaign. So in this thesis for building the target variable we had to use customer's historical response data to mass marketing campaigns.
6.3 Managerial Implications

The findings of this study have important implications for growing number of companies, especially financial services, banks and insurance companies.

Great deals of customer information are available to many companies and organizations. This data can be a rich source of knowledge, if only properly used. This can be very beneficial for the companies using data mining to extract knowledge and useful information from this data. Banks and insurance companies by mining this data can customize their product according to client needs and to better target their various marketing campaigns in order to reduce the cost. Actually data mining plays an important role in marketing by allowing the marketer to harness data about customers and prospects to manage relationships between customers and increase marketing efficiency. So, all the companies and organizations should realize the importance of data mining in their strategic planning and successful application of data mining techniques can be an enormous payoff for the organizations.

The main objective of this study was to build a response model with data mining techniques (by analyzing customers' data) for Parsian bank to select the customers that should be targeted in the next marketing. The result shows that by using this model, Parsian bank can obtain a percentage as large as possible of the targeted customers responds to the product offer and also minimize the marketing cost. Predictive response model helps companies (marketers) to identify a subset of customers who are more likely to respond than others and establish a direct relationship with them. By using this model companies not only can significantly reduce the overall marketing cost but also can prevent to irritate the customers. Since the product may only be interesting to a subset of the total customer base, sending many uninteresting product offers to customers and undesirable mail or phone calls leads to irritation, also sending product offer to all the customers the cost of such a full-scale campaign soon become too large, and rise above the expected returns.

Given a tendency of rising mailing costs and increasing competition, the importance of response modeling for companies increased. Companies and organization by improving the targeting of the offers (using response model) may indeed counter the challenges of increasing competition and high marketing cost by
lowering non response. Moreover, from the perspective of the recipient of the product offer (customers) companies do not want to overload customers by offering. Response model is a profitable tool in fine-tuning direct marketing strategies since even small improvements attributed to modeling can create great financial gains. The main purpose of response modeling is to improve future campaign return on investment and our study shows that an increase in response of only one percentage point can result in substantial profit increases.

In summary companies and organizations should realize the advantages and importance of the data mining in their strategic planning and use the customer data, which is stored in databases, to extract knowledge and useful information. Also it can be very beneficial for them to build the direct marketing response model with data mining techniques to identify a subset of customers who are more likely to respond than others and establish a direct relationship with them, so that the response to the campaign is maximized, the cost of marketing is minimized and the customer relationship is increased.

6.4 Suggestions for Further Research

With the development of data mining techniques and databases, some areas which are not covered in this study are interesting and need to be explored. In addition, the limitations and shortcoming of this study also provide suggestion for future research. Future research could add extension to this study.

Following future work can be done for this research:

• This study was mainly focus on building a predictive response model with Support Vector Machine (SVM) classifier and deal with all the practical difficulties when applying SVM to response model in direct marketing, such as feature selection and class balancing. It was not our ambition to compare the performances of different classifiers (classification algorithms) when applying to response model. Other classification algorithms can be used for response modeling. Further research is suggested that to compare the performances of different classification algorithms when apply to response model.

• Response modeling procedure consists of various steps. Different data mining techniques can be applied for implementing each step of modeling. Considering
available tools, time and literature we try to select the best and the most recent techniques and algorithms for each step. For example for feature selection step, F-score and Random Forests were used also under-sample the non-respondents class for class balancing was used for modeling and so on. It is suggested that for future research, apply other data mining techniques for feature selection, class balancing and sampling and etc. and it would be interesting to compare the performance of the different techniques. Therefore, the predictive accuracy of model might even increase when other techniques apply for modeling.

- Constructed response model included RFM variables and little demographic customer information to predict response. Only the customers' age was incorporated. Since the Parsian bank does not collect this type of information when customers register, such as gender (female/male), place of live and average income, so demographics were not use in this study. Therefore, the further research is needed to use demographics information for building a model. Predictive ability of model might even increase when demographics are available from the company’s internal data files.

- In feature construction step, because of lack of appropriate method and tool for searching two-way interactions between individual features, all the two-way interactions were constructed and then by the means of F-score selection the most appropriate ones were selected. It is suggested that for future work instead of constructing all the interactions, first interactions between the predictor variables identified by means of appropriate models and tools, which searches for two-way interactions between candidate predictors, and then after identifying which variables interacted, add products between interacting variables.

- In data cleaning step, records which had missing values and the data which were only an over head and were not helpful to us in any way were deleted. It is important to deal with noisy data too. So further research is needed to remove noisy data with clustering techniques.

- Since the Parsian bank use mass marketing as it strategy for offering a new product or service to its customers, they did not have customer's historical response data to direct marketing campaign. So in this thesis for building the target variable customer's historical response data to mass marketing campaign was used. Direct marketing campaign information can be used for predicting response.
References


## Appendix 1

### Table of Literature Review on RFM variables were used in Response Modeling

<table>
<thead>
<tr>
<th>References</th>
<th>Feature Used</th>
</tr>
</thead>
</table>
Number of days since the last purchase within a specific  
**Frequency**  
The number of purchases made in a certain time period (Hist versus Year).  
**Monetary**  
Total accumulated monetary amount of spending by a customer during a certain time period (Hist versus Year). |
| **Malthous (2002)** | **Recency**  
(Recency): days since most recent purchase  
(Dayfp): days since first purchase  
(Purseas): number of seasons with a purchase  
**Frequency**  
(Falord): LTD fall orders  
(Freq): LTD orders  
(Ordlyr): number of orders last year  
(Ordtyr): number of orders this year  
**Monetary**  
(Mon): LTD dollars  
(Slslyr): dollar demand from last year  
(Slstyr): dollar demand from this year  
**Interactions**  
int1: an indicator variable taking the value 1 if recency is less than 1 year and dayfp is greater than 1 year, and 0 otherwise (indicates “old” and recent customers)  
int2: ordtyr* ordlyr  
int3: dayfp/ recency  
int4: ordtyr* purseas  
int5: ordtyr* falord  
tran1: $\sqrt{\text{int3}}$  
tran2: 1/(recency +1)  
tran3: $\sqrt{\text{int2}}$  
tran4: log(slstyr +1)  
tran5: log(slslyr +1),  
tran6: dayfp2,  
tran7: log(recency) |
<table>
<thead>
<tr>
<th>References</th>
<th>Feature Used</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Malhous (2002)</strong></td>
<td>Recency&lt;br&gt;(recency) months since latest contribution.&lt;br&gt;(cnmonf) months since first contribution.&lt;br&gt;(cnmonl) months since largest contribution.</td>
</tr>
<tr>
<td></td>
<td><strong>Frequency</strong>&lt;br&gt;(Cntmlif) number of contributions life-to-date.&lt;br&gt;(sltmlif) number of solicitations life-to-date.</td>
</tr>
<tr>
<td></td>
<td><strong>Monetary</strong>&lt;br&gt;(Cntrlif) life-to-date dollars contributed.</td>
</tr>
<tr>
<td></td>
<td><strong>Interaction</strong>&lt;br&gt;tran1: dollars per contribution (cntrlif/ cntmlif)&lt;br&gt;tran2: dollars per solicitation (cntrlif/ sltmlif).&lt;br&gt;tran3: contributions per solicitation (cntmlif/ sltmlif).&lt;br&gt;tran4: log (recency +1).&lt;br&gt;tran5: square root of monetary value.&lt;br&gt;tran6: inverse of monetary value.&lt;br&gt;tran7: inverse of tran3, solicitations per contribution.&lt;br&gt;tran8: square root of dollars per solicitation.&lt;br&gt;tran9: square of solicitations.&lt;br&gt;tran10: log of dollars per contribution.&lt;br&gt;tran11: monetary value/ (recency + 1).&lt;br&gt;tran12: frequency/ (recency + 1).&lt;br&gt;tran13: log (monetary value/ (recency+1)).</td>
</tr>
<tr>
<td><strong>Malhous (2001)</strong></td>
<td>Recency&lt;br&gt;(purseas): Number of seasons with a purchase</td>
</tr>
<tr>
<td></td>
<td><strong>Frequency</strong>&lt;br&gt;(ordtyr): number of orders this year&lt;br&gt;(puryear): number of years with a purchase&lt;br&gt;(sprord): LTD spring orders</td>
</tr>
<tr>
<td></td>
<td><strong>Interaction</strong>&lt;br&gt;tran51: recency &lt;90&lt;br&gt;tran52: 90 &lt;recency &lt;180&lt;br&gt;tran53: 180 &lt;recency &lt; 270&lt;br&gt;tran54: 270 &lt;recency &lt;366&lt;br&gt;tran55: 366&lt; recency &lt; 730&lt;br&gt;comb2&lt;br&gt;tran46 = √comb2: number of product groups purchased from this year&lt;br&gt;tran42 =log(1 + ordtyr * falord): interaction between the number of orders this year and the number of fall orders&lt;br&gt;tran44 =√ordhist * sprord: interaction between LTD orders and LTD spring orders&lt;br&gt;tran25 = 1/(1 + lorditm): inverse of latest-season items</td>
</tr>
<tr>
<td>References</td>
<td>Feature Used</td>
</tr>
<tr>
<td>------------</td>
<td>--------------</td>
</tr>
</tbody>
</table>
(Recency) Order days since 10/1992  
(Purseas) Number of seasons with a purchase  
(Purveyear) Number of years with a purchase |
|  | **Frequency**  
(Falord) LTD fall orders  
(Ordtyr) Number of orders this year  
(Sprord) LTD spring orders |
|  | **Interaction**  
Tran51: I (0<recency<90)  
Tran52: I(90<recency<180)  
Tran53: I(180 <recency < 270)  
Tran54: I(270 < recency < 366)  
Tran55: I(366 <recency <730)  
Tran38: 1/recency  
Comb2: Number of product groups purchased from this year  
Tran46: √comb2  
Tran42: Interaction between the number of orders:  
\( \log(1 + \text{ordtyr} \cdot \text{falord}) \)  
Tran44: Interaction between LTD orders and LTD spring orders  
\( \text{ordhist} * \text{sprord} \)  
Tran25: Inverse of latest-season items 1/(1 + lorditm) |
### Appendix 2

### Variables in Random Forest

<table>
<thead>
<tr>
<th>Iteration</th>
<th>OOB error</th>
<th>No. of selected Variables</th>
<th>Selected Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>mean = 0.0777</td>
<td>40</td>
<td>totaltranlast</td>
</tr>
<tr>
<td></td>
<td>sd = 0.0018</td>
<td></td>
<td>totaltran</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totaltranshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totaltranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>avgtranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totaldepositshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totaldepositshortshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter19_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencylastaccount</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter28_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notrantotallast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notrantotal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_27</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter17_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notranshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencylasttrtran</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter87_55</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>longlastyear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencymaxtrandate</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter14_4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totaltranlonglast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totaltranlong</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>avgtranlonglast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>avgtranlong</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter58_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter74_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_28</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>mintranlonglast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>mintranlong</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>maxtranlong</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>maxtranlonglast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_19</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>long</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter16_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totalacclastyear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>short</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totalaccservlastyear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>totalacc</td>
</tr>
<tr>
<td>Iteration</td>
<td>OOB error</td>
<td>No. of selected Variables</td>
<td>Selected Variables</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>---------------------------</td>
<td>--------------------</td>
</tr>
</tbody>
</table>
| 2         | mean = 0.0772  
            sd = 0.0018 | 32 | Totaltranlast  
            Totaltran  
            Totaltranshort  
            Totaltranshortlast  
            avgtranshortlast  
            totaldepositshortlast  
            totaldepositshort  
            inter19_14  
            recencylastaccount  
            inter28_14  
            notrantotallast  
            notrantotal  
            inter55_27  
            inter17_14  
            notranshort  
            recencylasttran  
            notranshortlast  
            inter87_55  
            inter55_6  
            longlastyear  
            recencymaxtrandate  
            inter14_4  
            totaltranlonglast  
            totaltranlong  
            avgtranlonglast  
            avgtranlong  
            inter58_14  
            inter74_14  
            inter55_28  
            mintranlonglast  
            mintranlong  
            maxtranlong |
| 3         | mean = 0.0781  
            sd = 0.0018 | 26 | inter17_14  
            notranshort  
            recencylasttran  
            notranshortlast  
            inter87_55  
            inter55_6  
            longlastyear  
            recencymaxtrandate  
            inter14_4  
            totaltranlonglast  
            totaltranlong  
            avgtranlonglast  
            avgtranlong |
<table>
<thead>
<tr>
<th>Iteration</th>
<th>OOB error</th>
<th>No. of selected Variables</th>
<th>Selected Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaltranlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaltran</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaltranshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaltranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Avgtranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaldepositshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaldepositshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter19_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencylastaccount</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter28_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notrantotallast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notrantotal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_27</td>
</tr>
<tr>
<td>4</td>
<td>mean = 0.0788</td>
<td>21</td>
<td>Totaltranlast</td>
</tr>
<tr>
<td></td>
<td>sd = 0.0018</td>
<td></td>
<td>Totaltran</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaltranshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaltranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Avgtranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaldepositshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Totaldepositshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter19_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencylastaccount</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter28_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notrantotallast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notrantotal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_27</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter17_14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notranshort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencylasttran</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>notranshortlast</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter87_55</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>inter55_6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>longlastyear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>recencymaxtrandate</td>
</tr>
</tbody>
</table>