Chapter 9
Advances in Business Analytics at HP Laboratories

Business Optimization Lab, HP Labs, Hewlett-Packard

Abstract HP Labs' Business Optimization Lab is a group of researchers focused on developing innovations in business analytics that deliver value to HP. This chapter describes several activities of the Business Optimization Lab, including work in product portfolio management, prediction markets, modeling of rare events in marketing, and supply chain network design.

9.1 Introduction

Hewlett-Packard is a technology company that operates in more than 170 countries around the world. HP explores how technology and services can help people and companies address their problems and challenges and realize their possibilities, aspirations, and dreams.

HP provides infrastructure and business offerings ranging from handheld devices to some of the world's most powerful supercomputer installations. HP offers consumers a wide range of products and services from digital photography to digital entertainment and from computing to home printing. HP was founded in 1939. Its corporate headquarters are in Palo Alto, CA. HP is among the world's largest IT companies, with revenue totaling $118.36 billion for the fiscal year that ended Oct 31, 2008.

HP's three business groups drive industry leadership in core technology areas:

- Personal Systems Group: business and consumer PCs, mobile computing devices and workstations.

Dirk Beyer, M-Factor, Inc. • Scott Clearwater • Kay-Yut Chen, HP Labs • Qi Feng, McCombs School of Business, University of Texas at Austin • Bernardo A. Huberman, HP Labs • Shailendra Jain, HP Labs • Zainab Jamal, HP Labs • Alper Sen, Department of Industrial Engineering, Bilkent University • Hsiu-Khuern Tang, Intuit • Bob Tarjan, HP Labs • Krishna Venkatraman, Intuit • Julie Ward, HP Labs • Alex Zhang, HP Labs • Bin Zhang, HP Labs

M.S. Sodhi, C.S. Tang (eds.), *A Long View of Research and Practice in Operations Research and Management Science*, International Series in Operations Research & Management Science 148,
DOI 10.1007/978-1-4419-6810-4_9,

- Imaging and Printing Group: Inkjet, LaserJet and commercial printing, printing supplies, digital photography and entertainment.
- Enterprise Business Group: enterprise services, business products including storage and servers, software and technology services for customer support.

At its heart, HP is a technology company, fueled by progress and innovation. The majority of HP's research is conducted in our business groups, which develop the products and services we offer to customers. As Hewlett-Packard's central research organization, HP Labs' role is to invent for the company's future.

HP Labs' function is to deliver breakthrough technologies and technology advancements that provide a competitive advantage for HP and to create business opportunities that go beyond HP's current strategies. The lab also helps shape HP strategy, and it invests in fundamental science and technology in areas of interest to HP.

For more than 40 years, HP Labs has been advancing technology and improving the way our customers live and work. From the invention of the desktop scientific calculator and the HP LaserJet printer to blade technology innovations and power-efficiency improvements for data centers, HP Labs is continuously pushing the boundaries of research to deliver more valuable technology experiences.

With 600 researchers across 23 labs in seven worldwide locations, HP Labs brings together some of the most distinguished researchers across a diverse set of scientific and technical disciplines—including experts in economics, science, physics, computer science, sociology, psychology, mathematics, and engineering.

These dedicated researchers are tackling some of the most important challenges of the next decade through a focus on high-impact research, a commitment to open innovation, and a drive to transfer technology to the marketplace. HP Labs' goal is to create breakthrough technology experiences for individuals and businesses around the world.

HP's deep roots in technologies and very competitive business environment provide a very rich set of opportunities for applied research in advanced analytics. Some of this applied research thrust in analytics is directed toward new product or service creations, though the major share of activities is geared toward operational processes innovation. This chapter describes selected activities of HP Labs' Business Optimization Lab, a group focused on advancing technologies and building high-impact innovative applications for operations and personalization, both driven by advanced analytics.

The researchers in the Business Optimization Lab exploit opportunities to build upon existing methodologies and create advanced analytics models and solutions for a comprehensive array of business contexts. The applications of this work span a wide range of areas including marketing, supply chain management, enterprise-wide risk management, service operations, and new service creation. Methodologies driving this applied research at HP Labs include operations research, industrial engineering, economics, statistics, marketing science, and computer science. For a summary of these activities see Jain [15].

9.1.1 Diverse Applied Research Areas with High Business Impact

This chapter presents four applied research projects conducted in the Business Optimization Lab that address HP's business needs in diverse areas.

The first study describes HP Labs' work in product variety management, which is at the interface of marketing and supply chain management decisions. Conventional wisdom suggests that a manufacturer should offer a broad variety of products in order to meet the needs of a diverse set of customers. While this is true to an extent, product variety comes with significant operational costs, which in excess may be counter-productive to profitability. Since the 1990s HP has faced many of these challenges due to its vast product portfolio. Business units sought methods to understand the costs of complexity and to identify which products were truly important to their business, so that they could refine their product offering without compromising revenue. To address these challenges, HP Labs introduced a new metric, coverage, for evaluating product portfolios in configurable product businesses. Coverage looks beyond the individual performance of products and considers their interdependence through orders. This metric, and HP Labs' accompanying Revenue Coverage Optimization tool (RCO), enables HP to identify products most critical to its offering, as well as candidates for discontinuance. As a result, HP has improved its operational focus on key products while also reducing the complexity of its product offering, leading to significant business benefits.

The second section describes the methodology and application of prediction market for forecasting business events, when markets are not efficient. Forecasting has been important since the dawn of business. There are two approaches in the context of using information for forecasting. The popular approach, backed up by decades of development of computing technologies, is the use of statistical analysis on historical data. This approach can be very successful when the relevant information is captured in historical data. In many situations, however, there is either no historical data or the data contain no patterns useful for forecasting. A good example is forecasting the demand of a new product. Thus, a second approach is to tap into tacit and subjective information in the minds of individuals. This so-called wisdom of crowds phenomenon has been documented over the centuries. The prediction markets, where people are allowed to interact in organized markets governed by well-defined interaction rules, have been shown to be an effective way to tap into the collective intelligence of crowds. If these markets are large enough and properly designed, they can be more accurate than other techniques for extracting diffuse information, such as surveys and opinions polls. Forecasting business events, on the other hand, may involve only a handful of busy experts, and they do not constitute an efficient market. We describe an alternate method of harnessing the distributed knowledge of a small group of individuals by using a two-stage mechanism. This mechanism is designed to work on small groups, or even an individual. This technique has been applied to several real-world demand forecasting problems. We will present a case study of its use in demand forecasting a technology hardware product and also discuss issues about real-world implementation.

In the third area, we describe modeling of rare events in marketing. A rare event is an event with a very small probability of occurrence. Typical examples of such events from social sciences that readily come to mind are wars, outbreak of infections, and breakdown of a city's transport system or levies. Examples of such events from marketing are in the area of database marketing (e.g., catalogs, newspaper inserts, direct mailers sent to a large population of prospective customers) where only a small fraction (less than 1%) responded resulting in a very small probability of a response (event). More recent examples of rare events have emerged in marketing with the advent of the Internet and digital age and the use of new types of marketing instruments. A firm can reach a large population of potential customers through its web site, display ads, e-mails, and search marketing. But only a very small proportion of those exposed to these instruments respond. To make business and policy planning more effective it is important to be able to analyze and predict these events accurately. Rare event variables have been shown to be difficult to predict and analyze. There are two sources of the problem. The first source is that standard statistical procedures, such as logistic regression, can sharply underestimate the probability of rare events. The second source of the problem is that commonly used data collection strategies are grossly inefficient for rare events data. In this study we share a choice-based sampling approach to discrete-choice models and decision-tree algorithms to estimate the response probabilities at the customer level to a direct mail campaign when the campaign sizes are very large (in millions) and the response rates are extremely low. We use the predicted response probabilities to rank the customers which will allow the business to run targeted campaigns.

In our fourth and last study, we describe a mathematical programming model that constitutes the core of a number of analytical decision support applications for decision problems ranging from design of manufacturing and distribution networks to evaluation of complex supplier offers in logistics procurement processes. We provide some details on two applications of the model to evaluate various distribution strategy alternatives. In these applications, the model helps answer questions such as whether it is efficient to add more distribution centers to the existing network and which distribution centers and transport modes are to be used to supply each customer location and segment, by quantifying the trade-off between the supply chain costs and order cycle times.

9.2 Revenue Coverage Optimization: A New Approach for Product Variety Management

HP's Personal Systems Group (PSG) is a $40B business that sells workstations, desktops, notebooks, and handheld devices to consumers and businesses. In October 2004, PSG offered tens of thousands of distinct products in its product lines. PSG's Global Business Unit Team knew their large and complex product offering led to confusion among sales people and customers, high administrative costs for forecasting and managing inventory of each product, and, most seriously,

poor order cycle time (OCT). A typical PSG order consists of many products, and an order does not ship until each of its products is available, so a stock-out of a single product delays the entire order. Because PSG's product line was so large, it was difficult and costly to maintain adequate availability for all products. Consequently, PSG's average OCT ranged from 11 to 14 days in North America (depending on the product line) compared to 5–7 days for the leading competitor. This difference adversely affected HP's customer satisfaction and market share.

The PSG team sought to identify a "Core Portfolio" of products that were most important to achieve their business goals. Once these Core products were identified, PSG could reduce the wait time for these products by renegotiating supply contracts and increasing inventory as needed. PSG also hoped to identify lower-priority products and either eliminate them from the product offering or offer them with longer lead times than Core Portfolio products. Prior to 2004, PSG used revenue thresholds as the measure for product importance. However, revenue is an insufficient criterion because it fails to recognize that some low-revenue products, such as power supplies, are critical to fulfilling many orders. PSG needed a more effective way to measure each product's importance.

Similar product proliferation issues affected other parts of HP, including Business Critical Systems (BCS). Business leaders sought the help of OR researchers and practitioners in the company to manage HP's product portfolio in a disciplined manner. As a result, HP created two powerful OR-based solutions for managing product variety (see Ward et al. [29].) The first solution, developed by HP's Strategic Planning and Modeling (SPaM) group, is a framework for screening new products *prior to introduction*. It uses custom-built return-on-investment (ROI) calculators to evaluate each proposed new product; those that do not meet a threshold ROI level are targeted for exclusion from the proposed lineup. The second, HP Labs' Revenue Coverage Optimization (RCO) tool, is used to manage product variety *after introduction*. RCO enables HP businesses to increase operational focus on their most critical products. Together, these tools have enabled HP to streamline its product offerings, improve execution, achieve faster delivery, lower overhead, and increase customer satisfaction and market share.

This chapter focuses on the second solution. It describes the RCO technology for managing product variety after it has been introduced into the portfolio and its implementation in HP. The next section introduces the metric of coverage for evaluating a product portfolio and describes the evolution of approaches that led to a fast new maximum flow algorithm for revenue coverage optimization. The subsequent sections present the results achieved through the use of RCO in HP, followed by concluding remarks.

9.2.1 Solution

9.2.1.1 Coverage: A New Metric for Product Portfolios

The joint business unit and HP Labs team knew that when determining the importance of products in an existing product portfolio, it would not suffice to examine

each product in isolation in order history, particularly in a business where orders consist of many interdependent items. As mentioned previously, a product that generated relatively little revenue of its own could, in fact, be a critical component to some large-revenue orders, and therefore be essential to order fulfillment. To address this, HP Labs developed a new metric of a product portfolio that captures the interrelationship among products and orders. This metric, called *order coverage*, represents the percentage of past orders that could be completely fulfilled from the portfolio. Similarly, *revenue coverage* of a portfolio is the revenue of its covered orders as a percentage of the total revenue of orders in the data set. The concept of coverage provides a meaningful way of measuring the overall impact of each product on a business. The tool we developed, called the Revenue Coverage Optimization (RCO) Tool, finds the smallest portfolio of products that covers any given percentage of historical order revenue.[1] More generally, given a set of historical orders, RCO computes a nested series of product portfolios along the efficient frontier of order revenue coverage and portfolio size.

The black curve in Figure 9.1 illustrates this efficient frontier. In this example, 80% of order revenue can be covered with less than 27% of the total product portfolio, if those products are selected according to RCO's recommendations. One can use this tool to select the portfolio along the efficient frontier that offers the best trade-off—relative to their business objectives—between revenue coverage and portfolio size. The strong Pareto effect in the RCO curve presents an important

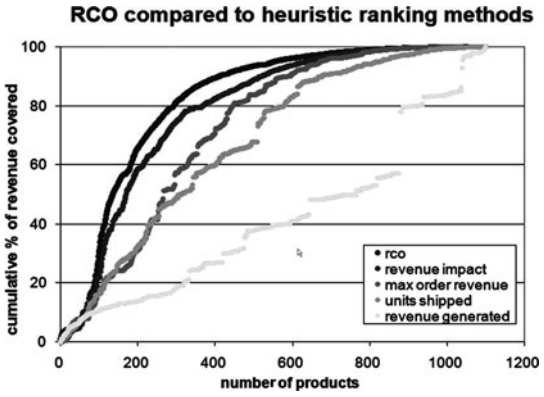

Fig. 9.1 This chart shows revenue coverage vs. portfolio size achieved by RCO (*black*) and four other product ranking methods, applied to the same historical data. The four other curves, in decreasingly saturated grays, are based on ranking by the following product metrics: revenue impact (the total revenue of orders containing the product); maximum revenue of orders containing the product; number of units shipped; and finally, individual product revenue

[1] In a nutshell, the RCO tool answers questions like "If I can pick only 100 products, which ones should I choose so I can maximize the revenue from orders that *only* have these products in it?" We argue, this is a better question to ask than "Which 100 products sold the most units?" or "Which 100 products show the highest line-item revenue?"

opportunity to improve on-time delivery performance. A small investment in improved availability of the top few products will significantly reduce average OCT.

In the remainder of this section, we describe the evolution of the RCO tool.

9.2.1.2 Math Programming Approaches to Optimize Coverage

The HP Labs team started by formulating the problem of finding the portfolio of size at most n that maximizes the revenue of covered orders as an integer program, IP(n):

$$\begin{aligned}
&\text{IP}(n)\text{: Maximize } \textstyle\sum_o r_o y_o \text{ subject to:}\\
&\quad (1)\ y_o \le x_p \text{ for each product-order combination } (o,p)\\
&\quad (2)\ \textstyle\sum_p x_p \le n\\
&\quad (3)\ x_p \in \{0,1\}, \quad y_o \in \{0,1\},
\end{aligned}$$

where r_o is the revenue of order o, and binary decision variables x_p and y_o represent whether product p is included in the portfolio and whether order o is covered by the portfolio, respectively.

Solving this integer program can be difficult in practice. Typical data sets have hundreds of thousands of product–order combinations, leading to hundreds of thousands of constraints of type (1). The integer program can take many hours to solve, and in some very large cases cannot be solved at all due to computer memory limitations.

However, it does have the nice property that constraints (1) are totally unimodular. This observation led to the following Lagrangian relaxation, denoted by LR(λ), in which we replace constraint (2) with a term in the objective penalizing the number of products used in the solution by a nonnegative scalar λ:

$$\begin{aligned}
&\text{LR}(\lambda)\text{: Maximize } \textstyle\sum_o r_o y_o - \lambda \sum_p x_p \text{ subject to:}\\
&\quad y_o \le x_p \text{ for each product–order combination } (o,p)\\
&\quad x_p \in [0,1],\ y_o \in [0,1].
\end{aligned}$$

The Lagrangian relaxation offers several advantages over the integer program. As mentioned previously, the remaining constraints are totally unimodular and so its optimal solution to a linear program is integer. Moreover, if a set of orders and products (O, P) is the optimal solution to LR(λ), then it will be an optimal solution to the original integer program IP($|P|$).

One very nice property of the series of solutions generated by this method is that they are nested, as is shown in the proof of the following theorem. This nested property is essential to application of the approach in business decisions, where a range of alternative portfolio choices are desired. Let $O(\lambda)$ denote the set of orders covered in the optimal solution to LR(λ), and let $P(O)$ denote the set of all products appearing in at least one order in O.

Theorem 1 *If* $\lambda_1 < \lambda 2$, *then* $O(\lambda_2) \subseteq O(\lambda_1)$.

Proof Suppose $O\lambda_2 \not\subseteq O(\lambda_1)$. Then let $O' = O(\lambda_2) \backslash O(\lambda_1) \neq \emptyset$. Then

$$\begin{aligned} 0 \geq & |O'| - \lambda_1 |P(O') \backslash P(O(\lambda_1))| \\ > & |O'| - \lambda_2 |P(O') \backslash P(O(\lambda_1))| \\ \geq & |O'| - \lambda_2 |P(O') \backslash P(O(\lambda_1))|. \end{aligned}$$

The first inequality holds by the optimality of $O(\lambda_1)$ for λ_1; if this inequality were not true, then one could increase the objective function of $\mathrm{LR}(\lambda_1)$ by adding the orders in O' to $O(\lambda_1)$. The second inequality follows from the fact that $\lambda_1 < \lambda_2$. The third inequality is true because, by the definition of O', the set of orders $O(\lambda_2) \backslash O'$ is contained in $O(\lambda_1)$ and so $P(O(\lambda_2) \backslash O') \subseteq P(O(\lambda_1))$. However, if $|O'| - \lambda_2 |P(O') \backslash P(O(\lambda_2) \backslash O')| \leq 0$, then one could improve the objective of $\mathrm{LR}(\lambda_2)$ by removing O' from $O(\lambda_2)$, which contradicts the optimality of $O(\lambda_2)$ for $\mathrm{LR}(\lambda_2)$. Thus $O(\lambda_2) \subseteq O(\lambda_1)$. □

Solving $\mathrm{LR}(\lambda)$ for a series of values of λ generates a series of solutions to $\mathrm{IP}(n)$ for several values of n. These solutions lie along the efficient frontier of revenue coverage vs. portfolio size. This series does not provide an integer solution for every possible value of n; solutions below the concave envelope of the efficient frontier are skipped. However, a wise selection of values of λ produces quite a dense curve of solutions for typical HP data sets; the number of distinct solutions is typically at least 85% of the total product count. To obtain a complete product ranking, we must break ties among products that are added between consecutive solutions to $\mathrm{LR}(\lambda)$. We employ a product's *revenue impact*, the total revenue or orders containing the product, as a tie-breaking metric. This metric proved to be the best approximation to RCO among the heuristics we tried (see Figure 9.1).

Our original implementation of RCO used a linear programming solver (CPLEX) to solve the series of problems $\mathrm{LR}(\lambda)$. However, for very large problems containing millions of order line items, each such problem can take several minutes to solve. To solve it for many values of λ in order to create a dense efficient frontier can take many hours. Large problems called for a more efficient approach to solve the series of problems $\mathrm{LR}(\lambda)$.

9.2.1.3 Relationship to Maximum Flow Problem

We learned that the problem $\mathrm{LR}(\lambda)$ for fixed λ is an example of a *selection problem* introduced independently in Balinski [4] and Rhys [25]. The former paper showed that a selection problem is equivalent to the problem of finding a minimum cut in a particular bipartite network. To see how $\mathrm{LR}(\lambda)$ can be viewed as a minimum cut problem, consider the network in Figure 9.2. Adjacent to the source node s is a set of nodes, each corresponding to one product. Adjacent to the sink node t is a set of nodes, each corresponding to one order. The capacity of the links adjacent to s

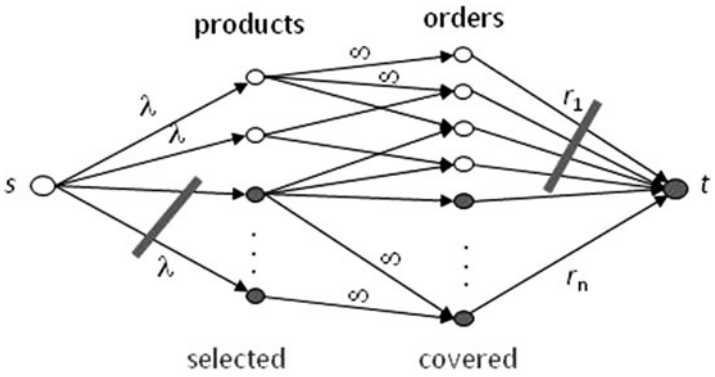

Fig. 9.2 A bipartite minimum cut/maximum flow problem corresponding to the Lagrangian relaxation LR(λ).

is λ. The capacity of the link from the node for order i is the revenue of order i. The capacity of links between product nodes and order node is infinite.

For the network shown in Figure 9.2, the set T in a minimum cut corresponds to the products selected and orders covered by an optimal solution to LR(λ). To see why, first observe that since the links from product nodes to order nodes have infinite capacity, they will not be included in a finite capacity cut. Therefore, for any order nodes in the T set of a finite capacity cut, each product that is in the order must also have its node in T. So a finite capacity cut corresponds to a feasible solution to LR(λ). Moreover, the value of an s–t cut is $\sum_o r_o(1-y_o)+\lambda\sum_p x_p$; in other words, the revenue of the orders *not* covered by the portfolio, plus λ times the number of products in the portfolio. Minimizing this quantity is equivalent to maximizing $\sum_o r_o y_o - \lambda\sum_p x_p$; therefore a minimum cut is an optimal solution to LR(λ).

It is a well-known result of Ford and Fulkerson [11] that the value of a maximal flow equals the value of a minimum cut. Moreover, the minimum cut can be obtained by finding a maximal flow.

If λ is allowed to vary, the problem LR(λ) becomes a parametric maximum flow problem, since the arc capacities depend on the parameter λ. There are several known algorithms for parametric maximum flow, such as those in Gallo et al. [12] for general networks and Ahuja et al. [1] for bipartite networks. In most prior algorithms for parametric maximum flow, a series of maximum flow problems is solved, and previous problem's solution is used to speed up the solution to the next one. By comparison, the HP Labs team developed a new parametric maximum flow algorithm for bipartite networks that finds the maximum flow for all *breakpoints* of the parameter values simultaneously (Zhang et al. [28], Tarjan et al. [30–32]). If we look at the maximum flow from the source s to the target t as a scalar function of the parameter λ, this maximum flow is a piecewise linear function of λ. A breakpoint of the parameter value is where the derivative of the piecewise linear function changes.

9.2.1.4 Parametric Bipartite Maximum Flow Algorithm

As mentioned above, the problem LR(λ) is equivalent to finding a feasible assignment of flows in the graph that maximizes the total flow from s to t. The SPMF algorithm takes advantage of the special structure of the capacity constraints.

The intuition behind the algorithm is as follows. First assume that λ=∞. Then the only constraints on flows result from the capacity limitations on arcs incident to t. It is easy to find flow assignments that saturate all capacitated links, resulting in a maximum total flow.

The next step is to find such a maximum flow assignment that distributes flows as evenly as possible across all arcs leaving s. The property "evenly as possible" means that it is impossible to rebalance flows between any pair of arcs in such a way that the absolute difference between these two flows decreases. Note that even in this most even maximum flow assignment, not all flows will be the same.

Now, with the most even assignment discussed above, impose capacity constraints of $\lambda < \infty$ on the arcs leaving s. If the flow assignment for one of these given arcs exceeds λ, reduce the flow on this arc to λ and propagate the flow reduction appropriately through the rest of the graph.

Since the original flow assignment was most evenly balanced, the total flow lost to the capacity constraint is minimal and the total flow remaining is maximal for the given parameter λ.

More formally, the algorithm works as follows:

Step 1. For a graph as in Figure 9.2 with $\lambda = \infty$, select an initial flow assignment that saturates the arcs incident to t. This is most easily done backward, starting from t and choosing an arbitrary path for a flow of size r_i from t through o_i to s.

Step 2. Rebalance the flow assignment iteratively to obtain a "most evenly balanced" flow assignment. Let $f(a \to b)$ denote the flow along the link from node a to node b. The rule for redistributing the flows is as follows. Pick i and j for which there exists an order node o_k as well as arcs $p_i \to o_k$ and $p_j \to o_k$ such that $f(s \to p_i) < f(s \to p_j)$ and $f(p_j \to o_k) > 0$. Then, reduce $f(s \to p_j)$ and $f(p_j \to o_k)$ by $\min\{(f(s \to p_j) - f(s \to p_i))/2, f(p_j \to o_k)\}$ and increase $f(s \to p_i)$ and $f(p_i \to o_k)$ by the same amount. Repeat *Step* 2 until no such rebalancing can be found.

The procedure in *Step* 2 converges, as proven in Zhang et al. [30, 31]. The limit is a flow assignment that is "most evenly balanced." In addition, since total flow is never reduced, the resulting flow assignment is a maximum flow for the graph with $\lambda = \infty$.

Step 3. To find a maximum flow assignment for a given value of λ, replace flows exceeding λ on arcs leaving the source s by λ and reduce subsequent flows appropriately to reconcile flow conservation. The resulting flow assignment is a maximum flow for λ.

For more details and a rigorous mathematical treatment of the problem, see Zhang et al. [31]. In Zhang et al. [30] it is shown that the algorithm generalizes to the case where arc capacities are a more general function of a single parameter.

In addition, since our application requires only knowledge of the minimum cut, one only needs to identify those arcs that exceed the capacity limit of λ after *Step* 2. Those arcs will be part of the minimum cut, and the ones leaving s with flows less than λ will not. To find the remaining arcs that are part of the minimum cut, one only has to identify which order nodes connect to s through one of the arcs with flows less than λ and cut through those nodes' arcs to t.

As discussed earlier, a bipartite minimum cut/maximum flow problem corresponds to the Lagrangian relaxation problem LR(λ). It can be shown that the t-partition of the minimum cut with respect to λ contains products whose flows from the source equals λ and the orders containing only those products. These products constitute the optimal portfolio for parameter λ.

Note that *Steps* 1 and 2 are independent of λ. The result of *Step* 2 allows us immediately to determine the optimal portfolio for any value of λ.

Since the flows are balanced between two arcs $s \rightarrow p_i$ and $s \rightarrow p_j$, in the algorithm described above, we call it arc-balancing method. Arc-balancing SPMF reduced the time for finding the entire efficient frontier from hours to a couple of minutes.

Another version of SPMF algorithm was developed based on the idea of redistributing the flows going into a node o in a single step so that for all pairs $p_i \rightarrow o$ and $p_j \rightarrow o$, flows $f(s \rightarrow p_j)$ and $f(p_j \rightarrow o_k)$ are "most evenly balanced." This method of redistributing flows around a vertex o is named vertex-balancing method [32]. Vertex-balancing SPMF further reduces the time for finding the entire efficient frontier to seconds.

9.2.1.5 Comparison to Other Approaches

Because the Lagrangian relaxation skips some portfolio sizes in its series of solutions, the worst-case difference between the RCO coverage and the optimal integer program's coverage can be significant. This can be illustrated through a simple example with four products and three orders shown in Table 9.1. The solutions to the integer program, Lagrangian relaxation, and RCO for this example are shown in Table 9.2. In this example, solving the Lagrangian relaxation LR(λ) for any $\lambda \in$[0,21/4] generates the portfolio {1, 2, 3, 4}; any larger value of λ yields the empty portfolio. Portfolio sizes 1, 2, and 3 are skipped and the corresponding revenue covered is zero. RCO invokes the revenue-impact heuristic to break ties among products, thereby achieving better coverage than the Lagrangian relaxation alone.

Table 9.1 A simple example of order data

Order	Products	Order Revenue
A	{1,2,3}	$12
B	{3,4}	$6
C	{1}	$3

Table 9.2 Solutions to example problem for several approaches

	Integer Program		Lagrangian Relaxation		RCO	
Portfolio Size	Solution	Revenue Covered	Solution	Revenue Covered	Solution	Revenue Covered
1	{1}	$3	skipped	$0	{3}	$0
2	{3,4}	$6	skipped	$0	{1,3}	$3
3	{1,2,3}	$12	skipped	$0	{1,2,3}	$12
4	{1,2,3,4}	$21	{1,2,3,4}	$21	{1,2,3,4}	$21

While this example illustrates worst-case behavior, in practice, RCO typically performs very close to optimal because the Lagrangian relaxation skips few solutions when applied to large order data sets from HP's business. RCO also has the added benefit of producing a nested subset of solutions, which is not true in general of the series of solutions to the integer program. Moreover, RCO compares favorably to other heuristics for ranking products (Figure 9.1). The gray curves show the cumulative revenue coverage achieved by four heuristic product rankings, in comparison to the coverage achieved by RCO. The best alternative to RCO for typical data sets is one that ranks each product according to its *revenue impact*, a metric our team devised to represent the total revenue of orders in which the product appears. The revenue-impact heuristic comes closest to RCO's coverage curve, because it is best among the heuristics at capturing product interdependencies. Still, in our empirical tests, we found that the revenue-impact ranking provides notably less revenue coverage than RCO's ranking. Given that RCO runs in less than 2 min for typical data sets and requires no more data than the heuristics, HP had no reason to settle for inferior coverage.

Another advantage of the RCO model is in its data requirements. Unlike metrics based on individual product performance, RCO does not require the metric associated with orders to be broken down to individual products in the order. This is an advantage in applying RCO to real-world data, where it is often difficult to break down an order-level metric to the product level.

9.2.1.6 Generalizations

While the discussion thus far has emphasized the application of maximizing historical revenue coverage subject to a constraint on portfolio size, this approach is flexible enough to accommodate a much wider range of objectives, such as coverage of order margin, number of orders, or any other metric associated with individual orders. It can easily accommodate up-front strategic constraints on product inclusion or exclusion. RCO can also be applied at any level of the product hierarchy, from SKUs down to components. Moreover, our algorithm has broader applications, such as in the selection of parts and tools for repair kits, terminal selection in transportation networks, and database record segmentation. Each of these problems can be naturally formulated as a parametric maximum flow problem in a bipartite network.

The SPMF algorithm has applications well beyond product portfolio management, such as in the selection of parts and tools for repair kits, terminal selection in transportation networks, and database record segmentation. The team's extension of SPMF to non-parametric max flows in general networks (Tarjan et al [28]) has an even broader range of applications in areas such as airline scheduling, open pit mining, graph partitioning in social networks, baseball elimination, staff scheduling, and homeland security.

9.2.1.7 Implementation

HP businesses typically use the previous 3 months of orders as input data to RCO, because this duration provides a representative set of orders. Significantly longer horizons might place too much weight on products that are obsolete or nearing end of life. When analysis on longer horizons is desired, RCO allows weighting of orders in the objective, thus placing more emphasis on covering the most recent orders in a given time window.

The RCO tool was not meant to replace human judgment in the design of the product portfolio. Portfolio design depends critically on knowledge of strategic new product introductions and planned obsolescence, which historical order data do not reveal. Instead, RCO is used to enhance and facilitate interactive human processes that include such strategic considerations.

9.2.2 Results

Various HP businesses have used RCO in different ways to manage their product portfolios more effectively. This section describes benefits obtained in several businesses across HP.

PSG Recommended Offering Program. PSG has used RCO to improve competitiveness by significantly reducing order cycle time. PSG used RCO to analyze order history for the USA, Europe, Middle East and Africa (EMEA), and Asia/Pacific (APJ). RCO revealed that roughly 20% of products, if optimally selected, would completely fulfill 80–85% of all customer orders. When these 20% of items are stocked to be ready-to-ship, they help significantly decrease order cycle time for a majority of orders. Using this insight, PSG established Recommended Offering for each region.

Today, the Notebook Recommended Offering ships 4 days faster than the overall Notebook offering. In EMEA, the Desktop Recommended Offering ships on average 2 days faster than the rest of the offering. The savings are impressive. Lower order cycle time improves competitiveness, each day of OCT improvement across PSG saves roughly $50M annually. PSG management estimates they have realized savings of $130M per year in EMEA and the USA. APJ is also anticipating strong benefits as they roll out the program there.

PSG Global Series Offering Program. RCO is used on an ongoing basis by the PSG Global Business Team to define the Global Series Offering for commercial notebooks. The Global Series Offering is the set of products available to HP's largest global customers. As a result of RCO, global customers are now ordering over 80% of their notebook needs from the global series portfolio, compared to 15% prior to the use of RCO. The total notebook business for global customers is $2.6B. PSG estimates the benefits of this 18% increased utilization of the recommended portfolio to be $130M in revenue.

BCS Portfolio Simplification. BCS runs RCO quarterly to evaluate its product portfolio. In the last 2 years, RCO has been used to eliminate 3,300 products from the portfolio of over 10,000 products. BCS Supply Chain Managers estimate that this reduction has resulted in $11M cost savings due to reduced inventory and planning costs. Moreover, BCS has used RCO to design options for new product platforms based on order history for previous generation platforms.

9.2.3 Summary

The coverage metric provides a new way to evaluate product portfolios. Coverage looks beyond the individual performance of products and considers their interdependence through orders, which is particularly important in configurable product businesses. This metric, and HP Labs' accompanying optimization tool, RCO, enables HP to identify products most critical to its offering, as well as candidates for discontinuance. As a result, HP has improved its operational focus on key products while also reducing the complexity of its product offering, leading to improved execution, significant cost savings, and increased customer satisfaction.

9.3 Wisdom Without the Crowd

Forecasting has been important since the dawn of business. Fundamentally, it is an exercise of using today's information to predict tomorrow's events. The popular approach, backed up by decades of development of computing technologies, is the use of statistical analysis on historical data. This approach can be very successful when the relevant information is captured in historical data.

In many situations, there is either no historical data or the data contain no useful pattern for forecasting. A good example is the forecast of the demand of a new product. A new approach is to tap into tacit and subjective information in the minds of individuals. Groups consistently perform better than individuals in forecasting future events. This so-called wisdom of crowds phenomenon has been documented over the centuries. The prediction market, where people are allowed to interact in organized markets governed by well-defined interaction rules, was shown to be an effective way to tap into the collective intelligence of crowds. Real-world examples include the Hollywood Stock Exchange and the Iowa Electronic Markets. There

are also several companies providing services of conducting prediction markets for business clients.

Prediction markets generally involve the trading of state-contingent securities. If these markets are large enough and properly designed, they can be more accurate than other techniques for extracting diffuse information, such as surveys and opinion polls. However, there are problems, particularly in the context of business forecasting. In particular, a market works when it is efficient. That is, the pool of participants is large enough, and there are plenty of trading activities. Forecasting business events, on the other hand, may involve only a handful of busy experts, and they do not constitute an efficient market.

Here, we describe an alternate method of harnessing the distributed knowledge of a small group of individuals by using a two-stage mechanism. This mechanism is designed to work on small groups, or even an individual. In the first stage, a calibration process is used to extract risk attitudes from the participants, as well as their ability to predict given outcome. In the second stage, individuals are simply asked to provide forecasts about an uncertain event, and they are rewarded according to the accuracies of their forecasts. The information gathered in the first stage is then used to de-bias and normalize the reports gathered in the second stage, which is aggregated into a single probabilistic forecast. As we show empirically, this nonlinear aggregation mechanism vastly outperforms both the imperfect market and the best of the participants. This technique has been applied to several real-world demand forecasting problems. We will present a case study of its use in demand forecasting of a technology hardware product and also discuss issues about real-world implementations.

9.3.1 Mechanism Design

We consider first an environment in which a set of N people have private information about a future event. If information across individuals is independent, and if the individuals truthfully reveal their probability beliefs, then it would be straightforward to compute the true aggregated, posterior, probabilities using Bayes' rule. If the individual i receives independent information then the probability of an outcome s, conditioned on all of their observed information I, is given by

$$P(s\,|\,I) = \frac{p_{s_1} p_{s_2} \cdots p_{s_N}}{\sum_{\text{all } s} p_{s_1} p_{s_2} \cdots p_{s_N}}, \tag{9.1}$$

where p_{s_i} is the probability that individual i predicts outcome s. This result allows us simply to take the individual predictions, multiply them together, and normalize them in order to get an aggregate probability distribution.

However, individuals do not necessarily reveal their true probabilistic beliefs. For that, we turn to *scoring rule mechanisms*. There are several proper scoring rules (for example, Brier [8]) that will solicit truthful revelation of probabilistic beliefs from risk-neutral payoff maximizing individuals. In particular we use the information entropy score. The mechanism works as follows. We ask each player to report a vector of perceived state probabilities $\{q_1, q_2, \ldots q_N\}$ with the constraint that the

vector sums to one. Then the true state x is revealed and each player paid $c_1 + c_2 \log(q_x)$, where c_1 and c_2 are positive numbers. It is straightforward to verify that if an individual believes the probability to be $\{p_1, p_2, \ldots, p_N\}$ and he or she maximizes the expected payoff, he or she will report $\{q_1 = p_1, q_2 = p_2, \ldots, q_N = p_N\}$.

Furthermore, there is ample evidence in the literature that individuals are not risk-neutral payoff maximizers. In most realistic situations, a risk-averse person will report a probability distribution that is flatter than their true beliefs as they tend to spread their bets among all possible outcomes. In the extreme case of risk aversion, an individual will report a uniform probability distribution regardless of their information. In this case, no predictive information is revealed by the report. Conversely, a risk-loving individual will tend to report a probability distribution that is more sharply peaked around a particular prediction, and in the extreme case of risk-loving behavior a subject's optimal response will be to put all the weight on the most probable state according to their observations. In this case, their report will contain some, but not all the information contained in their observations.

In order to account for both the diverse levels of risk aversion and information strengths, we add a first stage to the mechanism. Before each individual is asked to report their beliefs, their risk behavior is measured and captured by a single parameter. In the original research, and subsequent experiments that validated the effectiveness of the mechanism, we use a market mechanism, designed to elicit their risk attitudes and other relevant behavioral information. We use the portfolio held by individuals to calculate their correction factor. The formula to calculate this factor is determined empirically and has little theoretical basis.[2]

The aggregation function, after behavioral corrections, is

$$P(s \mid I) = \frac{p_{s_1}^{\beta_1} p_{s_2}^{\beta_2} \cdots p_{s_N}^{\beta_N}}{\sum_{\text{all } s} p_{s_1}^{\beta_1} p_{s_2}^{\beta_2} \cdots p_{s_N}^{\beta_N}}, \tag{9.2}$$

where β_i is the exponent assigned to individual i. The role of β_i is to help recover the true posterior probabilities from individual i's report. The value of β_i for a risk-neutral individual is one, as this individual should report the true probabilities coming out of their information. For a risk-averse individual, β_i is greater than one so as to compensate for the flatter distribution that such individuals report. The reverse, namely β_i smaller than one, applies to risk-loving individuals. The technique of soliciting this behavior adjustment parameter β_i has evolved over time. In some of the later applications, surveys were used for initial estimations and the estimates were updated using historical performance measures. Finally, a learning mechanism was used to only aggregate the best performing individuals on a moving average basis.

[2] In terms of both the market performance and the individual holdings and risk behavior, a simple functional form for β_i is given by $\beta_i = r(V_i/\sigma_i)c$, where r is a parameter that captures the risk attitude of the whole market and is reflected in the market prices of the assets, V_i is the utility of individual i, and σ_i is the variance of their holdings over time. We use c as a normalization factor so that if $r = 1$, β_i equals the number of individuals. Thus the problem lies in the actual determination of the risk attitudes both of the market as a whole and of the individual players.

9.4 Experimental Verification

A number of experiments were conducted at Hewlett-Packard Laboratories in Palo Alto, CA, to test this mechanism. Since we do not observe the underlying information in real-world situations, a large forecast error can be caused by either a failure to aggregate information or the individuals having no information. Thus, laboratory experiments, where we know the amount of information in the system, are necessary to determine how well this mechanism aggregates information. We use undergraduate and graduate students at Stanford University as subjects in a series of experiments. Five sessions were conducted with 8–13 subjects in each.

The two-stage mechanism was implemented in the laboratory setting. Possible outcomes were referred to as "states" in the experiments. There were 10 possible states, A through *J*, in all the experiments. The information available to the subjects consisted of observed sets of random draws from an urn with replacement. After privately drawing the state for the ensuing period, we filled the urn with one ball for each state, plus an additional two balls for the just-drawn true state security. Thus, it is slightly more likely to observe a ball for the true state than others. We also implemented the prediction market in the experiment, as a comparison.

The amount of information given to subjects is controlled by letting them observe different number of draws from the urn. Three types of information structures were used to ensure that the results obtained were robust. In the first treatment, each subject received three draws from the urn, with replacement. In the second treatment, half of the subjects received five draws with replacement and the other half received one. In a third treatment, half of the subjects received a random number of draws (averaging three, and also set such that the total number of draws in the community was $3N$) and the other half received three, again with replacement.

We compare the scoring rule mechanism, with behavioral correction, to three alternatives: the prediction market, reports from the best player (identified ex post, with behavioral correction), and aggregation without behavioral correction. Table 9.3 summarizes the results.

The mechanism (aggregation with behavioral correction) worked well in all the experiments. It resulted in significantly lower Kullback–Leibler measures than the no information case, the market prediction, and the best a single player could do. In fact, it performed almost three times as well as the information market. Furthermore, the nonlinear aggregation function, with behavioral correction, exhibited a smaller standard deviation than the market prediction, which indicates that the quality of its predictions, as measured by the Kullback–Leibler measure,[3] is more consistent than that of the market. In three of five cases, it also offered substantial improvements over the case without the behavioral correction.

[3] The Kullback–Leibler measure (KL measure) is a relative entropy measure, with respect to the distribution conditioned on all information available in an experiment. A KL measure of zero is a perfect match.

Table 9.3 Kullback–Leibler measure (smaller = better), by experiment

No Information	Prediction Market	Best Player	Aggregation *Without* Behavioral Correction	Aggregation *With* Behavioral Correction
1.977 (0.312)	1.222 (0.650)	0.844 (0.599)	1.105 (2.331)	0.553 (1.057)
1.501 (0.618)	1.112 (0.594)	1.128 (0.389)	0.207 (0.215)	0.214 (0.195)
1.689 (0.576)	1.053 (1.083)	0.876 (0.646)	0.489 (0.754)	0.414 (0.404)
1.635 (0.570)	1.136 (0.193)	1.074 (0.462)	0.253 (0.325)	0.413 (0.260)
1.640 (0.598)	1.371 (0.661)	1.164 (0.944)	0.478 (0.568)	0.395 (0.407)

9.5 Applications and Results

This mechanism was implemented into a web application called BRAIN (Behaviorally Robust Aggregation of Information in Networks). The process is used for forecasting tasks in several companies including a major European telecommunication company and several divisions of the largest technology company in the USA. Participants enter their reports through a web site. The behavioral corrections are carried out automatically and management can access the results directly from the web site.

A project was started in spring 2009 to make use of this process to forecast sales of a technology product. Two business events are to be forecasted. The first is the worldwide monthly shipment units of this product. This product sells into two different customer segments (designated A and B). The second is the percentage of the worldwide shipment going into customer segment A for a particular month.

For each event (for example, worldwide shipment in September 2009), there are six forecasts, two in each month for the 3 months leading up to the event. The forecasts are typically conducted in the first and third week of the month. For the September 2009 shipment, the forecasting process is conducted in late June, twice in July, twice in August and in early September. Note that partial information about shipment of September is available when the forecasting process is conducted. The design allows the forecasts to be updated if new information is available to the individuals. For each event, the real line is divided into distinct intervals and each interval is considered a possible outcome. Individuals are asked to "bet" (report) on each of the possible interval. Twenty-five individuals from different parts of the business organization, including marketing, finance, and supply chain management functions, were recruited for this process. The first forecast was conducted in late May 2009. Participation fluctuated. In the forecasts conducted in early August 2009, 16 out of the 25 recruits (64%) submitted their reports. A small budget was authorized as incentive to pay the participants.

The following figure shows the predictions and the actual events for July 2009. The predictions for Shipments and Customer Segment A have varied over the course of the predictions. The ranges are the bin widths. Prediction starts with the Early June forecasts, beginning about 7 weeks prior to the actual event.

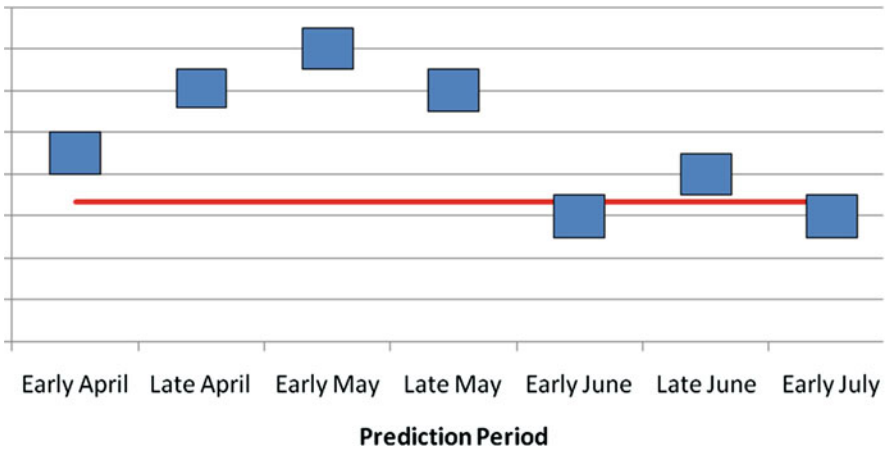

Fig. 9.3 Shipment forecast (units not available). *Note: Rectangles*: most likely interval; *thick line*: actual outcome

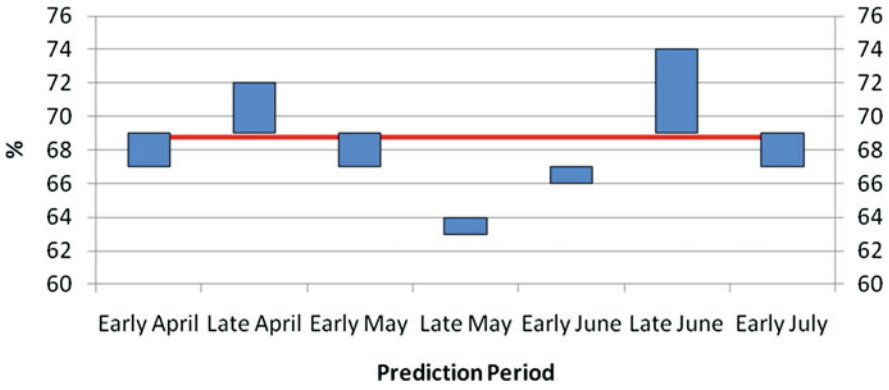

Fig. 9.4 Customer Segment A % forecast. *Note: Rectangles*: most likely interval; *thick line*: actual outcome

As one can see, the BRAIN process has provided accurate forecast at least 1 month in advance for the shipment prediction and 3 months in advance for July consumer percentage. BRAIN is also more accurate in comparison to other internal business forecasts. In particular, the shipment forecasts made 1 month prior for each month from May through July had an absolute error of 2.5% using BRAIN vs. an absolute error of 6.0% for the current forecasting method.

9.6 Modeling Rare Events in Marketing: Not a Rare Event

A rare event is an event with a very small probability of occurrence. Rare event data could be of the form where the binary dependent variable has dozens to thousands of times fewer ones ("events") than zeros ("nonevents"). Typical examples

of such events from social sciences that readily come to mind are wars, outbreak of infections, breakdown of a city's transport system, or levies. Past examples of such events from marketing are in the area of database marketing (e.g., catalogs, newspaper inserts, direct mailers sent to a large population of prospective customers) where only a small fraction (less than 1%) responded resulting in a very small probability of a response (event) [6, 18]. The examples of rare events where they occur infrequently over a period of time can be thought of as *longitudinal rare events*, while the examples where a small subset of the population responds can be thought of as *cross-sectional rare events*.

More recent examples of rare events have emerged in marketing with the advent of the Internet and digital age and the use of new types of marketing instruments. A firm can reach a large population of potential customers through its web site, display ads, e-mails, and search marketing. But only a very small proportion of those exposed to these instruments respond. For example, of the millions of visitors to a firm's web site only a handful of them click on a link or make a purchase. To make business and policy planning more effective it is important to be able to analyze and predict these events accurately.

Rare event variables have been shown to be difficult to predict and analyze. There are two sources to the problem. The first source is that standard statistical procedures, such as logistic regression, can sharply underestimate the probability of rare events. The intuition is that there are very few values available for the independent variables to understand the circumstances that cause an event and these few values do not fully cover the tail of the logistic regression. The model infers that there are fewer circumstances under which the event will occur resulting in an underestimate. Additionally, parametric link functions such as those used for probit or logit assume specific shapes for the underlying link functions implying a given tail probability expression that remains invariant to observed data characteristics. As a result these models cannot adjust for the case when there are not enough observations to fully span the range needed for estimating these link functions. The second source of the problem is that commonly used data collection strategies are grossly inefficient for rare events data. For example, the fear of collecting data with too few events leads to data collections with huge numbers of observations but relatively few, and poorly measured, explanatory variables, such as in international conflict data with more than a quarter-million dyads, only a few of which are at war [6, 16, 18].

Researchers have tried to tackle the problem of using logistic regression (or probit) to analyze rare events data in three ways [6]. First approach is to adjust the coefficients and predictions of the estimated logistic regression model. King and Zeng [18] describe how to adjust the maximum likelihood estimates of the logistic regression parameters to calculate approximately unbiased coefficients and predictions. Second approach is to use choice-based sampling where the sample is constructed based on the value of the dependent variable. This can cause biased results (sample selection bias) and corrections must be undertaken. Manski and Lerman [21] developed the weighted exogenous maximum likelihood (WESML) estimator for dealing with the bias. Third approach is to relax the logit or probit parametric link assumptions which can be too restrictive for rare events data. Naik and Tsai

[24] developed an isotonic single-index model and developed an efficient algorithm for its estimation.

In this study we apply the second approach of choice-based sampling to discrete-choice models and decision-tree algorithms to estimate the response probabilities at the customer level to a direct mail campaign when the campaign sizes are very large (in millions) and the response rates are extremely low. We use the predicted response probabilities to rank the customers which will allow the business to run targeted campaigns, identify best and at-risk customers, reduce their cost of running the campaign, and increase response rate.

9.6.1 Methodology

9.6.1.1 Choice-Based Sampling

In a discrete-choice modeling framework sometimes one outcome can strongly outnumber the other such as when many households do not respond (e.g., to a direct mailing). Alternative sampling designs have been proposed. A case–control or choice-based sample design is one in which the sampling is stratified on the values of the response variable itself and disproportionately more observations are sampled from the smaller group. This ensures that the variation in the dependent variable is maximized with subsequent statistical analysis accounting for this sampling strategy to ensure the estimates are asymptotically unbiased and efficient [10, 21, 22].

In the biostatistical literature, case–control studies were prompted by studies in epidemiology on the effect of exposure to possible hazards such as smoking on the risks of contracting a disease condition. In a prospective study design, a sample of individuals is followed and their responses recorded. However, many disease conditions are rare and even large studies may produce few diseased individuals (cases) and little information about the hazard. In a case–control study separate samples are taken of cases and controls—individuals without the disease [27].

In the economics literature, estimation of models to understand choices for travel modes or recreation sites has used different sampling designs to collect data on consumer choices. For example, studies of participation levels and destinations for economic activities such as recreation have traditionally been analyzed using random samples of households, with either cross-section observations or panel data on repeat choices obtained from diaries. In travel demand analysis, an alternative sampling design is to conduct intercept surveys at sites. This can result in substantial reductions in survey costs and guarantee adequate sample sizes for sites of interest, but the statistical analysis must take into account the "choice-based" sample frame [23].

There is a well-developed theory for this analysis in the case of cross-section observations, where data are collected only on the intercept trip. In site choice models when subjects are intercepted at various sites, a relevant statistical analysis is

the theory of estimation from choice-based samples due to Manski and Lerman [21] and Manski and McFadden [22]. This theory was developed for situations where the behavior of a subject was observed only on the intercept choice occasion and provided convenient estimators when all sites were sampled at a positive rate. One of these estimators, called weighted exogenous sample maximum likelihood (WESML), reweights the observations so that the weighted sample choice frequencies coincide with population frequencies. A second, called conditional maximum likelihood (CML), weights the likelihood function so that the weighted sample choice probabilities average to the sample choice frequencies. The WESML setup carries out maximum pseudolikelihood estimation with a weighted log likelihood function where in conventional choice-based sampling the weights are the sampling rates for the alternatives, given by the sample frequency divided by the population frequency for each alternative. The CML setup carries out maximum conditional likelihood estimation with a log likelihood function.

However, recently sampling schemes have emerged in the literature on recreational site choice that combine interception at sites with diaries that provide panel data on intercept respondents on subsequent choice occasions. McFadden's [23] paper provides a statistical theory for these "Intercept and Follow" surveys, and indicates where analysis based on random sampling or simple choice-based sampling requires correction.

9.6.1.2 Modeling Approach

We developed a discrete-choice (logit) model and a classification-tree algorithm (aucCART) for predicting a user's probability of responding to an e-mail. The discrete-choice model is statistical based while the classification-tree algorithm is machine-learning oriented. Both response modeling methods use as input dozens of columns (or attributes) from the data sample and identify the most important (relevant) columns that are predictive of the response. By employing different types of response models for predicting the same response behavior, we were able to cross-check the models and discover predictors and attribute transformations that would be overlooked and missed in a single model. We then performed hold-out (or out-of-sample) tests on the accuracy of both methods and select the best model.

The output of each model consists of the probability that each customer will respond to a campaign and the strength of each attribute that influences this probability. We extracted about 80 explanatory attributes from the transaction and campaign databases. These may be broadly classified as (1) customer static (nontime-varying) attributes such as gender and acquisition code; (2) customer dynamic attributes just prior to the campaign, which include the recency, frequency, and monetary (RFM) attributes for customer actions, responses to previous campaigns, etc.; and (3) campaign attributes such as the campaign format and the offer type (e.g., fixed price and percentage discounts, free shipping, and freebies).

Choice-Based Sampling

A typical campaign gets very low response rate. To learn a satisfactory model, we would need thousands of responses and hence millions of rows in the training data set. Fitting models with data of this size requires a considerable amount of memory and CPU time. To solve this problem, we used choice-based sampling [21]. The idea is to include all the positive responses (Y=1) in the training data set, but only a fraction f of the non-responses (Y=0). A random sample, in contrast, would sample the same fraction from the positive responses and the negative responses. Choice-based sampling dramatically shrinks the training data set by about 20-fold when f = 0.05. To adjust for this "enriched" sample, we used case weights that are inversely proportional to f. We found that this technique yields the same results with only a very slight increase in the standard errors of the coefficients in the learned model [10].

Discrete-Choice Logit Model

The logit (or logistic regression) model is a discrete-choice model for estimating the probability of a binary response (Y=1 or 0). In our application, each user i is described by a set of static attributes $X_s(i)$(such as gender and acquisition source); each campaign j is described by a set of attributes $X_c(j)$ (such as campaign offer type and message style type); each user has dynamic attributes $X_{\mathrm{d}}(i,j)$ just before campaign j (such as recency of action, i.e., the number of days between the last action and the campaign start date). Our pooled logit model postulates

$$P\{Y(i,j)=1\} = \frac{\exp[X_s(i)\beta_s + X_c(j)\beta_c + X_d(i,j)\beta_d]}{1+\exp[X_s(i)\beta_s + X_c(j)\beta_c + X_d(i,j)\beta_d]}.$$

A numerical optimization procedure finds the coefficient vectors (β_s, β_c, β_d) that maximize the following weighted likelihood function:

$$L = \prod_{i=1}^{N} [P\{Y(i,j)=1\}]^{Y(i,j)} [1-P\{Y(i,j)=1\}]^{[1-Y(i,j)]/f},$$

where f is the choice-based sampling fraction.

Decision-Tree Learner aucCART

We developed a new decision-tree model, aucCART, for scoring customers by their probability of response. A decision tree can be thought of as a hierarchy of questions with Yes or No answers, such as "Is attribute1 $>$ 1.5?" Each case starts from the root node and is "dropped down the tree" until it reaches a terminal (or leaf) node; the answer to the question at each node determines whether that case goes to the left or right sub-tree. Each terminal node is assigned a predicted class in a way that

minimizes the misclassification cost (penalty). The task of a decision-tree model is to fit a decision tree to training data, i.e., to determine the set of suitable questions or splits.

Like traditional tree models such as CART (Classification and Regression Trees) [7], aucCART is a non-parametric, algorithmic model with built-in variable selection and cross-validation. However, traditional classification trees have some deficiencies for scoring:

They are designed to minimize the misclassification risk and typically do not perform well in scoring. This is because there is a global misclassification cost function, which makes it undesirable to split a node whose class distribution is relatively far away from that of the whole population, even though there may be sufficient information to distinguish between the high- and low-scoring cases in that node. For example, assume that the two classes, say 0 and 1, occur in equal proportions in the training data and the costs of misclassifying 0 as 1 and 1 as 0 are equal. Suppose that, while fitting the tree, one finds a node with 80% 1s (and 20% 0s) which can be split into two equally sized children nodes, one with 90% 1s and the other with 70% 1s. All these nodes have a majority of 1s and will be assigned a predicted class of 1; any reasonable decision tree will not proceed with this split since it does not improve the misclassification rate. However, when scoring is the objective, this split is potentially attractive since it separates the cases at that node into a high-scoring group (90% 1s) and a lower-scoring group (70% 1s).

A related problem is the need to specify a global misclassification cost. This is not a meaningful input when the objective is to score cases.

The aucCART method is based on CART and is designed to avoid these problems. It combines a new tree-growing method that uses a local loss function to grow deeper trees and a new tree-pruning method that uses the penalized AUC risk $R_\alpha(T) = R(T) + \alpha|T|$. Here, the AUC risk $R(T)$ is the probability that a randomly selected response scores lower than a randomly selected non-response, $|T|$ is the size of the tree, and α is the regularization parameter, which is selected by cross-validation. This method is (even) more computationally intensive than CART, in part because it runs CART repeatedly on subsets of the data and in part because minimizing the penalized AUC risk requires an exhaustive search over a very large set of sub-trees; in practice, we avoid the exhaustive search by limiting the search depth. Our numerical experiments on specific data sets have shown that aucCART performs better than CART for scoring.

9.6.2 Empirical Application and Results

9.6.2.1 Background

Customers continue to use e-mails as one of their main channels for communicating and interacting online. According to Forrester Research (2007) 94% of online customers in the USA use e-mails at least once a month. Customers also ranked opt-in

e-mails among their top five sources of advertisements they trust for product information (Forrester Research 2009). E-mail marketing has become an important part of any online marketing program. In fact, according to the 2007 Forrester Research report, 60% of marketers said that they believe marketing effectiveness of e-mail as a channel of communication will increase in the next 3 years.

An HP online service with millions of users uses e-mail marketing as one of their marketing vehicles for reaching out to its customers with new product announcements and offers. In general, each e-mail campaign is sent to all users and on a regular basis with millions of customers contacted during any specific campaign. One drawback of this "spray-and-pray" approach is the increased risk of being blacklisted by Internet Service Providers (ISPs) when they receive too many complaints. In addition to direct loss of revenue when an e-mail program is stopped early, it increases the risks of using e-mail as a regular channel for communication in the future. So the marketing team was interested in methods that would help them to identify who their best customers and "at-risk" customers were and understand what key factors are that drive customer response. This would enable them to send more targeted e-mail campaigns with relevant messages and offers.

9.6.2.2 Data Set and Variables

We selected a subset of past e-mail campaigns from the marketing campaigns database that were representative of (and similar to) the planned future campaigns. We, then, selected a subset of customers from the sent list of these past campaigns. Each campaign had a date–time and a number of attributes associated with it. The campaign date allowed us to "go back in time" and derive the user's behavioral attributes just before each of the past campaigns. We, a priori, split the customers into two customer segments based on whether they did a specific action in the past (in line with the business practice). Table 9.4 gives some descriptive statistics of the two samples.

The outcome variable, response to a campaign, indicates whether or not (1 or 0) the user responded to each of the selected campaigns. For each campaign we used the campaign database to create the campaign-specific attributes. Some examples of these attributes are the e-mail message's subject line, the format of the e-mail, the value offered in the e-mail (percentage discounts, dollar amount of free products, the

Table 9.4 Descriptive statistics of the data samples

Customer Segment	Number of Campaigns	Number of Observations (Customer campaign)	Number of Observations Choice-based Sample	Number of Customers Choice-based Sample
Action-Active	32	4.2 X	0.21 X	0.16 X
Action-Inactive	25	7.8 X	0. 39 X	0.33 X

Note: We depict the sample sizes as multiples of X to anonymize the data

type of product featured), the time-of-the-year occasion of the e-mail timing (such as Christmas shopping season).

For each customer we used the full history of their transactions since registration, available in the transaction database, to create customer-specific attributes just prior to the beginning of each campaign. These attributes included recency (how many days prior to the campaign did the user take an action), frequency (how many times in the month, quarter, or year prior to the campaign did the user undertake an action), and monetary (how much in dollars did the user spend in the month, quarter, or year prior to the campaign and in which product categories). In addition, we used data sources like the US Census Bureau and other sources of first names and gender to create a first-name-to-gender translator which predicted the probability of a person being male or female given the person's first name.

We tried all reasonable transformations of the attributes and selected the ones that yielded the best model. We determined the best transformation by investigating the residual plots for the logit model. Furthermore, the output produced from our classification tree-based aucCART algorithm (which automatically transforms some attributes) also gave us some suggestions for the most appropriate transformations. In the logit model, we selected the final set of attributes by using both forward and backward step-wise selection. In forward selection, we started with a single predictor variable (attribute) and added variables (with appropriate variable transformations) one by one, until no statistically significant variable can be added, or AIC (Akaike Information Criterion) value can be improved. In backward selection, we started with all attributes (properly transformed) included in the model and delete statistically insignificant variables one at a time, until all remaining variables are statistically significant. For the classification tree-based aucCART algorithm, variable selection was automatically performed (a built-in feature of classification tree-based algorithms).

The final data sample had several hundred thousands of data rows (each row represents a user) and approximately 80 columns (each column is an attribute describing the user at various points of time). We randomly selected 50% of the rows in the data sample as training data and the rest as testing data to evaluate the two approaches and select the best one.

9.6.2.3 Validation, Model Selection, and Results

We validated our models on holdout data sets with different customers and campaigns than the training data. Our holdout tests were designed to simulate an in-the-field application of our models to existing and new customers and new campaigns.

In addition to the two approaches outlined, various heuristics or scoring rules have been commonly used by marketing professionals to predict responses and selecting target recipients. One such heuristic for selecting recipients is by action recency, which ranks recipients by the most recent to least recent in their last action;

the more recent a user's action is, the higher the probability of responding to an e-mail the heuristic predicts. We used the action recency heuristic as the baseline of what the business is using and compare it to our two approaches.

To evaluate various rules, models, and algorithms, we needed a metric that is applicable to a wide variety of models, and that is also relevant to how the models will be used.

Figure 9.5 shows a capture curve for each model or scoring rule. The capture curve measures the percentage (Y-axis) of positive responses captured (in a holdout data set) if the model is used to select a given percentage (X-axis) of customers. The capture curves indicate that the logit model approach was the most effective in predicting and capturing customer responses to e-mails than the simple RFM method (action recency) or the decision-tree approach. For example, the logit model for action-active users is able to capture 92.1% of the campaign responses by selecting only the top 50% of the users.

The model results also indicated the strongest predictors of customer response. We are not sharing those numbers to preserve business confidentiality. In general, recency of action, the dollar amount of the user's past purchases, and the user's recorded responses to prior e-mail campaigns were significant. Additional predictors were gender, e-mail format, and offer type.

HP business group is incorporating the scoring model into their customer segmentation strategy for e-mail marketing. One of the key findings was that the business can generate 90% of the total expected response by contacting just the top 50% of users. By identifying this high-response half of its user base, they will be able to (1) tailor the message content and frequency to specific user segments based on the likelihood of response, (2) greatly increase the average response per message, and (3) reduce the total volume (and cost) of messaging.

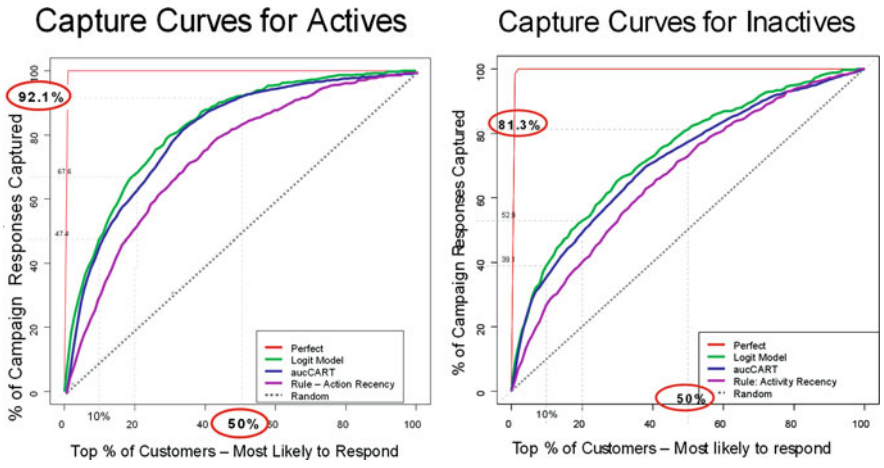

Fig. 9.5 Comparison of capture curves

In future studies we want to see if our conclusions hold for direct mail. Further we want to examine if the customers with low ranks based on the model are also the ones most likely to unsubscribe, complain, and create negative word-of-mouth.

9.7 Distribution Network Design

Hewlett-Packard provides a wide range of products and services for a diverse set of customers located across the globe leveraging a worldwide network of suppliers, partners, and facilities. As the operator of the largest supply chain in the IT industry HP relies on analytical modeling to support many strategic and operational decisions with detailed optimization models enabling evaluation of alternative supply chain strategies—procurement, location, inventory—to investigate opportunities to decrease supply chain-related costs and improve order cycle times. HP has a long tradition of employing operations research for its supply chain problems [20]. Some recent examples include reverse supply chain redesign for Personal Systems Group (PSG) in Europe [14], network design for Imaging and Printing Group (IPG) in Europe [19], production line design for IPG in the USA [9], and inventory management for former network server division [5].

In this section we describe a mathematical programming model that constitutes the core of a number of analytical decision support applications for decision problems ranging from design of manufacturing and distribution networks to evaluation of complex supplier offers in logistics procurement processes. We provide some details on two applications of the model to evaluate various distribution strategy alternatives—to answer questions such as whether it is efficient to add more distribution centers to the existing network and which distribution centers and transport modes are to be used to supply each customer location and segment—by quantifying the trade-off between the supply chain costs and order cycle times.

9.7.1 Outbound Network Design

HP provides personal computers, workstations, handheld computing devices, digital entertainment systems, calculators and other related accessories, software and services for commercial and consumer markets. The customers in the commercial segment include direct customers such as big corporations, small and medium size businesses (SMB), government agencies and indirect channel partners. Supply chain configurations vary by product as well as by customer segments. HP utilizes a number of contract manufacturers (CMs) and original design manufacturers (ODMs) to manufacture certain HP-designed products to generate cost efficiencies and reduce time to market. There are three types of nodes in a typical supply chain: inbound hubs, manufacturing sites, and outbound hubs. The inbound hubs store components and are usually situated close to the manufacturing sites. The inventory at these

locations is owned by the suppliers and is pulled by the manufacturing sites per customer order. For some critical parts, the inventory may also be owned by HP. Once the products are manufactured, they are shipped to outbound hubs (or distribution centers) for further shipment to customer locations. Certain products may also be shipped directly to customers from the manufacturing sites.

The outbound hubs play a number of critical roles in a typical supply chain. First, outbound hubs are used to consolidate shipments from manufacturing sites to customer locations for a portion of the trip. Finished goods are first shipped to an outbound hub in a bulk mode. Individual customer shipments are then scheduled for a shorter distance. Thus outbound hubs are used to leverage from volume of shipments for a particular region. Second, outbound hubs are used to merge shipments from different manufacturing sites into a single shipment to customer locations. Finally, outbound hubs are used to carry finished goods inventory for certain customers with short order cycle time requirements and for certain stable SKUs (e.g., certain standard configurations).

Given the existing and potential outbound hubs, the model is used to seek answers to the following questions: Which of the existing and potential outbound hubs should HP use in its operations? Which customer locations and segments should be assigned to each outbound hub? Which product groups should be assigned to each outbound hub? What should be the mode of transportation in meeting customer demands for each customer location and segment?

The answers to these questions hinge on various aspects of the fundamental tradeoffs between customer service levels and supply chain-related costs. The former is measured by Order to Delivery Time (ODT), the time between customer order and order delivery to customer. The latter, supply chain costs, fall in four broad categories: First, Inventory-Driven Costs (IDC) include all of the costs that derive from the level of inventory in the regions, such as obsolescence and component devaluations. Second, Trading Expenses (TE) include freight, duties, taxes, allocations, and warehousing. Third, Manufacturing Expenses (MOH) include the cost of manufacturing products, as well as any costs related to the support of that manufacturing activity including customization and rework. Finally, Cash to Cash (C2C) takes into account how long inventory is held in the region and how long it takes to pay the suppliers and to receive payment from customers.

ODT is an important metric for a product division's supply chain. Service level agreements with customers usually involve explicit ODT requirements. ODT is composed of several components such as order entry time, material wait time, factory cycle time, and delivery time. Of these components, material wait time and the delivery time are likely to get impacted by the supply chain configuration. Furthermore, for a given supply chain configuration, the three components of ODT—order entry, factory cycle time, and the delivery time—are not likely to change from one order to another (for the same customer location and product group), while the material wait time can be considerably variable depending on the immediate availability of the components at the designated inbound hub. Also note that from the above four components, delivery time is the only component that will be impacted by the outbound strategy. Different customer groups—corporate, small and medium

businesses, public sector, indirect channel partners—may have distinct ODT requirements. Any outbound strategy should ensure that the ODT requirements are satisfied for each customer segment.

Trading expenses and inventory-driven costs are likely to be impacted most by the outbound strategy. Major components of Trading Expenses are transportation costs from manufacturing sites into the outbound hubs and from outbound hubs to the customer locations, material handling costs, and facility costs. Main elements of Inventory-Driven Costs are costs due to inventory in transit from manufacturing sites to the outbound hubs, and from outbound hubs to customer locations, and inventory in the outbound hubs.

The decision problem is to minimize trading expenses and inventory-driven costs while satisfying order to delivery time targets set by management. Various business constraints such as limiting the total number of outbound hubs that will be used, forcing a particular outbound hub to stay open or closed will also need to be incorporated as constraints in the model.

Products can be modeled at the SKU level or at the product category level after aggregation. Customer segments are modeled separately as shipment volumes and ODT requirements vary by segment. For customer locations various levels of aggregation—by state, zip code, etc.—are possible. HP works with many different transportation service providers including parcel carriers, airfreight companies, less-than-truckload (LTL) and full truckload (FTL) carriers. Transportation mode can be modeled using the physical mode of transportation (type of vehicle or type of company) or using delivery times to code transportation modes—e.g., 1-day service, 2-day service, 3-day service.

In order to capture the variability in ODT targets, two modes of delivery are defined. For a fraction θ^r_{js} of orders originating from customer segment s for product j, the order needs to be shipped from the factory with regular delivery within w^r_{js}. Likewise, for a fraction $\theta^e_{js} = 1 - \theta^r_{js}$ of orders originating from customer segment s for product j, the order needs to be shipped from the factory with emergency delivery within w^e_{js}.

9.7.2 A Formal Model

We next introduce the notation needed for a formal presentation of the mathematical model. Let M denote the set of manufacturing sites, I denote the set of potential outbound hub sites, K denote the set of customer locations, S denote the set of customer segments, J denote the set of product groups, and T denote the set of transportation modes available.

The following variables define the parameters of the model:

d_{ksj} : demand in location k for customer segment s for product j

c_{mitksj} : cost to satisfy demand in location k for customer segment s for product j by manufacturing in site m through outbound hub i with transport mode t

ℓ_{mitksj} : delivery time to satisfy demand in location k for customer segment s for product j by manufacturing in site m through outbound hub i with transport mode t
f_i : fixed operating cost of outbound hub site i
C_i : capacity of outbound hub site i
w^r_{js} : time window specified for product j for customer segment s for regular delivery
w^e_{js} : time window specified for product j for customer segment s for emergency delivery
θ^r_{js} : fraction of orders in segment s for product j requiring regular delivery
θ^e_{js} : fraction of orders in segment s for product j requiring emergency delivery

We also define the following variables to be used in the mathematical program:

$$\delta^r_{mitksj} = \begin{cases} 1 & \text{if } \ell_{mitksj} \le w^r_{sj} \\ 0 & \text{otherwise} \end{cases}$$

$$\delta^e_{mitksj} = \begin{cases} 1 & \text{if } \ell_{mitksj} \le w^e_{sj} \\ 0 & \text{otherwise} \end{cases}$$

The following parameters are used to enforce a specific scenario for the network design:

$$\alpha_i = \begin{cases} 1 & \text{if outbound hub } i \text{ needs to be open in a scenario} \\ 0 & \text{otherwise} \end{cases}$$

$$\beta_i = \begin{cases} 1 & \text{if outbound hub } i \text{ needs to be closed in a scenario} \\ 0 & \text{otherwise} \end{cases}$$

$$\gamma_i = \begin{cases} 1 & \text{if outbound hub } i\text{'s capacity needs to be enforced in a scenario} \\ 0 & \text{otherwise} \end{cases}$$

The following variables are the decision variables of the problem:

$$x^r_{mitksj} = \begin{cases} 1 & \text{if segment } s\text{'s regular demand in location } k \text{ for product } j \text{ is satisfied by manufacturing site } m \text{ through outbound hub } i \text{ with mode } t \\ 0 & \text{otherwise} \end{cases}$$

$$x^e_{mitksj} = \begin{cases} 1 & \text{if segment } s\text{'s emergency demand in location } k \text{ for product } j \text{ is satisfied by manufacturing site } m \text{ through outbound hub } i \text{ with mode } t \\ 0 & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if outbound site } i \text{ is used} \\ 0 & \text{otherwise} \end{cases}$$

With the notation introduced the mathematical program is written as

$$\min \sum_{m\in M}\sum_{i\in I}\sum_{t\in T}\sum_{k\in K}\sum_{s\in S}\sum_{j\in J} d_{ksj}c_{mitksj}[\theta^r_{sj}x^r_{mitksj}+\theta^e_{sj}x^e_{mitksj}] + \sum_{i\in I} f_i y_i, \tag{9.3}$$

$$\sum_{m\in M}\sum_{i\in I}\sum_{t\in T}\delta^r_{mitksj}x^r_{mitksj} = 1 \text{ for all } k\in K, s\in S, j\in J, \tag{9.4}$$

$$\sum_{m\in M}\sum_{i\in I}\sum_{t\in T}\delta^e_{mitksj}x^e_{mitksj} = 1 \text{ for all } k\in K, s\in S, j\in J, \tag{9.5}$$

$$x^r_{mitksj} - y_i \le 0 \text{ for all } m\in M, i\in I, t\in T, k\in K, s\in S, j\in J, \tag{9.6}$$

$$x^e_{mitksj} - y_i \le 0 \text{ for all } m\in M, i\in I, t\in T, k\in K, s\in S, j\in J, \tag{9.7}$$

$$y_i \ge \alpha_i \text{ for all } i\in I, \tag{9.8}$$

$$y_i \le (1-\beta_i) \text{ for all } i\in I, \tag{9.9}$$

$$\gamma_i\left(\sum_{m\in M}\sum_{t\in T}\sum_{k\in K}\sum_{s\in S}\sum_{j\in J} d_{ksj}[\theta^r_{sj}x^r_{mitksj} + \theta^e_{sj}x^e_{mitksj}]\right) \le C_i \text{ for all } i\in I, \tag{9.10}$$

$$x^r_{mitksj} \in \{0,1\} \text{ for all } m\in M, i\in I, t\in T, k\in K, s\in S, j\in J, \tag{9.11}$$

$$x^e_{mitksj} \in \{0,1\} \text{ for all } m\in M, i\in I, t\in T, k\in K, s\in S, j\in J, \tag{9.12}$$

$$y_i \in \{0,1\} \text{ for all } i\in I. \tag{9.13}$$

The objective in (9.3) minimizes all incoming and outgoing transportation costs, material handling, inventory, and the facility costs. The constraints in (9.4) and (9.5) ensure that each customer segment in each location is assigned to one outbound site, manufacturing site, and one transportation mode for each product group that are within delivery time requirements for regular and emergency demands, respectively. Note that the product groups from a single customer location and segment can be assigned to different manufacturing sites, outbound hubs, and transportation modes. The constraints in (9.6) and (9.7) ensure that service from an outbound hub is available only if the facility is open. The constraints in (9.8) and (9.9) ensure that the outbound hubs are forced to be open or closed based on the scenario specification.

The constraints in (9.10) ensure that the capacity of the outbound hub is enforced if specified in the scenario. The constraints in (9.11), (9.12), and (9.13) ensure that all decision variables are binary. Note that the formulation in (9.3)–(9.13) assumes that the model is full, e.g., every customer location has demand from all $|S|$ segments and for all $|J|$ product groups. This is only to make the exposition simple. The actual model used in implementation takes advantage of the link sparsity.

9.7.3 Implementation

The model in the previous section was implemented using ILOG's OPL Studio and solved using CPLEX. The raw data are stored in several tables in a database and can be imported from a spreadsheet or a flat file. A typical implementation may involve up to 1,000 customer locations (actual customer locations aggregated at the 3 digit zip code), 5–10 customer segments, 5–10 transport modes, up to 10 product groups, and up to 100 potential outbound hub sites. Standardized forms are used to allow the user to specify the parameters for what–if scenarios in order to see the impact

of several critical variables. Through various forms the user can change the delivery time targets, enforce a particular outbound hub to stay open or closed, activate or deactivate the capacity constraint on a particular hub, and limit the number of outbound hubs. The user sees the results of the model via several reports. Location Summary report shows which outbound hubs are open and what costs are incurred in doing so. Location Usage report shows the total number of units in each product category that flows through each outbound hub. Delivery Performance report shows the resulting average delivery times for each customer segment and product group. Location Customer Assignment report shows the detailed assignment of customer locations/customer segments to manufacturing sites/outbound hubs/transportation modes.

9.7.4 Regarding Data

The data requirements can be categorized into four groups: logistics, financial, demand, and customer service requirements. Some critical data elements need to be estimated from various data sources. Transportation costs and times between manufacturing sites and outbound hubs are estimated assuming that a bulk mode is used and scale economies are fully utilized, considering the typically large volume of shipments. Based on the manufacturing scenario, the shipments can be originating from various locations worldwide. Depending on the origin, the shipments may be made over the ocean by major carriers or by FTL carriers. Cost and time estimates are created using data on rate tables and maps from major carriers. Estimation of transportation costs and times between outbound hubs and customer locations is based on data on shipment histories and representative carrier cost and time information for various weight categories. Annual demands at the product group, customer segment, and customer locations were estimated from data on shipment history. In addition to these three items, data such as material handling and facility setup costs for outbound hubs, unit manufacturing costs (estimates) at different manufacturing sites, inventory holding cost rates and customer service level requirements are obtained from various sources in finance, logistics, and procurement operations.

9.7.5 Exemplary Analyses

The outbound model proved to be very useful for internal consulting teams for evaluating alternative distribution strategies for various product groups. The outbound model was also used as a primary input to the assessment of end-to-end manufacturing scenarios for a product group.

The model described above provides the core for analysis of a number of broader supply chain strategy decisions including the selection for manufacturing sites. The analysis for the outbound strategy clearly depends on the locations of the

manufacturing facilities. For this purpose, viable scenarios included the baseline scenario describing the manufacturing locations at the time of implementation. These manufacturing scenarios specify manufacturing location(s) for each product category.

For each manufacturing scenario, various analyses can be carried out. The first category of analysis takes the current level of OTD targets as input and develops an outbound strategy for each manufacturing scenario. The analysis in this category was used to determine the optimal outbound hub locations and to assess the value of additional outbound hubs for each manufacturing scenario. The analysis proved very useful in understanding the marginal value of each additional outbound hub. An example of this analysis (with fictitious data) is provided in Figure 9.6.

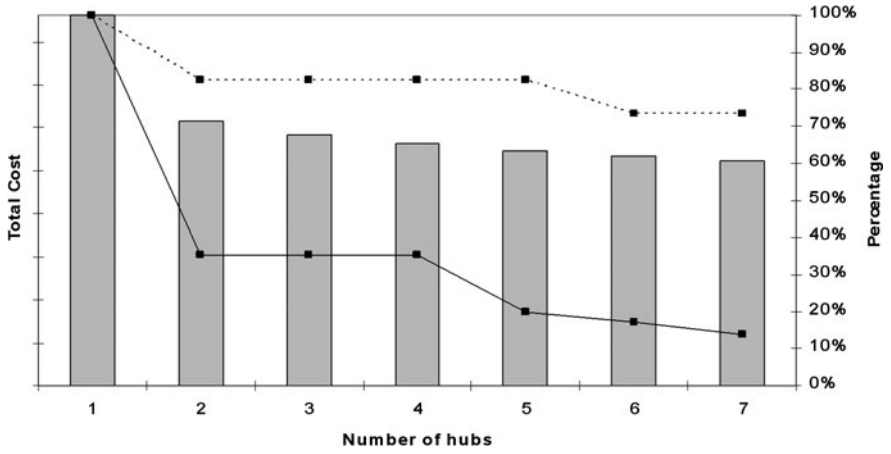

Fig. 9.6 Impact of number of hubs

In the example, we consider a manufacturing scenario with a single manufacturing location co-located with one of the outbound hubs. Since each additional hub provides the flexibility to consolidate a portion of the trip for shipments to customer locations, the total costs decline. However, as expected, there are decreasing marginal returns. Understanding the exact value of each additional outbound hub, together with an evaluation of operational complexity, provides valuable guidance for the management decisions on the number and locations of each outbound hub. In Figure 9.6, we also show the percentage of products shipped directly from the outbound hub co-located with the single manufacturing site for two product groups: bulky products (product group 1) and small/light products (product group 2). Clearly, shipment consolidation is more beneficial for bulkier items (product group 1), and we see that more of this group of products are shipped via additional hubs than the second group. The analysis is also useful in estimating the transportation cost component of different manufacturing scenarios. In addition to the strategic insights, the analysis can also be used to support detailed operational decisions such as which manufacturing sites, outbound hubs, and

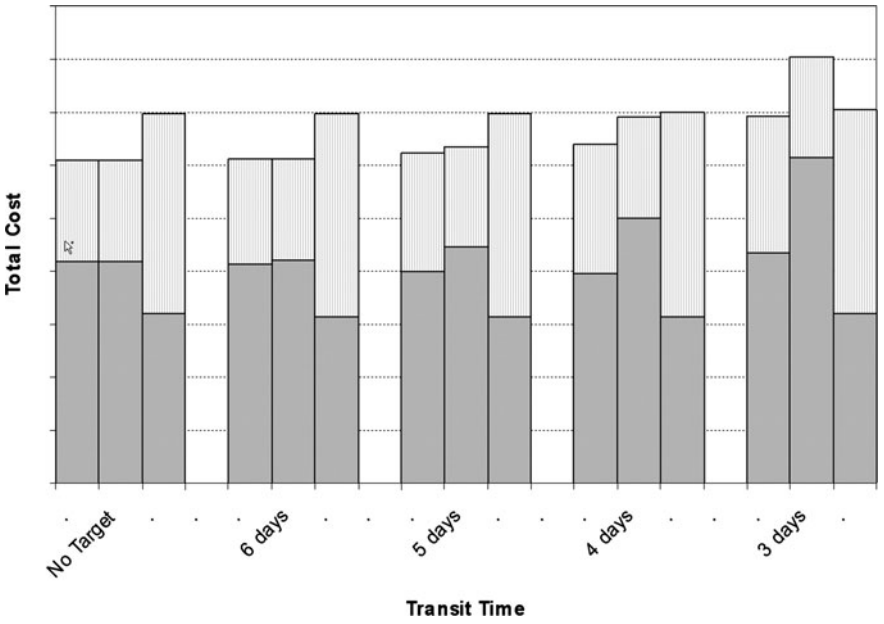

Fig. 9.7 Impact of delivery time targets

transportation modes should be used to deliver orders of a particular region and a customer segment.

The model can also be used to study the trade-off between the supply chain costs and OTD targets. Naturally, this trade-off varies with manufacturing scenarios. In Figure 9.7 we present an example of such analysis (with fictitious, but representative data). In this example, we consider three manufacturing scenarios. In the first scenario, the products can be manufactured in an offshore location as well as locally in the USA. Modifying the per unit costs c_{mitksj} in the mathematical model to include manufacturing costs, the model is used to select a manufacturing site among the set of possible sites for satisfying demand in a particular customer location. In the second scenario, all manufacturing is done at an overseas site. Finally in the third scenario, all manufacturing is done locally in the USA. Each manufacturing scenario is combined with various target delivery time levels starting from a case where there is no time constraint on shipping a customer order.

The analysis reveals that, for all three manufacturing scenarios, the total costs increase as the delivery time targets are more aggressive. The mixed scenario outperforms the other two scenarios for all target levels, since it has the flexibility of employing offshore as well as local manufacturing. For this scenario, as the delivery time targets get more aggressive, more manufacturing is moved to domestic sites. Note also that while the total cost for the offshore-only scenario is quite sensitive to the delivery time targets, the total cost for the US-only scenario is rather insensitive.

9.8 Collaborations and Conclusion

For development and deployment of decision sciences solutions, the HP Labs team works very closely with business units. In addition, in most cases HP's Information Technology (HP IT) group has a significant role in the success of the projects. The HP Labs team takes the ownership of development of underlying algorithms and core algorithmic software engine. HP IT is generally responsible for integration of the core analytical engine with back-end IT systems, database design and development, system architecture, deployment, and support of the complete system.

Over the years, HP Labs' Business Optimization Lab has built strong research collaborations with leading faculty members in several areas of interests to HP and the academic community. The university collaboration for the work presented in this chapter is reflected in the author list.

This chapter covers a very narrow slice of advanced analytics project at HP Labs and at HP. It is safe to say that the creation and application of rigorous mathematical models is well established throughout the company. Applied researchers and practitioners are making contributions that directly impact the top and bottom line.

Acknowledgments

In this chapter, we have summarized the work of several members of the HP Labs and business units of HP. In particular, we are very thankful to Kemal Guler for organizing the content of distribution network design portion of this chapter.

References

1. Ahuja RK, Orlin RB, Stein C, Tarjan RE (1994) Improved algorithms for bipartite network flow. SIAM Journal of Computing 23:903–933
2. Ansari A, Mela CF (2003) E-customization. Journal of Marketing Research XL:131–145
3. Babenko M, Derryberry J, Goldberg A, Tarjan R, Zhou Y (2007) Experimental evaluation of parametric max-flow algorithms. Proceedings of WEA. Lecture Notes in Computer Science 4525. Springer, Berlin–Heidelberg, Germany, pp. 612–623
4. Balinski ML (1970) On a selection problem. Management Science 17(3):230–231
5. Beyer D, Ward J (2002) Network server supply chain at HP: A case study. In: Song J, Yao D (eds) Supply chain structures: Coordination, information and optimization. International Series in Operations Research and Management Science, Kluwer, Norwell, MA
6. Blattberg RC, Kim P, Neslin S (2008) Database marketing: Analyzing and managing customers. Springer, New York
7. Breiman L, Friedman J, Olshen R, Stone C (1984) Classification and regression trees. Chapman and Hall, New York
8. Brier, GW (1950) Verification of forecasts expressed in terms of probability. Mon. Wea. Rev., 78:1–3
9. Burman M, Gershwin SB, Suyematsu C (1998) Hewlett-Packard uses operations research to improve the design of a printer production line. Interfaces 28:24–36

10. Donkers B, Franses PH, Verhoef PC (2003) Selective sampling for binary choice models. Journal of Marketing Research XL:492–497
11. Ford LR, Fulkerson DR (1956) Maximum flow through a network. Canadian Journal of Mathematics 8:339–404
12. Gallo G, Grigoriadis MD, Tarjan RE (1989) A fast parametric maximum flow algorithm and applications. SIAM Journal of Computing 18:30–55
13. Goldberg AV, Tarjan RE (1986) A new approach to the maximum flow problem. Proceedings of the 18th Annual ACM Sympos Theory Computation (Berkeley, CA), May 28–30, pp. 136–146
14. Guide Jr VDR, Mulydermans L, Van Wassenhove LN (2005) Hewlett-Packard company unlocks the value potential from time-sensitive returns. Interfaces 35:281–293
15. Jain S (2008) Decision sciences—A story of excellence at Hewlett-Packard. OR/MS Today, April
16. Kamakura WA, Mela CF, Ansari A, Bodapati A, Fader P, Iyengar R, Naik P, Neslin S, Sun B, Verhoef P, Wedel M, Wilcox R (2005) Choice models and customer relationship management. Marketing Letters 16(3/4):279–291
17. Kamakura WA, Russell GJ (1989) A probabilistic choice model for market segmentation and elasticity structure. Journal of Marketing Research 26(4):379–390
18. King G, Zeng L (2001) Logistic regression in rare events data. Political Analysis 9(2):137–163
19. Laval C, Feyhl M, Kakouros S (2005) Hewlett-Packard combined or and expert knowledge to design its supply chains. Interfaces 35:238–247
20. Lee HL, Billington C (1995) The evolution of supply chain management models and practice at Hewlett-Packard company. Interfaces 25:42–46
21. Manski CF, Lerman SR (1977) The estimation of choice probabilities from choice based samples. Econometrica 45(8)(November):1977–1988
22. Manski CF, McFadden D (1981) Structural analysis of discrete data with econometric applications. MIT, Cambridge, MA
23. McFadden D (1996) On the analysis of "Intercept and Follow" surveys. Working Paper. University of California, Berkeley, CA
24. Naik PA, Tsai CL (2004) Isotonic single-index model for high-dimensional database marketing. Computational Statistics & Data Analysis 47(4):775–790
25. Rhys JMW (1970) A selection problem of shared fixed costs and network flows. Management Science 17(3):200–207
26. Rossi PE, McCulloch RE, Allenby GM (1996) The value of purchase history data in target marketing. Marketing Science 15(4):321–340.
27. Scott AJ, Wild CJ (1986) Fitting logistic models under case-control or choice based sampling. Journal of the Royal Statistical Society 48(2):170–182
28. Tarjan R, Ward J, Zhang B, Zhou Y, Mao J (2006) Balancing applied to maximum network flow problems. Proceedings of the ESA, Lecture Notes in Computer Science 4168, pp. 612–623
29. Ward J, Zhang B, Jain S, Fry C, Olavson T, Mishal H, Amaral J, Beyer D, Brecht A, Cargille B, Chadinha R, Chou K, DeNyse G, Feng Q, Padovani C, Raj S, Sunderbruch K, Tarjan R, Venkatraman K, Woods J, Zhou J (2010) HP transforms product portfolio management with operations research. Interfaces 40(1):17–32
30. Zhang B, Ward J, Feng Q (2004) A simultaneous parametric maximum flow algorithm for finding the complete chain of solutions. HP Technical Report: HPL-2004-189, Palo Alto, CA
31. Zhang B, Ward J, Feng Q (2005a) Simultaneous parametric maximum flow algorithm for the selection model. HP Technical Report HPL-2005-91, Palo Alto, CA
32. Zhang B, Ward J, Feng Q (2005b) Simultaneous parametric maximum flow algorithm with vertex balancing, HP Technical Report HPL-2005-121, Palo Alto, CA