UNIVERSITY OF SOUTHAMPTON

FACULTY OF SOCIAL, HUMAN AND MATHEMATICAL SCIENCES

Mathematics

Estimation and Pricing for Substitutable Products in Choice-Based Revenue Management

by

Yalin Bi

Supervisor: Dr. Christine Currie Prof. Joerg Fliege

Thesis for the degree of Doctor of Philosophy

October 2017

UNIVERSITY OF SOUTHAMPTON

<u>ABSTRACT</u>

FACULTY OF SOCIAL, HUMAN AND MATHEMATICAL SCIENCES
Mathematics

<u>Doctor of Philosophy</u>

ESTIMATION AND PRICING FOR SUBSTITUTABLE PRODUCTS IN CHOICE-BASED REVENUE MANAGEMENT

by Yalin Bi

It has been proved that choice-based Revenue Management can result in significant increases in revenue in situations where a seller is pricing a set of substitutable products. This is particularly applicable to the transport industry and we present an example of train ticket sales.

Estimating customer choice models is difficult, particularly in situations where the data file is incomplete. We use the Multinomial Logit (MNL) model to describe customer preferences, and a two-step algorithm to jointly estimate the parameters of this model and the customer arrival rate. A simple Markov Chain Monte Carlo (MCMC) method is also applied to update our belief of arrival rate and customer choice model.

The dynamic programming model for the choice-based pricing problem suffers from the "curse of dimensionality ". The computational time increases dramatically and makes it impossible to solve the problem with exact solutions. Approximate dynamic programming methods can be used to solve the problem. We propose a new approximation method that reduces the running time. The thesis will describe the complete methodology that we have implemented and provide some numerical results.

As these are live sales systems, it is important that the system continues to earn revenues while the parameters are being estimated. A decision-making problem is needed to maintain a balance between the learning of customer preference (exploration) and earning (exploitation) in choice-based Revenue Management. In order to maximise the total revenue, the seller must decide whether to choose the current optimal price (exploitation) or to set prices that help to better estimate customer choice behaviour (exploration). We propose two pulling policies in a Multi-armed Bandit (MAB) experiment to balance the trade-off between exploration and exploitation.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, Yalin Bi , declare that the thesis entitled *Estimation and Pricing for Substitutable Products in Choice-Based Revenue Management* and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- none of this work has been published before submission

Signed:...

Date:...

Acknowledgements

First and foremost I am grateful to my advisors, Dr. Christine Currie and Professor Joerg Fliege for their guidance, support and empirically infinite patience over the last four years. I wish to thank them for all the feedback and advice they gave to me. They help me to improve my presentation skill and the enlightening discussions every week brought me closer to the solutions.

I would like to thank Dr. Yue Wu for providing helpful feedback and advice on my upgrade report. I also wish to thank Dr Houduo Qi for his encouragement. Thanks to all my colleagues at Cormsis group. I have learnt a lot from each one of them. In addition, a thank you to Sophie N'jai, who help me to proofread this thesis.

To my friends: David Yang, Shuanghua Bai, Sohana Jahan, Mol Chaiwuttisak, Jiabin Luo, Yi Qu and Qingyun Li.

I also thank my parents, parents-in-law and brother for their permanent support and encouragement.

All of this would not have been possible without the love of my husband Yan Huang. Thanks to everyone else who has helped me.

Chapter 1

Introduction

In many applications, a seller has to make decisions regarding its pricing or quantity control policies. The pricing policies include how to set a price at the beginning of the selling period, when and by how much to discount a product, and so on. The quantity control policies include how to decide the initial inventory level, whether to accept a certain offer from a customer, whether to offer a product to the market, and so on. Revenue management (RM) encompasses these two areas of quantity control and price optimisation and has been an active area of research for several decades (see Talluri and van Ryzin (2006) for an introduction to RM). Compared to the quantity control policies that control sales by limiting supply, it is more profitable for a firm to apply a pricing policy that controls the sales by increasing prices (Gallego and van Ryzin (1994)). In addition, the development of Internet has given firms greater flexibility to change the prices of their products. In this study, we apply a dynamic pricing policy with the aim of maximising the expected revenue.

Traditionally, there is a common assumption in RM that the demand for a product will not be affected by the availability or price of other products in the market. This is called the independent demand model. However, this assumption is not always true. For example, when buying rail tickets customers consider trains with the same origin-destination pair to be substitutable. If we assume a customer's first choice is a train on Wednesday afternoon, he/she may choose a train on Wednesday morning if the ticket price is much cheaper than the cost of the ticket on Wednesday afternoon, or if the Wednesday afternoon ticket is not available. In these situations, when setting prices it is important to model how a customer is making decisions about which ticket to buy. The RM models that consider customer choice behaviour are called choice-based RM. Vulcano et al. (2010) report potential revenue gains of 1%-5% if the seller converts to a choice-based RM system.

In this thesis, we study a choice-based dynamic pricing problem with demand and substitution behaviour uncertainty. In the problem, a seller offers a set of substitutable products to the customers. Each customer chooses a product individually from the set of products on offer, by employing a certain customer choice behaviour. The result of customer selection depends on the price and other characteristics of the products in the offer set. We assume the customer choice behaviour can be modelled with a Multinomial Logit (MNL) model and the parameters in the MNL model are fixed but unknown to the seller. The seller can estimate the parameters by offering different price vectors for products and observing the choices of customers (exploration). Then, the seller uses the estimation of customer choice behaviour in a choice-based dynamic pricing problem to find the optimal price vector and improve their profit (exploitation). The ratio and order of exploration and exploitation can be controlled by a Multi-armed bandit (MAB) algorithm.

The methods we propose in this work can be applied in many areas. The estimation method can solve the customer preference estimation problem for the companies in transportation, airlines, fashion retailing, and many other areas in which it is difficult to replenish the inventory. The new approximation method can benefit the companies which need to update optimal prices frequently due to large volumes of inventory and customers. These applications can be found in a range of companies, including railway companies, the focus of this research. The proposed policies in MAB can be applied in the Internet retailing sector - for example, Amazon and Alibaba, and in the area of transportation. The policies are specifically appropriate for the companies that already have their own estimation and pricing systems.

1.1 Overview

We decompose the problem into two parts. The first part is the estimation of arrival rate and customer choice models. The second part finds the optimal set of prices to charge for a given set of parameters, by solving a choice-based RM problem. Then we solve the trade-off between the estimation and optimisation with a MAB algorithm.

In order to apply a choice-based RM system, the seller needs a good estimate of the demand and customer choice behaviour from the sales data. Given the finite set of alternatives, the choice behaviour is typically described by a discrete choice model, with the MNL model being the most widely used. The MNL model can be estimated using maximum likelihood if complete data are available. However, when we consider the situation that some products may be sold out before the end of the selling period, we cannot determine whether a customer would have chosen this product if the full product set was available. In addition, the arrivals of customers who buy products are recorded in the data, but the data of arrivals without a purchase are unobservable in sales data. Hence, the sales data we have are incomplete and the maximum likelihood method cannot be used directly. In this situation, more information is needed to reduce the uncertainty in the incomplete data. Vulcano et al. (2012) assume they have an exogenous estimate of the aggregate market share and estimate the unknown parameters with an Expectation-Maximisation (EM) method. Assuming the market share information is available, we show that we can jointly estimate the arrival rate and customer choice model with maximum likelihood estimation directly, thus avoiding the use of the EM method.

Another approach to estimating parameters is to make use of Bayesian statistics. This has the benefit of allowing us to take any prior beliefs about the parameter values into account. Compared with point estimation, a Bayesian method also provides more information about the unknown parameters. Previously, Letham et al. (2015) solved this estimation problem using a stochastic gradient Markov Chain Monte Carlo (MCMC) algorithm, and the parameter of base utility is unidentifiable. We show that, given the information about market share, this problem can be solved with a MCMC algorithm with unique solution for the parameter of base utility.

After we obtain the estimates of customer choice behaviour, we can use this information in a dynamic pricing problem to find the optimal price vector for the products. Finding an exact solution to the problem is computationally intractable for practical-sized problems; therefore, Zhang and Cooper (2009) propose an approximate dynamic programming method to reduce the running time of the original dynamic program problem. For examples where the price vector must be updated regularly, the computation time for the approximation method in Zhang and Cooper (2009) can still be too long. We provide a new approximate dynamic programming method which further reduces the running time.

A MAB algorithm aims to achieve the right balance between exploration and exploitation. We use a MAB algorithm to choose when to offer diverse price vectors that help us learn more about the parameter values and when to offer the current set of optimal prices. If the firms spend too much time in estimation (exploration), they may lose the opportunity to offer the optimal price for a sufficient period of time, thus maximising the short-term revenue. If they spend most of the time using the optimal price vector which is based on the current estimation (exploitation), they may lose the chance to find a better understanding of choice behaviour that affect the results of optimal price and the long-term revenue. Besbes and Zeevi (2009) follow a learning-and-earning pattern and Broder and Rusmevichientong (2012) propose a MLE-CYCLE policy which performs the learning phase and earning phase in a cycle. Schwartz et al. (2016) apply an Upper Confidence Bound algorithm to balance the exploration and exploitation. We propose two methods in the Multi-armed Bandit (MAB) to solve the trade-off between exploration of demand and substitution behaviour uncertainty and exploitation of short-term profit.

In Chapter 3, we jointly estimate the arrival rate and parameters where choice behaviour is described by the MNL model. We have adapted Newman's two-step method to consider the problems with stockout effect. With information of market share, we can avoid the EM algorithm in Vulcano et al. (2012) and solve the problem with a globally concave function. We also apply a MCMC method to solve the estimation problem with information of market share. Different from Letham et al. (2015), our MCMC method provides a unique solution for the parameter of the base utility.

We describe a simulation study that measures estimation performance and how much data are needed to have good estimates. The comparison with the EM method introduced by Vulcano et al. (2012) is also presented. Compared with the EM methods that need to calculate expected value and solve optimisation problems iteratively, maximum likelihood estimation is quick and can be solved with many existing software programs. The Bayesian method can provide the posterior distribution on the unknown parameters and this will afford the user a better understanding of the uncertainty of the parameters. Compared with the stochastic gradient MCMC method in Letham et al.(2015), a standard MCMC method is easier to implement with existing software programs.

In Chapter 4, we propose a new approximation method to solve the choice-based dynamic pricing problem with substitutable products. Compared to the approximation method in Zhang and Cooper (2009), our method can reduce the computation time significantly, particularly for the problems with a large number of time periods or a high inventory level. The approximation method can benefit firms that need to update their pricing policy frequently.

In Chapter 5, we propose two new policies in MAB—RP policy and RP-a policy—to solve the trade-off between exploration and exploitation. The policies are easy to implement with any existing estimation methods and optimising algorithms. The RP-a policy can adjust the ratio of exploration and exploitation automatically based on the status of estimation. We also provide numerical results to show the performance and compare these two policies with three alternative policies. The results suggest that a firm can improve their profit by applying a RP-a policy instead of following standard policies of learning-then-earning policy or passive learning policy.

Figure 1.1 presents the structure of our model.

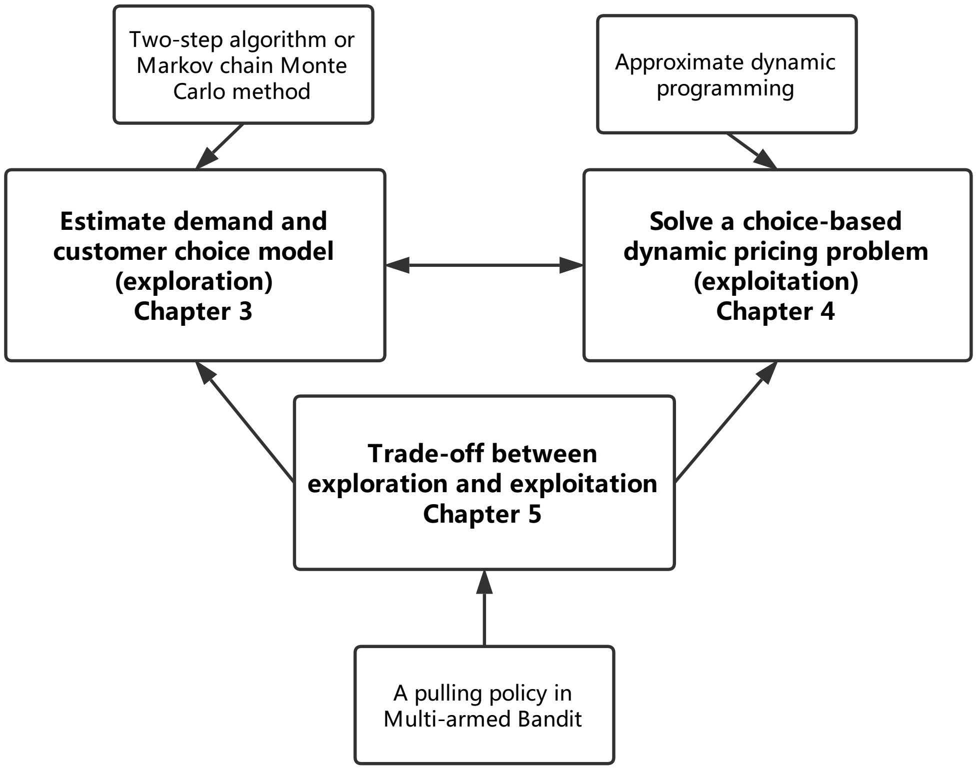


Figure 1.1: Model Structure

1.2 Research Challenges and Contributions

The overall aim of the thesis is to solve a choice-based dynamic pricing problem for substitutable products with demand and substitution behaviour uncertainty. There are three main research challenges in the project.

1. Estimate the parameters of a customer choice model with incomplete data.
2. Reduce the computation time associated with finding the optimal prices to charge for a choice-based RM problem.

3. Develop a pricing strategy that balances the trade-off between estimation and exploitation in choice-based RM.

Related to the first challenge, Vulcano et al. (2010) estimate the parameters of MNL with an EM method. They assume the inventory level is unlimited and no information of market share is provided. Considering the same problem, Newman et al. (2014) decompose the estimation into two parts with a two-step algorithm. Different from the problem solved in these papers, Vulcano et al. (2012) assume the aggregate market share is available and estimate the preference weights of the MNL model with an EM method. In the paper, they assume that products may be sold out before the end of the selling period. This problem has not been solved with the two-step algorithm. Given the aggregate market share and incomplete sales data, we identify a research gap in how to estimate the parameters in the MNL model with direct maximisation. We fill this gap by decomposing the estimation with a two-step algorithm and estimate the parameters of MNL with the maximum likelihood method or the MCMC method.

Zhang and Cooper (2009) address a choice-based RM problem by approximating the value function with separable bounds. However, the computation time of the method prevents the user from updating the prices over a short time period with large instances. No method has been provided to improve the computation time of their method. We fill the gap by regarding the bounds as a value function of the single product dynamic programming problem and providing new bounds of the value functions.

The third challenge has been considered in Schwartz et al. (2016), which applies an Upper Confidence Bound algorithm to balance the exploration and exploitation. The algorithm cannot be applied to the existing pricing and estimation method directly. Apart from the learning and then earning pattern applied in Besbes and Zeevi (2009), Broder and Rusmevichientong (2012) provide a MLE-CYCLE policy. Both of these policies can solve the trade-off between existing pricing and the estimation phase. However, an algorithm in MAB is still needed to improve the performance of existing algorithms. We address the gap by designing efficient pulling algorithms that satisfy our requirements.

Given the research challenges and research gaps summarised above, we make the following contributions in this thesis.

- We show that the estimation of primary demand can be solved with maximum likelihood or the MCMC method with unique solution when the aggregate market share is available.
- We propose an approximation method for the choice-based dynamic pricing problem which was shown in our numerical tests to reduce the computational time of the existing approximation method.
- We propose two pricing policies, RP and RP-a, which solve the trade-off between estimation and choice-based dynamic pricing.

1.3 Outline

The remainder of this thesis is set out as follows:

- Chapter 2 reviews the papers relating to estimation of choice model, pricing problems and MAB problems. In Section 2.1, we introduce widely used choice models and review the papers that estimate the customer choice model in different areas. In Section 2.2, we introduce the papers that focus on dynamic pricing problems and dynamic pricing problems with demand uncertainty. We also give a detailed review of the choice-based RM problem with substitutable products or network structure. Finally, we review the papers with different pulling policies in Multi-armed Bandit experiments.
- Chapter 3 discusses the estimation methods of the MNL model. In Section 3.1, we describe the use of the EM methods that jointly estimate the arrival rate and MNL model for the problems with or without stockout effect. Section 3.2 shows that the problem with stockout effect can be solved with a two-step algorithm if the information of market share is available. In Section 3.3, we show that this estimation problem can be solved with a simple MCMC method. Numerical experiments are presented in Section 3.4 and assess the quality of the different estimation techniques.
- Chapter 4 contains the formulation of a choice-based dynamic programming problem. We propose a new approximate dynamic programming method to reduce the running time of the existing approximation methods. The numerical results show the computation time comparison and revenue comparison of these two approximation methods with a large instance.

- In Chapter 5, we propose two MAB algorithms to solve the trade-off between exploration and exploitation. We draw comparisons between these two algorithms and three standard methods.
- Chapter 6 concludes the thesis. It states the findings and results and provides directions for future work.

Chapter 2

Literature review

Revenue Management (RM) is an approach used by commercial businesses which aims to maximise their revenue by making appropriate management decisions over resources which have limited capacity. A comprehensive introduction to RM can be found in Talluri and van Ryzin (2006). RM methods rely on good estimates of parameter values that provide an accurate description of the system being optimised.

In Section 2.1, we review some discrete choice models and the estimation methods in different application areas. Section 2.2 focuses on the literature that covers the dynamic pricing problem with or without demand uncertainty, as well as papers that consider the RM problems with a customer choice model. In Section 2.3, we introduce the pulling strategies for the Multi-armed Bandit problem, before concluding in Section 2.4.

2.1 Estimation of the Customer Choice Model

2.1.1 Customer Choice model

Customer choice behaviour is usually modelled with two categories of discrete choice models: reservation price model and random utility model. A reservation price model assumes that each customer has a reservation price v for each of the products and will not buy the product until the price of the product p is below their reservation price. The probability that a customer chooses to buy a product is given by

$$F(p) = P(p < v).$$

A random utility model assumes that the customer has a utility for each product. The term, 'utility' is used to measure the attractiveness of a product and is defined as a function of its attributes, which is the factor considered by customers. For example, for a problem in transportation, the factor under consideration can be departure time, departure date, the cost of travel, or the indicator variable for a first-class ticket.

The random utility model assumes the probability that a customer chooses a product is equal to the probability that this product has the highest utility among all the products in the offer set. Let U_i denote the utility of product i in offer set S. The probability that a customer chooses product i is given by

$$P_i(S) = P(U_i = \max\{U_j : j \in S\}).$$

The multinomial logit (MNL) model (Ben-Akiva and Lerman, 1985) is the most widely used customer choice model in operational research, marketing and travel demand forecasting. It describes the probability that a customer chooses a certain product from the products set. A detailed review of the MNL model can be found in Ben-Akiva and Lerman (1985) and Train (2009). As we will see later, it is also the most common model for use in RM; for example, Liu and van Ryzin (2008), Zhang and Adelman (2009), Dong et al. (2009) and Vulcano et al. (2010).

The MNL model belongs to the set of random utility models, which assumes the customer utilities for alternatives are random variables. Specifically, let U_i denote the utility of a customer for alternative i, which can be defined as $U_i = u_i + \xi_i$, where u_i is the deterministic portion, which represents the utility of observable attributes, like travel time, travel day, cost, and first-class ticket or not. Here, ξ_i is the random portion, which describes the utility of unobservable attributes; for example reliability of the service, ease of transfer and condition of the coach.

Random utility models differ in the assumptions that they make about the distribution of the ξ_i; the interactions between decision-makers and the different alternatives and the correlation structure between the ξ_i. We consider the MNL model, which assumes the ξ_i is an independent and identically distributed random variable with a Gumbel distribution and the cumulative distribution function of the random part is

$$F(\xi_i) = \exp(-\exp(-\xi_i)).$$

An alternative option is the multinomial probit model (considered by Grammig et al., 2005 in the RM area), where the ξ_i follows a joint normal distribution. This provides more flexibility with regard to condition structure but is harder to manipulate.

For the MNL, we assume that customers are homogeneous in their preferences; the probability that a customer chooses product i from offer set S is given by

$$P_i(S) = \frac{e^{u_i}}{\sum_{j \in S} e^{u_j} + e^{u_0}} = \frac{e^{u_i}}{\sum_{j \in S} e^{u_j} + 1},$$

and the no-purchase probability can be given by

$$P_0(S) = \frac{1}{\sum_{j \in S} e^{u_j} + 1},$$

where the assumption that the utility associated with no purchase $u_0 = 0$.

We assume u_i has a linear-in-parameters function as

$$u_i = \beta x_i,$$

where β is an unknown vector of parameters we need to estimate, and x_i is a vector of attributes for option i. We measure utility relative to a base or reference option, which is assumed to have a base utility of the β_0. The purchase probability can then be presented by

$$P_i(S) = \frac{\exp(\beta x_i)}{\sum_{j \in S} \exp(\beta x_j) + 1}. \tag{2.1}$$

Note that the ratio of probabilities for two alternatives is given by

$$\frac{P_i}{P_j} = \frac{\frac{e^{u_i}}{\sum_{k \in S} e^{u_k} + 1}}{\frac{e^{u_j}}{\sum_{k \in S} e^{u_k} + 1}} = \frac{e^{u_i}}{e^{u_j}},$$

which means that the ratio will not be affected by the offer sets which contain these two alternatives. This property is called the independence-from-irrelevant-alternatives (IIA) property and it is often illustrated by an example called "red bus/blue bus" (Ben-Akiva and Lerman, 1985).

Finite-mixture logit models and nested logit models are two kinds of model that can avoid the IIA property by making assumptions about the correlation structure in the

MNL models. The Nested Logit (NL) model is a relaxation of MNL model, which allows correlation between alternatives. In the NL model, the alternatives can be divided into exclusive nests. The alternatives in the same nest are correlated with each other and there is no correlation between two alternatives which are in different nests. Therefore, the IIA property holds within groups but not across groups. This model can be applied to the problems which include choice of destination or choice of travel frequency for different purposes (Ben-Akiva and Lerman, 1985). The papers that consider the nested logit model can be found in Li and Huh (2011), Gallego and Wang (2014), Li et al. (2015) and Rayfield et al. (2015). We do not use this choice of model, because the problem we study does not include this kind of choice.

The finite-mixture logit model is another relaxation of the MNL model. Different from the assumption in the MNL model that customers have the same preference over the choices, finite-mixture logit models allow customers to have different preferences. This model assumes that customers who have the same preferences belong to a segment and that the preference in each segment can be modelled with MNL. In our work, we assume the preferences of customers are homogeneous. We do not consider the finite-mixture logit model and apply the MNL model to model the customer preference.

In a finite-mixture logit model, it is assumed that the customers can be divided into L segments and the probability that a customer is in segment l is given by

$$q_l = \frac{e^{w_l}}{\sum_{j=1}^{L} e^{w_j}}, l = 1, \cdots, L,$$

where w_l is the parameter associated with segment l. Assume that the coefficient for attributes in segment l is β_l. The probability of a customer choosing alternative i is given by

$$P_i(S) = \sum_{l=1}^{L} q_l \frac{\exp(\beta_l x_i)}{\sum_{k\in S} \exp(\beta_l x_k)}, i \in S.$$

2.1.2 Estimation Methods for Choice Model

In this section, we review the estimation methods in different application areas. Then wenfocus on the papers which consider the problem with incomplete data.

In the transportation field, choice models have been used to estimate demand of transportation route, destination and different transportation facilities. Bhat (2001) estimates a mixed multinomial logit model with quasi-random maximum simulated likelihood estimation and applies the approach to an intercity travel mode choice problem. Wen and Koppelman (2001) also solve a similar intercity choice problem. They choose to apply a constrained maximum likelihood method to estimate a generalised nested logit model. Hess et al. (2006) modify the Latin Hypercube Sampling method to estimate a mixed logit model and apply the method to find customer preference between different types of vehicle. Yang et al. (2014) estimate a MNL model by solving the first-order derivatives of the log likelihood function with Quasi-Newton methods.

In the economics field, Berry (1994) proposes a methodology which inverts the market-share equation to estimating the discrete choice model with aggregate data. Ackerberg and Rysman (2002) propose a choice model, which adjusts the standard logit model to account for the unobserved product characteristics. For individual-level data, they estimate the model with the maximum likelihood method. For group-level data, the Berry (1994) inversion is used to estimate the model.

The literature focusing on estimation of discrete choice models is also applicable to the marketing field. The basic estimation method is maximum likelihood estimation. Morrow-Jones et al. (2004) apply this method to estimate the customer preference over neighbourhood design characteristics. The preference is modelled with a probit choice model. Geottler and Shachar (2001) use a maximum simulated likelihood method to estimate a discrete choice model with unobserved characteristics and apply the method in the network television industry. Another stream of estimation method is Bayesian estimation. Arora and Huber (2001) propose a hierarchical Bayes choice model and estimate the model with a Gibbs sampler. Similarly, Daziano and Bolduc (2013) apply a Markov Chain Monte Carlo Gibbs sampler to estimate a hybrid choice model. The approach is implemented to explain the environmental attribute in customer vehicle choice.

In most cases in RM, researchers only have access to sales data, which record the customers who arrive and purchase a product. This means that they cannot distinguish a time period in which there is an arrival but no purchase occurred from a time period without an arrival. If we just ignore the customers who arrive into the selling system and choose to leave without a purchase, we will have a severe bias in estimation. In statistical terms, these are referred to as constrained data. Methods that are designed to estimate true parameters for demand distribution or choice model from constrained sales data are called unconstraining methods. A focus on unconstraining methods with single produce can be found in work by Guo et al. (2012).

The unconstraining method for the problem of multiple products is the Expectation-Maximisation (EM) algorithm. The EM algorithm is an iterative algorithm that estimates parameters with incomplete data. The algorithm was first introduced by Dempster et al. (1977) and has been widely used in the revenue management area (Talluri and van Ryzin, 2004, Vulcano et al., 2010, Vulcano et al., 2012). Ratliff et al. (2008) propose a recapture heuristic to unconstraining data. Talluri (2009) provides a log risk-ratio estimation method to account for the unconstraining data. Newman et al. (2014) propose a two-step method to solve the unconstraining data problem. A detailed review is provided below.

Talluri and van Ryzin (2004) provide an EM method to jointly estimate the choice parameters and arrival rate with incomplete data for an example in the airline industries. The EM algorithm consist of two steps: the Expectation (E) step and the Maximisation (M) step. The Expectation step calculates the expected value of the conditional log likelihood of the missing data. The Maximisation step calculates the maximum value of the expected log-likelihood function. The parameters maximising the function are recorded and used in the next expectation step. These two steps are performed iteratively until the estimation stops improving or a stopping criterion is reached. They update the belief of parameters by calculating the expectation value of the log-likelihood function with the current estimation and maximise the expected log-likelihood function iteratively. Vulcano et al. (2010) provide a simulation study of a choice-based RM problem, again from the airline industry, which uses the EM algorithm to obtain the estimation of parameters. In these two papers, the EM method is applied with a fixed start point. The quality of estimation is highly dependent on the setting of the start point, as we show later in Chapter 3.

To improve the computational speed of the estimation in Talluri and van Ryzin (2004), Newman et al. (2014) present a new algorithm which uses a marginal log-likelihood function and split the estimation procedure into two steps. The first step is estimating parameters in the MNL model, except the parameter for base utility. Using the estimates from the first step, the parameters for base utility and the arrival rate are estimated in the second step. Unfortunately, multiple maxima may still exist in the second step. They provide methods for discrete and time processes. This method can be adopted in the problem with more than one offer set. We extend the idea of this two-step method to jointly estimate parameters in MNL and arrival rate in a problem with the situation of stockout and substitution behaviour, as described in Section 3.2.

The above three papers assume that products are in stock during the whole selling period. With a different assumption, the aim of Vulcano et al. (2012) is to estimate substitution effects with stockout effect. They transfer the problem to the estimation of customer primary demand. In the paper, they jointly estimate the preference weights and arrival rates with sale, product-availability data and market size. The MNL model is used as the customer choice model and arrivals of customers are assumed to be non-homogeneous Bernoulli arrivals over multiple periods. An EM method is applied to estimate the model parameters. In the paper, the authors report that the incomplete data likelihood function has a continuum of maxima, and they solve this problem by adding constraints on the parameters.

Unlike the assumption in Vulcano et al. (2012) that the market size is known, Talluri (2009) estimates market size with provided data. The author proposes a finite population model to discard the Poisson demand assumption and allow a wider range of demand distributions. However, it assumes that the parameter for the no-purchase option is known before the estimation, which restricts the application of this method. The estimation of market size can be regarded as a binomial estimation problem which is challenging; this paper provides an estimation heuristic with log risk-ratio for the MNL model.

Rusmevichientong and Topaloglu (2012) consider the assortment problem and propose an adaptive policy to learn the unknown parameters of the MNL model and optimise the assortment problem at the same time. The adaptive policy is divided into exploration steps and exploitation steps. In exploration steps, they provide different assortments to the customers' observed selection probabilities and find the order of the products based on the selection probabilities. Then they propose an algorithm to find a sequence of assortments with given ordering. After that, exploitation steps that employ a sampling-based golden ratio search are performed to compare different assortments. The optimal assortment can be found after enough iterations.

Instead of the parametric approaches used in the above papers, Farias et al. (2013) provide a non-parametric approach, which has no prior assumption on the structure of the choice model. Close to Farias et al. (2013), van Ryzin and Vulcano (2011) consider a general, non-parametric discrete choice model with a Bernoulli process of arrivals over time. An EM algorithm solves the estimation. With the numerical results, they show that the method has one order of magnitude improvement compared to the maximisation of incomplete likelihood function.

Another stream of estimation methods is Bayesian methods. Since the coefficients in the MNL model have no closed-form posterior distribution, a Markov Chain Monte Carlo (MCMC) method is usually applied to solve the problem. The Markov chain is a discrete time stochastic process with a transition operator with memoryless property that means the future decision depends only on the current state instead of any other historic state. This memoryless property is called the Markov property. The Monte Carlo methods comprise a form of algorithm that simulates an independent and identically distributed (i.i.d.) set of samples from the required distribution and estimates the target distribution.

A detailed description of MCMC methods can be found in Robert and Casella (2013). The first MCMC method is provided in Metropolis et al. (1953) and is named the Metropolis algorithm. This algorithm is generalised by Hasting (1970) with a simulation method to avoid the curse of dimensionality in the Metropolis algorithm. Another widely used MCMC method is Gibbs sampling, which is described by Geman and Geman (1984). MCMC methods have been used to estimate the MNL model. Dellaportas and Smith (1993) use the adaptive rejection Gibbs sampling method to compute Bayesian inferences for generalised linear models and this method can be applied to estimate the coefficient of the MNL models. Gamerman (1997) address the estimation of generalised linear mixed models with Metropolis-Hastings (MH) method. Holmes and Held (2006) apply auxiliary variable approaches and Metropolis-Hastings methods for the inference of MNL model. Another paper by Scott (2011) also uses data-augmented Metropolis-Hasting sampling to estimate the MNL model. Different from the above two papers, Frühwirth-Schnatter and Frühwirth (2007) propose a data augmentation and Gibbs sampling method for the MNL model.

Letham et al. (2015) apply the stochastic gradient MCMC algorithm to estimate arrival rate and substitution behaviour with stockout. In the paper, sales data with incompleteness are applied. With their method, the parameter of base utility is unidentifiable. With the idea of the two-step algorithm in Newman et al. (2014), we can apply a MCMC algorithm to estimate the arrival rate and parameters in the MNL model with unique solution. We describe how we have incorporated the MCMC into Newman et al.'s (2014) algorithm in Section 3.3.

2.2 Revenue Management

In Section 2.2.1, we introduce the studies that focus on dynamic pricing problems in general. Section 2.2.2 reviews the papers that solve choice-based revenue management problems. Both substitutable products problems and network revenue management problems are included in this section. In Section 2.2.3, we discuss the papers that solve the dynamic pricing problems with demand uncertainty.

2.2.1 Dynamic Pricing

In general, RM deals with three main classes of decisions: structural-based decisions, price-based decisions and quantity-based decisions. This research focuses on price-based decisions which incorporate customer choice behaviour. Two overviews of dynamic pricing can be obtained from Elmaghraby and Keskinocak (2003) and Bitran and Caldentey (2003). The book written by Talluri and van Ryzin (2006) also provides a detailed description of dynamic pricing.

The 1962 paper by Kincaid and Darling is one of the first to dynamically price a single perishable product in a continuous time model with a homogeneous Poisson demand pattern, with the focus being on retail.

Gallego and van Ryzin (1994) introduce the ideas to RM and present a continuous-time model that considers stochastic demand for a single product and gives an upper bound on the revenue. For the problems with exponential demand function, they provide an exact solution. Since continuously changing the price is impractical, problems with finite price vectors are addressed in the paper as well. Several extensions are also included in this paper. Zhao and Zheng (2000) extend the model of Gallego and van Ryzin (1994) to incorporate non-homogeneous demand and time-dependent reservation price. The model in Feng and Gallego (2000) also has a non-homogeneous Poisson demand pattern. An efficient algorithm is developed in the paper to find the optimal decisions. The models in Feng and Gallego (1995), Bitran and Mondschein (1997) and Feng and Xiao (2000) have the constraint that the number of prices changes is finite.

The above papers solve problems with single product to sell. Gallego and van Ryzin (1997) extend their 1994 model by considering multiple products in the network setting. Two asymptotically optimal heuristics are provided to solve the problem. The paper also presents an upper bound on the original problem based on the deterministic model.

Maglaras and Meissner (2006) consider a dynamic pricing problem with multiple products, which share a single resource. The paper shows that the dynamic pricing problem and capacity control problem can be reduced to a common formulation. Several heuristics are also provided based on its deterministic model.

Sen (2013) provides a comparison of fixed and dynamic pricing policies for the problem of selling a fixed capacity or inventory of items over a finite selling period. The author proposes an approximation method which is similar to the method in Zhang and Cooper (2009) to solve a dynamic pricing problem with a single product. He approximate the optimal expected revenue with a combination of the lower bounds and upper bounds for the value function. The lower bounds are approximated by dividing the whole selling period by the number of inventory levels. Only one unit of product is sold in each small period and any unsold product is ignored in the next small period. The upper bound comes from a result in Gallego and van Ryzin (1994). We use the ideas of Sen (2013) in the work presented here by treating the bounds in Zhang and Cooper (2009) as a value function for single product problem. More details can be found in Section 4.3.

Elmaghraby and Keskinkocak (2003) classify the studies of dynamic pricing problems with three characteristics – replenishment scheme, demand dependency over time and customer purchase behaviour – which we discuss below

Replenishment vs. No Replenishment of inventory

Whether the inventory is allowed to replenish or not is an important policy for dynamic pricing problems. If products can be replenished, the seller needs to make both pricing decisions and inventory decisions. In our problem, we cannot replenish our inventory periodically; however, we can make the initial inventory decision or have a fixed amount of inventory at the beginning of the selling period and then make pricing decision with initial inventory level. Gallego and van Ryzin (1994), Bitran and Mondschein (1997), Zhao and Zheng (2000) and Maglaras and Meissner (2006) consider problems without inventory replenishment. A dynamic pricing problem with replenishment decisions can be found in Federguen and Heching (1999). In the problem we consider here, we assume no replenishment of inventory.

Dependent vs. Independent Demand over Time

For a product which has a duration that is longer than the selling horizon, the current sale will reduce sales in the future, so the demand is dependent over time. For non-durable products, like food, clothes, tickets for airlines, and trains, among others, demand is independent over time. We focus our study on time-independent demand models. Problems with time-dependent demand can be found in Zhao and Zheng (2000).

Myopic vs. Strategic Customers

If customers are assumed to purchase when the offer price is below their willingness to pay, we call them myopic customers. If customers optimise their purchase by considering the company's pricing scheme, we call them strategic customers. We consider myopic customers in this study. A detailed review of models that incorporate strategic customers can be found in Shen and Su (2007).

Another characteristic which can be used to classify the dynamic pricing problem is the level of competition.

Monopoly vs Competitive Pricing

Many models assume that the sellers enjoy a monopoly so that the demand for products depends only on their own price; however, this is not always realistic. With the development of the Internet, customers can compare the prices of similar products from different sellers within several minutes, so the prices of similar product from other sellers are incorporated in many studies. Currie et al. (2008) provide the optimal price structure under competition and the assumptions needed to guarantee the uniqueness of optimal price. Both unresponsive and responsive competitors are considered in the paper. Gallego and Hu (2007) present a continuous-time stochastic game with multiple players to solve the choice-based RM problem under competition.

In this paper, we focus on choice-based dynamic pricing models with no inventory replenishment, independent demand over time, and myopic customers. We assume that customers choose between the products on offer from just one company and ignore competition with external companies. This stream of studies can be found in Gallego and van Ryzin (1994), Bitran and Mondschein (1997) and Zhao and Zheng (2000).

2.2.2 Choice-based Revenue Management

There is a common assumption in RM that the demand for products is an independent stochastic process. The probability that the customer chooses a product will not be affected by the availability or prices of other products. However, in many cases, the customer will choose a product from an offer set of products. The company needs to find a proper choice model to incorporate customer choice behaviour. The revenue improvement by considering customer choice behaviour in RM problems was shown by Vulcano et al. (2010) to be, on average, between 1% and 5%.

With the estimation of the choice model, we can incorporate customer choice behaviour in RM problems. Empirical studies of choice-based RM problems can be found in Ratliff et al. (2008) and Vulcano et al. (2010).

Zhang and Cooper (2005) solve a quantity control choice-based RM problem with substitutable products. In their paper, they formulate the problem as a Markov decision process (MDP) and provide the upper and lower bounds of the value function. Due to the complexity of the MDP, an approximation of the value function and a heuristic, which is based on linear programming, are proposed to solve the problem. Based on the comments in Talluri and van Ryzin (2006), Zhang and Cooper (2009) argue that setting an optimal pricing policy may achieve a better performance than quantity control would. Therefore, Zhang and Cooper (2009) consider a pricing control problem for substitutable products and a general choice model is used in the paper. In the paper, they assume that there is at most one arrival in each period and the order of arrivals is not determined. In addition, they allow the products to have different prices even in the same period. Because of the intractability of the MDP, they propose an approximate dynamic programming to solve the problem. We consider the method of Zhang and Cooper in more detail in Chapter 4, where we provide a further approximation to their algorithm to improve the computational efficiency.

Unlike the general choice model used in these two papers referred to above, Dong et al. (2009) use the MNL model to incorporate customer choice behaviour. Dong et al. (2009) jointly consider the starting inventories and dynamic pricing decision for the problem with horizontally differentiated products, which are not uniformly ordered according to customer preference. They provide the optimal dynamic pricing of substitute products and a heuristic to decide proper starting inventory levels. The value of dynamic pricing is also demonstrated by a numerical study in the paper. Akcay et al. (2010) study both horizontally differentiated products and vertically differentiated products, which both have a clear order. For horizontally differentiated products, they prove that the MNL profit function is a uni-modal function in price. They provide a number of analytical results on the choice-based dynamic pricing problem.

Suh and Aydin (2011) also use the MNL model to model the customer choice behaviour. In the paper, they study a two-product dynamic pricing problem in finite time. They show that the marginal value of a product is increasing in time periods and decreasing in inventory levels. However, the optimal price of a product does not have a monotonic property in time period and inventory level.

Li and Huh (2011) and Gallego and Wang (2014) consider a different choice model. They study a multiple products pricing problem with a Nested Logit model. The customer behaviour is modelled with a two-stage model. The customers choose a nest at the first stage and then choose a product within the nest at the second stage. Li and Huh (2011) impose an additional restriction on the nest coefficients and the price sensitivities are the same for all the products. Gallego and Wang (2014) consider a more general NL model and get rid of these restrictions.

Different from the papers presented above, Yang et al. (2014) combine a choice-based dynamic pricing problem with a vehicle routing model. They propose a policy that dynamically adjusts the delivery costs to control the customers' choices, which affect the routing schedules. In the paper, they also provide the estimation of a MNL model that fits the customer choice behaviour.

The majority of the research related to choice-based RM focuses on network revenue management, which has many applications in the airline industries. However, we do not consider network RM in our study. While acknowledging that train travel takes place in a complex network, we consider the simple problem of single-leg price optimisation where the trains with the same origin-destination pair can be treated as substitutable products instead of the products in network RM. However, for completeness, we review choice-based RM for networks below for completeness.

Gallego et al. (2004) is the first paper which studies choice-based RM within a network setting. A deterministic linear programming model is proposed to solve the problem. Liu and van Ryzin (2008) show that the model in Gallego et al. (2004) is asymptotically optimal. They also extend the efficient sets proposed by Talluri and van Ryzin (2004) to a network setting. A heuristic, which decomposes the DP, is provided in the paper. This method is proved to be an efficient method when the problem has a MNL choice model and disjointed segments. Van Ryzin and Vulcano (2008), Bront et al. (2009), Zhang and Adelman (2009) and Kunnumkal and Topaloglu (2010) extend the study of the linear programming model with different methods.

Bront et al. (2009) extend the model in Liu and van Ryzin (2008) to incorporate overlapping customer segments. A column generation algorithm is provided to address the linear programming model, and the sub-problem of column generation is solved by a greedy heuristic. The finite mixture MNL model is used as the customer choice model in this paper.

Van Ryzin and Vulcano (2008) present a stochastic gradient algorithm for the network RM problem with virtual nesting control policy. The choice modelling and optimisation are separated in the method; therefore this method can be applied to general customer choice models.

Zhang and Adelman (2009) solve the linear programming model first studied by Gallego et al. (2004) through the use of an approximate dynamic programming method and prove that the bound from this method is tighter than the one from the choice-based linear programming method. The customer choice model they used is a MNL model. A column generation algorithm is generated to solve the problem.

Different from the papers above that consider quantity control problems, Zhang and Lu (2013) consider a dynamic pricing problem in network revenue management. They decompose the network problem into a collection of single resource problems. The performance of the method is tested with the comparison of static pricing and the choice-based quantity control model. They show that a firm can improve its profit by applying the dynamic pricing policies. The upper bound in revenue that is tighter than that in the deterministic models is also presented in the paper. Different from the work of Zhang and Lu (2013), Du et al. (2016) incorporate a customer choice behaviour that is modelled with the MNL model. They study a pricing problem with network effect and show that the pricing policies depend on the network effect. For the problem with weak network effect, the optimal pricing policy sets the same prices for all products; otherwise, the prices of products differ. The problems with both heterogeneous and homogeneous products are considered in the paper.

2.2.3 Dynamic Pricing with Demand Uncertainty

Most of the literature in the dynamic pricing stream assumes that the parameters of the demand function for the products are known in advance. However, in many situations demand information is not available; in particular, if the products are new to the market. In recent years, dynamic pricing with demand uncertainty has become an area that has attracted considerable attention. In this section, we give an overview of the papers that focus on the dynamic pricing problem with demand uncertainty. A detailed review of dynamic pricing and learning can be found in den Boer (2015). The paper also introduces the historical origins of the problem and the directions for future research in different scientific areas. We split our discussion into three sections – Bayesian methods, non-Bayesian methods, estimation for multiple products – with the final section being of most relevance to this project.

Bayesian Method

Similar to the problem we considered, Aviv and Pazgal (2002) study a problem that has a finite amount of perishable products to be sold in a finite time. Different from our choice model, they assume the customers make their choice based on their own reservation price. The seller has the information about the distribution of the reservation price, which is exponentially decreasing in the selling price. In the paper, the authors assume the demand is the number of customers who arrive and purchase at least one available product. With this assumption, they ignore the no-purchase arrival and stockout effect. In the problem, the authors assume that the arrival rate is fixed but unknown to the seller. The seller has a prior belief about the arrival rate and updates the belief with a Bayesian method as the sales data are collected. They combine this learning mechanism with a continuous-time dynamic pricing problem and compare the results with three other pricing policies: a fixed price policy, a certainty equivalent pricing heuristic and naive pricing policy that ignores uncertainty over demand.

Aviv and Pazgal (2005) propose a partially observed Markov decision process (POMDP) framework to solve a perishable dynamic pricing problem. They assume that the seller sells a finite stock of products in a finite time with the aim to maximise expected revenues. Similar to our work, they consider the uncertainty about the arrival rate, actual number of arrivals, times of arrival and individual purchase decisions. They also have no information about customers' reservation prices and the state of the market. The belief of these uncertainty values is updated with a Bayesian method. The authors also provide a rigorous upper bound approximation for the POMDPs.

Different from the setting of perishable products in the above two papers, Araman and Caldentey (2009) study a single non-perishable product problem with finite inventory. A Bayesian method is applied to learn the parameter in a non-homogeneous Poisson arrival process. They show that the optimal price is not a decreasing function of inventory level for the problem with non-perishable product. The factor affecting the optimal price is the market size and the optimal price is a monotonically increasing function of market size with a given inventory level.

The above three studies focus on problems with finite inventory. Harrison et al. (2012) consider a problem with infinite inventory. Their study aims to solve the trade-off between learning and earning. The authors consider the problem with a Bayesian method and they show that the myopic Bayesian pricing policy may lead to incomplete learning. To prove the theory and simulation performance, they propose a constrained myopic Bayesian policy which avoids incomplete learning and they prove that it has a bounded regret.

Non-Bayesian Method

Broder and Rusmevichientong (2012) study a single product problem of offering different prices to sequential customers. The customers make their own decision on whether to purchase this product or not based on a willingness-to-pay model. In the paper, they propose a price policy, which uses maximum-likelihood estimation to explore the demand model and they use a greedy price policy in the exploitation phase. The exploration phase and exploitation phase perform in a cycle. The length of the exploration phase is fixed and the length of the exploitation phase is expanded as more cycles are performed. The regret of the price policy has the order of $\sqrt{T}$. The authors also consider a special problem that precludes uninformative price, which is that the price cannot provide any information about demand model for any price policy. For the problem without uninformative price, the paper provides a greedy pricing policy. The regret of the policy has the order of $\log T$. In Chapter 5, we show the comparison between our policies and this policy in numerical results.

Tehrani et al. (2012) consider a similar sequential pricing problem to Broder and Rusmevichientong (2012) with a finite set of possible demand models. The information they needed to solve the problem is the prices under each possible demand model and the values of demand at these prices. The authors regard the optimal price as the arms in a Multi-armed Bandit problem. Since the observation under each arm is obtained from the true demand model, the arms are correlated, and the original problem is solved as a Multi-armed Bandit with dependent arms. The pulling policy in Multi-armed Bandit they proposed is based on a likelihood ratio test and the authors prove that the policy has a bounded regret. The difference between this paper and our study is that the authors assume their problem has a finite set of possible demand models. In our problem, we have no information of the parameters in the choice model.

Different from the above two papers that use the data that reflect true demand information, the data in Besbes and Muharremoglu (2013) are the sales data that differ from demand data when the demand for products exceeds the inventory level. They study a repeated newsvendor problem with demand uncertainty. It is similar to our problem, which aims to estimate primary demand of products. In the paper, they compare the cumulative costs of a policy without demand uncertainty and the cumulative costs with known demand distribution and define the difference as the regret of the policy. They also provide the upper and lower bounds to find the magnitude of the worst case of regret. Both continuous demand and discrete demand are considered in the paper. The authors show that active exploration is more important for problems with discrete demand distribution. However, if the seller has information that any sales were lost, the impact of demand censoring can be reduced in the problems with discrete demand distribution.

Besbes and Zeevi (2015) study a single-product pricing problem in a finite time period. The monopoly can change the prices of the product in each period with a stationary demand environment, which is unknown to the monopoly. Unlike the assumption in other works that assume the structure of the demand is known, the authors adopt a linear model for the demand distribution. Even if the true demand model does not have a linear structure, they show that the pricing policy converges to the optimal price, which is obtained with true demand distribution under fairly general assumption. The expected revenue is also asymptotically close to the optimal revenues. In addition, the authors show that estimation-optimisation cycles can improve that the expected revenue even with a demand model, which is different from the true demand distribution.

Unlike the papers above, which assume the market conditions are unchanged, Chen and Farias (2013) consider a single-product pricing problem with time-varying market condition and imperfect forecasts. In the problem, finite inventory is sold over a finite time period. They provide a sub-optimal heuristic for the problem. They show that the expected revenue can be improved by re-optimising the fixed pricing policy.

To avoid the risks of mis-specification, over-fitting or under-fitting in the estimation of parametric models, Farias et al. (2013) provide a non-parametric approach to find the optimal assortment of products with a generic choice model, which is the preference list of products. They prove the approach is efficient and provides accurate sales predictions. The approach is applied for a major US automaker and shows that 10% improvement in revenue can be achieved.

Multiple Products

Boer (2014) considers a dynamic pricing problem with multiple products and infinite inventories. Different from our problem, they have partially knowledge about the demand distribution. The information about the first two moments of the demand distributions is available. Some parameters in the demand distributions are unknown and can be estimated with maximum quasi-likelihood estimation. The author provides an adaptive pricing policy, which finds optimal prices with current estimation of parameter and meets the requirement of price dispersion. The price dispersion is measured with the smallest eigenvalue of a design matrix, which grows with a pre-determined parameter. The shortcoming of the approach is that the exact solution of the optimisation problem is computationally intractable for the large problem.

Keskin and Zeevi (2014) study a dynamic pricing problem with infinite inventories in finite time periods. They assume the demand distribution is unknown but that it has a linear structure. In the paper, a policy called greedy iterated least squares is applied to combine the estimation and optimisation. The authors show that the policy leads to incomplete learning; however, the minimum asymptotic loss rate can be obtained by a modification of the policy. They also extend the policy to the case of multiple products with substitutable demand using an idea of orthogonal pricing which search for the price evenly in different directions of the price vectors. Compared to the orthogonal pricing policy, our pricing policy proposed in Chapter 5 is more straightforward and easier to implement.

2.3 Multi-armed Bandit Problem

The Multi-armed Bandit problem (MAB) is a sequential decision-making problem to solve the trade-off between exploration and exploitation. The term "bandit" is a slot machine, which has one arm, which can be pulled. The MAB problem assumes that there is an array of one-arm slot machines. At each time, the decision maker chooses one of the arms and pulls it down. A reward can be observed from the pulled arm and the reward is generated from some unknown distribution. The decision maker needs to learn which arm returns better rewards by repeatedly selecting different arms and observing the rewards. The aim of the player is to maximise the expected sum of rewards over a sequence of pulls. The decision maker may choose to do exploitation by pulling the arm that returns the highest reward based on current belief of the distributions of arms. Otherwise, he may choose to explore the arms by pulling the other arms to achieve a better understanding of the distribution of the chosen arm. If the decision maker spends too much time in exploration, he may lose the opportunity to pull the optimal arm enough times and improve the short-term profit. If the decision maker spends too much time in exploitation, he may lose the opportunity to find the actual best arm and improve the long-term profit. The pulling policies in MAB will solve the trade-off between exploration and exploitation.

Early application of MAB problems can be found in clinical trials to minimise patient loss. With the development of the Internet, MAB problems have had an increasing impact on the areas of web search and Internet advertising (Scott, 2010). It also has many applications in queueing and scheduling and fast fashion.

The application of MAB in the RM area can be found in Rothschild (1974; this was the first paper to formulate the dynamic pricing problem as a MAB problem. In the paper, there are only two prices to choose from. The author treats the prices as the arms in the bandit problem and finds the better price with the experiment in the MAB problem. Instead of treating the prices as the arms in MAB, Tehrani et al. (2012) treat the demand functions as the arms in MAB. They assume that the demand model is chosen from N possible demand functions, which are known in advance, and formulate the dynamic pricing problem as MAB with dependent arms. They prove that the policy in MAB achieves complete learning. Our setting is more similar to that of Broder and Rusmevichientong (2012) which uses MAB to balance the exploration of unknown parameters and exploitation of short-term profit.

In this section, we review the existing pulling strategies for the MAB problem. A review of pulling strategies for MAB problems can be found in Vermorel and Mohri (2005). The strategies of MAB fall into four broad categories, as described below.

2.3.1 Semi-uniform Strategies

Semi-uniform strategies consist of exploration steps and exploitation steps. In the exploitation step, the agent chooses the greedy action by playing the arm with the current highest expected reward. In the exploration step, the agent chooses a random action by playing an arm which is chosen with uniform probability.

The simplest and most popular semi-uniform strategy is the **epsilon-greedy strategy** which was first introduced by Watkins (1989). With this strategy, the agent chooses the greedy action with probability $(1-\epsilon)$ and chooses the random action with probability ϵ, where ϵ is a variable which is decided by the player in the interval (0,1). The numerical results in Vermorel and Mohri (2005) show that this strategy is hard to beat, but it is a sub-optimal strategy. The parameter ϵ ensures that the arms with lower expected reward are chosen with probability $(1-\epsilon) * (n-1)/n$, where n is the number of arms in the experiment.

A variant of the epsilon-greedy strategy is the **epsilon-decreasing strategy**, which was first studied in Cesa-Bianchi and Fischer (1998). It replaces ϵ by a decreasing factor $\epsilon_t = min\{1, \epsilon_0/t\}$, where t is the index of time, and ϵ_0 is a positive number which is chosen by the player.

Epsilon-first strategy is another variant of the epsilon-greedy strategy. The epsilon-first strategy chooses to explore before the exploitation steps. For a problem with T steps, the player chooses arms randomly during the ϵT steps and chooses the arm with highest expected reward during the remaining $(1-\epsilon)T$ steps. The ϵ-first policy cannot achieve the guarantee of asymptotic convergence; however, it can outperform many more complicated pulling strategies. We use this strategy in Chapter 5.

2.3.2 Probability Matching Strategies

Probability matching strategies choose an arm based on the probability of it being optimal. The **SoftMax strategy** is the most popular probability matching strategy which was first developed by Luce (1959). The player chooses arm k with probability

$$P_k = \frac{e^{\mu_k/\tau}}{\sum_{i=1}^{n} e^{\mu_i/\tau}},$$

where μ is the mean of the reward and τ is called the temperature parameter, which can be tuned by the player to improve the performance of the strategy.

A variant of the SoftMax strategy is **decreasing SoftMax**, which was developed by Cesa-Bianchi and Fisher (1998). The temperature parameter used in the strategy is $\tau_t = \tau_0/t$, where t is the time.

Exp3 or "exponential weight algorithm for exploration and exploitation" is another variant of the SoftMax strategy. The strategy is introduced in Auer et al. (1995). The main idea behind this strategy is dividing the actual reward by the probabilities that the actions are chosen. This strategy keeps a list of weights w for each of the arms and uses the weights to find the arm which is pulled in the next time period. The list of weights is updated after a reward is observed. γ is a parameter which is given in advance. If $\gamma = 1$, the list of weights has no effect on the choice of arm. For a MAB problem with K arms, the probability of choosing arm k at time t is

$$P_k(t) = (1-\gamma)\frac{w_k(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K}, (0 \leqslant \gamma \leqslant 1)$$

$$w_j(t+1) = \begin{cases} w_j(t)\exp(\gamma \frac{r_j(t)}{P_j(t)K}) & \text{if arm } j \text{ is pulled at } t \\ w_j(t) & \text{otherwise} \end{cases}$$

where $r_j(t)$ is the actual reward of arm j at time t.

Scott (2010) describes how the MAB is currently used for Google adverts. In the paper, a heuristic called **randomised probability matching** is introduced. The heuristic chooses optimal arms by using Bayesian posterior probability. The posterior can be obtained from the Markov Chain Monte Carlo method.

Thompson Sampling was first proposed by Thompson (1933), which is a Bayesian algorithm and chooses the arm with the probability of it being the best arm. The paper has been ignored until recently. Agrawal and Goyal (2012) prove that Thompson Sampling achieves logarithmic expected regret for the stochastic MAB problem.

In Chapter 5, we adopt the idea of matching a probability with an arm and propose two policies in MAB.

2.3.3 Interval Estimation

The interval estimation strategy was first developed by Kaelbling (1993). It associates each arm with a certain confidence interval. The strategy will choose the arm with the highest upper bound on the confidence interval. Infrequently chosen arms have a wider interval and over-valued upper bound. Therefore, the probability of choosing this arm is higher. The more times an arm is chosen, the tighter its corresponding confidence interval will be.

The upper confidence bound method (UCB) was introduced by Auer et al. (2002), which is a class of algorithm which optimise the reward with uncertainty. In the paper, they prove that the UCB methods have finite-time regret bounds and have optimal asymptotic convergence. UCB1 is a algorithm in the UCB method. In the algorithm, the player plays each arm once, then in the following steps, plays arm j, which maximises

$$\bar{x}_j + \sqrt{\frac{2\ln n}{n_j}},$$

where $\bar{x}_j$ is the empirical mean reward of arm j, n_j is the number of times arm j has been pulled so far, and j is the total number of arms that have been pulled. Auer et al. (2002) also propose the algorithms UCB-tuned and UCB2. Schwartz et al. (2016) apply an Upper Confidence Bound policy to balance the learning and earning phase and solve a similar problem to the problem we study.

2.3.4 Gittins Index Strategies

The index Theorem was first published by Gittins and Jones (1974). The Theorem associates a real scalar with each arm and the scalar is the measurement of the reward function. The bandit with the highest index is chosen to be pulled in the next iteration. The Gittins index is an optimal solution for the problem, which allows only one arm to be pulled in each step. For the problem that allows multiple arms to be pulled, the

Gittins index is a sub-optimal solution. A detailed review of the Gittins index strategies can be found in Gittins et al. (2011). Leloup and Deveaux (2001) use the Gittins index strategies to solve the Bayesian dynamic program.

2.4 Conclusion

The literature review has focused on previous work in estimation and optimal pricing for RM problems involving customer choice, but we have also discussed research into Multi-armed Bandit algorithms in RM.

We pointed out that there are several research gaps in literature. First, given the information of market share, the estimation of a MNL model with stockout effect has not been solved with a maximum likelihood estimation or a MCMC method. Second, the choice-based dynamic pricing problem has been solved with an approximation method from Zhang and Cooper (2009). However, the computation time of the method prevents the user from updating the prices in a short period of time with large instances. Third, a policy in the MAB which has a good performance and combines an existing estimation method and optimisation method is needed.

The problem of estimating the parameters of choice models has received more interest in recent years since choice-based RM has been more widely researched. We combine ideas from choice-based RM and MAB algorithms drawing on work in key papers from Zhang and Cooper (2009), Vulcano et al. (2012) and Newman et al. (2014) in choice-based RM and incorporating ideas of learning and earning from Broder and Rusmevichientong (2012) to balance the trade-off between learning and earning.

Chapter 3

Parameter Estimation in Choice-based Revenue Management

The multinomial logit (MNL) model can be estimated using a maximum likelihood estimation method if complete data are provided. However, when we consider the situation that some products may be sold out before the end of the selling period, we cannot determine whether a customer would have chosen this product if all the products were available. In addition, only the arrivals of customers who buy products are recorded in the sales data. The data of arrivals without a purchase are not observable. Hence, the sales data we have are incomplete. Therefore, the maximum likelihood estimation method cannot be used directly. In this situation, the expectation maximisation (EM) algorithm can be applied to estimate the MNL model.

For the problem that has the information of market share, we show that the MNL model can be estimated by optimising a globally concave function which avoids the iteration in the EM algorithm. In certain simulations, we may have prior information about the MNL parameters, e.g., from survey data, which we wish to incorporate into our estimation process. Hence, Bayesian methods can be used. We also investigate the use of the Markov Chain Monte Carlo (MCMC) method and find the posterior distribution of the parameters in the MNL model.

In this chapter, we mainly consider a problem in which a seller offers a set of substitutable products to customers in a finite time. The arrival pattern of the customers follows a non-homogeneous Poisson process. The customer choice behaviour is homogeneous and can be modelled with a multinomial logit (MNL) model. The data available to the seller is a record of transactions, an indicator of the product availability in each time period and the aggregate market share. We jointly estimate the arrival rate and the parameters in the MNL model.

In Section 3.1, we introduce the use of EM algorithms to estimate the MNL parameters with or without stockout effect. Section 3.2 describes the two-step algorithm which solves the problem with stockout effect. In Section 3.3, the MCMC method is applied to do the estimation with stockout effect. The numerical results of algorithms are presented in Section 3.4.

3.1 Estimation using the Expectation-Maximisation Algorithm

3.1.1 Estimation without stockout effect (Vulcano et al., 2010)

In this section, we apply the EM algorithm provided by Vulcano et al. (2010) to solve the estimation problem when all products are always available, i.e. ignoring the effect of stockout. The seller has access to the sales data and has no information about the aggregate market share. In this problem, only one type of missing data exists. Only the arrivals of customers who buy products are recorded in the sales data. The data of arrivals without a purchase are not observable. If we ignore this type of missing data, we can cause a severe bias in estimation. The arrival rate of customers will be underestimated and the seller will lose potential customers. This problem has been solved in Vulcano et al. (2010) with an EM algorithm. We show that the estimation quality is highly dependent on the initial point, and that further information is needed to achieve a good estimation.

The EM algorithm is an iterative algorithm that can be used to estimate parameters for the problem with incomplete data. It has been widely used in the RM area (Talluri and van Ryzin, 2004; Vulcano et al., 2010, 2012). The EM algorithm consists of two steps: the Expectation (E) step and the Maximisation (M) step. The Expectation step calculates the expected value of the log-likelihood function of the missing data with the current estimate for the parameters. The Maximisation step calculates the maximum value of the expected log-likelihood function. The parameters maximising the function are recorded and used in the next expectation step. These two steps are performed iteratively until the estimation stops improving or a stopping criterion is reached. The procedure is given in Algorithm 1 and described in more detail below.

We consider a problem that a seller has a set of products which is denoted by S sold over T time periods. The length of each time period is assumed to be small enough such that the probability of more than one arrival in a period is negligible. The arrival pattern is modelled with the Poisson arrival process. In each time period, the arrival rate is denoted with λ. The only data available to the seller are the sales data, which record actual purchases. These have the information about whether a particular time period has a purchase or not and, if a purchase is observed in a time period, which product is sold. The seller cannot observe the customers who arrive into the selling system without a purchase.

Specifically, let β denotes the unknown coefficients for the attributes in the MNL model. The attributes can be cost, travel time or travel day, which are the characteristics that affect the choices of customers. Let y_i denote the vector of attributes for product i from product set S. Define $a_t = 1$ if there is an arrival in period t, and $a_t = 0$, if there is no arrival. Let $j(t)$ denote the choice made by an arrival in period t. The complete log-likelihood function can be written as

$$\begin{aligned}\mathcal{L} &= \sum_{t\in P}\left[\log(\lambda) + \beta^{\mathsf{T}} y_{j(t)} - \log(\sum_{i\in S} e^{\beta^{\mathsf{T}} y_i} + 1)\right] \\ &\quad + \sum_{t\in \overline{P}}\left[a_t\left(\log(\lambda) - \log(\sum_{i\in S} e^{\beta^{\mathsf{T}} y_i} + 1)\right) + (1-a_t)\log(1-\lambda))\right], \qquad (3.1)\end{aligned}$$

where P denotes the set of time periods with purchase and $\bar{P}$ denotes the set of time periods without purchase. The variable a_t is the unobservable information which presents the application of this complete log-likelihood function. The expected value of a_t can be obtained by Bayes' method as shown below. Let $\hat{a}_t$ denote the expected value of a_t. We can write $\hat{a}_t$ as

$$\begin{aligned}\hat{a}_t &= E[a_t|t\in\overline{P},\beta,\lambda] \\ &= P(a_t=1|t\in\overline{P},\beta,\lambda) \\ &= \frac{P(t\in\overline{P}|a_t=1,\beta,\lambda)P(a_t=1|\beta,\lambda)}{P(t\in\overline{P}|\beta,\lambda)} \\ &= \frac{\lambda P_0(S|\beta)}{\lambda P_0(S|\beta)+(1-\lambda)}. \qquad (3.2)\end{aligned}$$

The number of arrivals is the number of time periods with purchase ($|P|$) plus the number of time periods with arrivals but no purchase ($\sum_{t\in\overline{P}}\hat{a}_t$). Therefore, the arrival rate can be calculated by the ratio of the number of arrivals to the number of time periods as

$$\lambda = \frac{|P| + \sum_{t\in\overline{P}} \hat{a}_t}{|P| + |\overline{P}|}. \tag{3.3}$$

In the expectation step of each iteration, we calculate the expected value of a_t from Equation 3.2 with the estimates of β and λ from the last maximisation step. Then we calculate the value of λ from Equation 3.3.

In the maximisation step, we determine the value of β by maximising the expected log-likelihood function

$$\sum_{t\in P}\left(\beta^{\mathsf{T}} y_{j(t)} - \ln(\sum_{j\in S} e^{\beta^{\mathsf{T}} y_j} + 1)\right) - \sum_{t\in\overline{P}} \hat{a_t}^{(k)} \ln(\sum_{j\in S} e^{\beta^{\mathsf{T}} y_j} + 1),$$

where $j(t)$ denotes the product which is chosen at time period t.

In each iteration, we test the convergence by calculating the norm of the difference between two consecutive estimates. If $\left\|(\lambda^{(k+1)}, \beta^{(k+1)}) - (\lambda^{(k)}, \beta^{(k)})\right\| < \delta$, we stop and use $\lambda^{(k+1)}$ and $\beta^{(k+1)}$ as the final estimation. If the stop criterion is not met, we go to the next expectation step and perform a new iteration. The convergence of this EM algorithm is not guaranteed. However, the EM algorithm has been proved to be a robust method to solve incomplete data problems in practice (Vulcano et al., 2010).

Algorithm 1 EM algorithm for the problem without stockout effect (Vulcano et al., 2010)

Initialise: Set $\beta^{(0)} = 0$, $\lambda^{(0)} = 0$ and $k = 0$.

while $\left\|(\lambda^{(k+1)}, \beta^{(k+1)}) - (\lambda^{(k)}, \beta^{(k)})\right\| \geqslant \delta$, **do**

E-step: For $t \in \bar{P}$, calculated $\hat{a}_t^{(k)}$ with equation

$$\hat{a}_t^{(k)} = \frac{\lambda^{(k)} P_0(S|\beta^{(k)})}{\lambda^{(k)} P_0(S|\beta^{(k)}) + (1 - \lambda^{(k)})}.$$

M-step: Calculate $\lambda^{(k+1)}$ with equation

$$\lambda^{(k+1)} = \frac{|P| + \sum_{t \in \overline{P}} \hat{a}_t^{(k+1)}}{|P| + |\overline{P}|}$$

and compute $\beta^{(k+1)}$ by maximising

$$\sum_{t \in P} \left(\beta^{\mathsf{T}} y_{j(t)} - \ln(\sum_{j \in S} e^{\beta^{\mathsf{T}} y_j} + 1) \right) - \sum_{t \in \overline{P}} \hat{a}_t^{(k)} \ln(\sum_{j \in S} e^{\beta^{\mathsf{T}} y_j} + 1).$$

In the results section, we show that the quality of the estimation results with this method will be affected by the initial point, and that multiple maxima exist.

3.1.2 Estimation with Limited Inventory

In this section, we estimate customer choice behaviour for the problems with limited inventory; that is, some products may be sold out before the end of the selling period. In this situation, two types of missing data are considered. The first type of missing data is the same as the type in the previous section. Only the arrivals of customers who booked the tickets have been recorded, while the data of arrivals without a booking are not observable. The second type of missing data are generated by the limited inventory level. When a product is sold out during the selling period, we cannot determine whether a customer would have chosen this product if the full product set was available. If we ignore the first type of missing data, the arrival rate of customers will be under-estimated and the seller will lose potential customers. If we ignore the second type of missing data, we will under-estimate the levels of preference for the products that are out of stock and over-estimate the preference on the products that are in stock. The estimation with bias will affect the seller's pricing decisions and decision on initial inventory. When we consider these two types of missing data, it will be difficult to develop a method that can estimate the parameters. Therefore, we assume we have access to additional information to improve the estimation.

In this section, we consider a problem whereby a seller sells n substitutable products over T time period, indexed $t = 1, 2, \cdots, T$. In each time period, the prices offered to customers and the availability of the products remain unchanged. The time period may have different lengths and we do not assume that at most one arrival in each period. This is a different assumption from the one in the unlimited case. The data available are the sales data that record the number of purchases in each period and the indicators of product's availability in each time period.

Vulcano et al. (2012) solve this missing data problem with an EM algorithm which jointly estimates arrival rate λ and the preference weights v in MNL. The preference weight v_j shows the "attractiveness" of product j and the no-purchase preference weight v_0 is normalised to have $v_0 = 1$. For the problem with changing of price during the selling period, preference weight of the products cannot remain unchanged. Therefore, the preference weights need to be broken down into a function of attributes, which includes the attribute of price. Let β_0 denote the coefficient for attribute of price and β denote a vector of the parameters for the other characteristics of the products. Meanwhile y_j denotes the characteristics of product j, except the attribute of price. The customer preference weight v_{jt} for product j at time t can be calculated by

$$v_{jt} = \exp(\beta_0 p_{jt} + \beta y_j).$$

For the problem that allows the prices of products to be changed, we jointly estimate the arrival rate and the parameters in the MNL model directly from an EM algorithm. We assume customers make their choice over the products with the MNL model. If their first choice is unavailable, they will choose not to buy or choose another product from the available product set. Given available product set S_t at time period t, the probability that a customer chooses product i, which is available, is calculated by

$$P_{it}(S_t) = \frac{v_{it}}{\sum_{j \in S_t} v_{jt} + 1}.$$

For product i which is unavailable at time period t, we have

$$P_{it}(S_t) = 0.$$

Let $P_0(S_t)$ denote the probability that a customer chooses not to buy or purchase from another company. It can be calculated by

$$P_0(S_t) = \frac{1}{\sum_{j \in S_t} v_{jt} + 1}.$$

To solve the problem of multiple maxima which exist in the unlimited estimation method, Vulcano et al. (2012) assume the price vector is fixed during the selling period and the market share is fixed and estimated exogenously.

In the problem that we consider, the price vector will change during the selling period. Therefore, we need to extend the method of Vulcano et al. (2012). We assume the market share of the products is known to the seller under a certain price vector p^* and that the other sellers will not respond to the price change of this seller. Therefore, we can calculate a new market share when a different price vector is offered to customers. Market share s is defined as

$$s = \frac{1}{1+r} = \frac{\sum_{i=1}^{n} \exp(\beta_0 p_i^* + \beta y_i)}{\sum_{i=1}^{n} \exp(\beta_0 p_i^* + \beta y_i) + 1},$$

where r shows the preference weight of purchasing products from other companies. When the price vectors offered to customers are changed, the new market share can be calculated by

$$s_t = \frac{1}{1+r_t} = \frac{\sum_{i \in S_t} \exp(\beta_0 p_{it} + \beta y_i)}{\sum_{i \in S_t} \exp(\beta_0 p_{it} + \beta y_i) + 1}.$$

We use X_{jt} to denote the number of purchases for product j at time t assuming all products are available. The log-likelihood function can be calculated by

$$\begin{aligned}
\mathcal{L} &= \sum_{j=1}^{n} \sum_{t=1}^{T} X_{jt} \left[log(\frac{v_{jt}}{\sum_{i=1}^{n} v_{it} + 1}) \right] \\
&\quad + \sum_{j=1}^{n} \sum_{t=1}^{T} (r_t X_{jt}) \left[log(\frac{1}{\sum_{i=1}^{n} v_{it} + 1}) \right] \\
&= \sum_{j=1}^{n} \sum_{t=1}^{T} X_{jt} \left[\beta_0 p_{jt} + \beta y_j - (r_t + 1) log(\sum_{i=1}^{n} v_{it} + 1) \right], \qquad (3.4)
\end{aligned}$$

where r_t is the preference weight of the outside alternatives which is calculated by

$$r_t = \frac{1}{\sum_{i=1}^{n} v_{it}}.$$

Let z_{jt} denote the actual number of purchases for product j at time t.
When product j is unavailable, we have

$$X_{jt} = \frac{v_{jt}}{\sum_{i=1}^{n} v_{it} + 1} A_t \tag{3.5}$$

and

$$\sum_{h \in S_t} z_{ht} = \frac{\sum_{h \in S_t} v_{ht}}{\sum_{h \in S_t} v_{ht} + 1} A_t. \tag{3.6}$$

where A_t is the number of arrivals in time t. Substituting Equation 3.6 into Equation 3.5, we have

$$X_{jt} = \frac{v_{jt}}{\sum_{i=1}^{v} v_{it} + 1} \frac{\sum_{h \in S_t} v_{ht} + 1}{\sum_{h \in S_t} v_{ht}} \sum_{h \in S_t} z_{ht}.$$

When product j is available for customers, we have

$$X_{jt} = \frac{v_{jt}}{\sum_{i=1}^{n} v_{it} + 1} A_t. \tag{3.7}$$

and

$$z_{jt} = \frac{v_{jt}}{\sum_{h \in S_t} v_{ht} + 1} A_t. \tag{3.8}$$

Substituting Equation 3.8 into Equation 3.7, we have

$$X_{jt} = \frac{\sum_{h \in S_t} v_{ht} v_{it}^{k} + 1}{\sum_{i=1}^{n} v_{it} + 1} z_{jt}$$

Let λ_t denote the number of arrivals in time period t. It can estimated with

$$\lambda_t = X_{0t} + \sum_{i=1}^{n} X_{it},$$

where X_{0t} is the primary demand for the no-purchase option.

The steps of the EM algorithm are given in Algorithm 2.

Algorithm 2 EM algorithm for the problem stockout effect (Vulcano et al., 2012)

Initialise:

Set $\beta_0^{(0)} = 0$, $\beta^{(0)} = 0$ and $k = 0$.

if $j \in S_t$ **then** set $X_{jt}^0 = z_j t$

else

set $X_{jt}^0 = 0$

end if

while $\left\|(\lambda^{(k+1)}, \beta^{(k+1)}) - (\lambda^{(k)}, \beta^{(k)})\right\| \geqslant \delta$, **do**

E-step:

for $t := 1, \ldots, T$ **do**

for $j := 1, \ldots, n$ **do**

if $j \notin S_t$ **then** calculate X_{jt}^k with equation

$$X_{jt}^k := \frac{v_{jt}^k}{\sum_{i=1}^n v_{it}^k + 1} \frac{\sum_{h \in S_t} v_{ht}^k + 1}{\sum_{h \in S_t} v_{ht}^k} \sum_{h \in S_t} z_{ht},$$

else

set

$$X_{jt}^k = \frac{\sum_{h \in S_t} v_{ht}^k + 1}{\sum_{i=1}^n v_{it}^k + 1} z_{jt}$$

end if

end for

Calculate r_t^k with equation:

$$r_t^k = \frac{1}{\sum_{jinS_t} v_{jt}^k}$$

end for

M-step:

Compute β_0^{k+1} and β^{k+1} by maximising

$$\sum_{j=1}^n \sum_{t=1}^T X_{jt}^k \left[\beta_0^k p_{jt} + \beta^k y_j - (r_t^k + 1) log(\sum_{i=1}^n v_{jt}^k + 1)\right]$$

The first correct convergence analysis of the EM algorithm can be found in Wu (1983) and Vulcano et al. (2012) prove the convergence of their EM algorithm. With similar analysis, we can prove the convergence of the EM algorithm that we show above.

Theorem 1. The log-likelihood function $\mathcal{L}$ is continuous in β_0 and β, and hence all the limit points of any instance $(\beta_0^k,\ \beta^k)$ of the EM algorithm are stationary points of the corresponding incomplete-data log-likelihood function $\mathcal{L}_1(\beta_0, \beta)$, and $\mathcal{L}_1(\beta_0^k, \beta^k)$ converges monotonically to a value $\mathcal{L}_1(\beta_0^*, \beta^*)$, for some stationary point (β_0^*, β^*).

Proof. From Theorem 1 in Vulcano et al. (2012), we have that the expected log-likelihood function $\mathcal{L}$ is continuous in $v > 0$. Since the function $v = \exp(\beta_0 p + \beta y)$ is, therefore, continuous in β_0 and β the expected log-likelihood function is continuous in β_0 and β.

The function $v = \exp(\beta_0 p + \beta y)$ is a linear system with a unique solution. Therefore, the function $\mathcal{L}_1(\beta_0, \beta)$ is unimodal with the Proposition 2 in Vulcano et al. (2012). Hence all the limit points of any instance $(\beta_0^k,\ \beta^k)$ of the EM algorithm are stationary points of the corresponding incomplete-data log-likelihood function $\mathcal{L}_1(\beta_0, \beta)$.

With Theorem 2 in Wu (1983), $\mathcal{L}_1(\beta_0^k, \beta^k)$ converges monotonically to a value $\mathcal{L}_1(\beta_0^*, \beta^*)$, for some stationary point (β_0^*, β^*).

3.2 Estimation using the Two-step Algorithm

In the previous section, the arrival rate and parameters in MNL model are estimated with the EM algorithm, which is an iterative method that alternates between finding the expected value of missing data and optimising the expected log-likelihood function. The process takes longer than the direct maximum likelihood estimation. In this section, we use the idea of the two-step method from Newman et al. (2014) to estimate customer choice behaviour with stockout effect. The method only needs to solve two globally concave functions which means the computation time is shorter than that of the EM algorithm.

The two-step method in Newman et al.(2014) split the estimation into two steps. In the first step, they estimate customer choice model parameters except the parameter for base utility. In the second step, they estimate the arrival rate and the parameter for base utility. In their paper, an estimation problem with unlimited inventory and no information about market share is provided. With the two-step method, the problem can transfer to solve a globally concave problem and a nonlinear optimisation problem and multiple maxima may exist in the second step.

In this section, we consider the same problem as the one in Vulcano et al. (2012). A slight difference is that we use the setting in Vulcano et al. (2010) and Newman et al. (2014) which has α as the base utility in the MNL model. The base utility is the utility of a base or reference option. The utilities of all other options are measured relative to the base utility. More details and an example can be found in the following results section.

Let $P_j(S_t, \alpha, \beta_0, \beta)$ denote the probability that a customer chooses product $j \in S_t$ given parameters α, β_0 and β in the MNL model. S_t is the product set offered at time period t. Then, the purchase probability can be calculated by

$$P_j(S_t, \alpha, \beta_0, \beta) = \frac{\exp(\alpha + \beta_0 p_{jt} + \beta y_j)}{\sum_{i \in S_t} \exp(\alpha + \beta_0 p_{it} + \beta y_i) + 1}. \tag{3.9}$$

For product j which is unavailable at time period t, we have

$$P_j(S_t, \alpha, \beta_0, \beta) = 0.$$

Let $P_0(S_t)$ denote the probability that a customer chooses not to buy or purchase from another company. It can be calculated by

$$P_0(S_t, \alpha, \beta_0, \beta) = \frac{1}{\sum_{i \in S_t} \exp(\alpha + \beta_0 p_{it} + \beta y_i) + 1}.$$

.

Vulcano et al. (2012) provide the incomplete likelihood function of this problem as

$$\begin{aligned} \mathcal{L}(\alpha, \beta_0, \beta, \lambda) \; = \; & \prod_{t=1}^{T} \Bigg\{ Prob(m_t \text{ customers buy in period } t | \alpha, \beta_0, \beta, \lambda) \frac{m_t!}{z_{1t}! z_{2t}! \cdots z_{nt}!} \\ & \prod_{j \in S_t} \left[\frac{1}{\sum_{i \in S_t} P_i(S_t, \alpha, \beta_0, \beta)} \right]^{z_{jt}} \Bigg\}, \end{aligned} \tag{3.10}$$

where

$$\begin{aligned} & Prob(m_t \text{ customers buy in period } t | \alpha, \beta_0, \beta, \lambda) \\ = \; & \frac{\left[\lambda_t \sum_{i \in S_t} P_i(S_t, \alpha, \beta_0, \beta)\right]^{m_t} \exp(-\lambda_t \sum_{i \in S_t} P_i(S_t, \alpha, \beta_0, \beta))}{m_t!}. \end{aligned} \tag{3.11}$$

Let $l(t)$ denote length of time period t and λ_t denote the number of arrivals in time period t. We have $\lambda_t = l(t)\lambda$. After we substitute Equation 3.11 into Equation 3.10, we have

$$\begin{aligned}\mathcal{L}(\alpha,\beta_0,\beta,\lambda) &= \prod_{t=1}^{T}\left\{\frac{\left[\lambda_t\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)\right]^{m_t}\exp(-\lambda_t\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta))}{z_{1t}!z_{2t}!\cdots z_{nt}!}\right.\\ &\left.\prod_{j\in S_t}\left[\frac{P_j(S_t,\alpha,\beta_0,\beta)}{\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)}\right]^{z_{jt}}\right\}. \qquad (3.12)\end{aligned}$$

Then the incomplete log-likelihood function can be represented by

$$\begin{aligned}\mathcal{LL}(\alpha,\beta_0,\beta,\lambda) &= \sum_{t=1}^{T}(m_t\log(\lambda_t\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)) - \lambda_t\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta) - \log(z_{1t}!z_{2t}!\cdots z_{nt}!))\\ &+ \sum_{t=1}^{T}\sum_{j\in S_t}z_{jt}\log(\frac{P_j(S_t,\alpha,\beta_0,\beta)}{\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)}). \qquad (3.13)\end{aligned}$$

With Equation 3.9, the last term in the incomplete log-likelihood function can be calculated by

$$\begin{aligned}\sum_{t=1}^{T}\sum_{j\in S_t}z_{jt}\log(\frac{P_j(S_t,\alpha,\beta_0,\beta)}{\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)}) &= \sum_{t=1}^{T}\sum_{j\in S_t}z_{jt}\log(\frac{\frac{\exp(\alpha+\beta_0p_{jt}+\beta y_j)}{\sum_{h\in S_t}\exp(\alpha+\beta_0p_{ht}+\beta y_h)+1}}{\sum_{i\in S_t}\frac{\exp(\alpha+\beta_0p_{it}+\beta y_i)}{\sum_{h\in S_t}\exp(\alpha+\beta_0p_{ht}+\beta y_h)+1}})\\ &= \sum_{t=1}^{T}\sum_{j\in S_t}z_{jt}\log(\frac{\exp(\alpha+\beta_0p_{jt}+\beta y_j)}{\sum_{i\in S_t}\exp(\alpha+\beta_0p_{it}+\beta y_i)})\\ &= \sum_{t=1}^{T}\sum_{j\in S_t}z_{jt}\log(\frac{\exp(\beta_0p_{jt}+\beta y_j)}{\sum_{i\in S_t}\exp(\beta_0p_{it}+\beta y_i)}). \qquad (3.14)\end{aligned}$$

The approach in Newman et al. (2014) decomposes the log-likelihood function into two parts. The first part is the arrival part which models the arrival pattern and the customer choice of purchase or not purchase. The second part is the choice part, which models the product choices of the customers who have decided to purchase. The incomplete log-likelihood function is split with following equations:

$$\mathcal{LL}(\alpha,\beta_0,\beta,\lambda) = \mathcal{LL}_{arrivals}(\alpha,\beta_0,\beta,\lambda) + \mathcal{LL}_{choice}(\beta_0,\beta),$$

$$\begin{aligned}\mathcal{LL}_{arrivals}(\alpha,\beta_0,\beta,\lambda) &= \sum_{t=1}^{T}(m_t \log(\lambda_t \sum_{i\in S_t} P_i(S_t,\alpha,\beta_0,\beta)) - \lambda_t \sum_{i\in S_t} P_i(S_t,\alpha,\beta_0,\beta) \\ &- \log(z_{1t}!z_{2t}!\cdots z_{nt}!)) \qquad (3.15)\end{aligned}$$

and

$$\mathcal{LL}_{choice}(\beta_0,\beta) = \sum_{t=1}^{T}\sum_{j\in S_t} z_{jt}\log(\frac{\exp(\beta_0 p_{jt}+\beta y_j)}{\sum_{i\in S_t}\exp(\beta_0 p_{it}+\beta y_i)}).$$

Theorem 1 in Newman et al. (2014). If D is a fixed subset of alternatives from the complete set of alternatives S, and the choice model is MNL, then maximising the log likelihood function

$$\mathcal{LL}_N = \frac{1}{N}\sum_{t=1}^{N}\log\left[\frac{\exp(\beta_0 p_{jt}+\beta y_j)}{\sum_{i\in D}\exp(\beta_0 p_{it}+\beta y_i)}\right],$$

which yields consistent estimates of β_0^*, β^* under normal regularity conditions.

Step 1: Find the estimates of (β_0,β)=argmax$_{(\beta_0,\beta)}\{\mathcal{LL}_{choice}(\beta_0,\beta)\}$ $\mathcal{LL}_{choice}(\beta_0,\beta)$ can be regarded as a complete log likelihood function for a MNL model without the no-purchase option. Therefore, the function is globally concave with respect to β_0 and β (Ben-Akiva and Lerman, 1985) and we can estimate β_0 and β by maximising the log likelihood function directly. With the Theorem 1 in Newman et al. (2014), the values of β_0 and β are consistent estimates for the original choice model.

Our approach differs from that of Newman et al. (2014) in this step because we assume the market share is estimated exogenously with fixed price vector p^* and is known to the seller. The preference weight of purchasing products from other companies, which is denoted by r, is then also known. From the equation

$$s = \frac{1}{1+r} = \frac{\sum_{i=1}^{n}\exp(\alpha+\beta_0 p_i^*+\beta y_i)}{\sum_{i=1}^{n}\exp(\alpha+\beta_0 p_i^*+\beta y_i)+1},$$

we can find the relation between the value of α and the value of (β_0,β). Therefore, the estimate of α can be calculated by

$$\alpha = -\log(r\sum_{i=1}^{n}\exp(\beta_0 p_i^*+\beta y_i)). \qquad (3.16)$$

Following the method in Newman et al. (2014), with the estimates of α, β_0 and β, we can find the estimates of λ by maximising $\mathcal{LL}_{arrivals}$.

$$\begin{aligned}\mathcal{LL}_{arrivals}(\alpha,\beta_0,\beta,\lambda) &= \sum_{t=1}^{T}(m_t\log(\lambda_t\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)) - \lambda_t\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)\\ &- \log(z_{1t}!z_{2t}!\cdots z_{nt}!))\\ &= \sum_{t=1}^{T}(m_t\log(l(t)\lambda\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)) - l(t)\lambda\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)\\ &- \log(z_{1t}!z_{2t}!\cdots z_{nt}!)). \qquad (3.17)\end{aligned}$$

The first derivative of 3.17 is

$$\frac{\partial\mathcal{LL}_{arrivals}(\lambda|\alpha,\beta_0,\beta)}{\partial\lambda} = \sum_{t=1}^{T}(\frac{m_t}{\lambda} - l(t)\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta))$$

and the second derivative is

$$\frac{\partial^2\mathcal{LL}_{arrivals}(\lambda|\alpha,\beta_0,\beta)}{\partial\lambda^2} = -\frac{\sum_{t=1}^{T}m_t}{\lambda^2}.$$

$\sum_{t=1}^{T}m_t$ is the total purchase during the whole selling period and it should be positive. Therefore, the second derivative is negative and $\mathcal{LL}_{arrivals}(\lambda|\alpha,\beta_0,\beta)$ is globally concave. Also, the estimates of λ can be calculated by setting the first derivative equal to zero. We then have

$$\lambda(\alpha,\beta_0,\beta) = \frac{\sum_{t=1}^{T}m_t}{\sum_{t=1}^{T}l(t)\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)}. \qquad (3.18)$$

Algorithm 3 Two-step algorithm

Step 1: Find β_0 and β with

$$argmax_{(\beta_0,\beta)}\{\mathcal{LL}_{choice}(\beta_0,\beta)\}.$$

Step 2: Find α and λ with equations

$$\alpha(\beta_0,\beta) = -\log(r\sum_{i=1}^{n}\exp(\beta_0 p_i^* + \beta y_i))$$

and

$$\lambda(\alpha,\beta_0,\beta) = \frac{\sum_{t=1}^{T}m_t}{\sum_{t=1}^{T}l(t)\sum_{i\in S_t}P_i(S_t,\alpha,\beta_0,\beta)}.$$

3.3 Estimation with the Markov Chain Monte Carlo Method

To solve a dynamic pricing problem with unknown choice behaviour, we need to re-estimate the choice model regularly. All the information, which has been used in estimation, needs to be stored and used in the next estimation. This situation can be avoided with a Bayesian method and by storing the information as prior distributions. In this section, we apply a Markov Chain Monte Carlo (MCMC) algorithm to estimate the MNL model. A related paper is Letham et al. (2015) that jointly estimates arrival rate and substitution behaviour with stockout effect. The authors employ a stochastic gradient MCMC algorithm. They show that with their methods, the parameter of base utility is unidentifiable without the information of market share. With the idea of a two-step algorithm in the previous section, we can solve the problem with a standard MCMC method and provide a unique solution, if the information of market share is provided. This provides a simpler method of solving the Bayesian version of the problem than that provided by Letham et al. (2015).

With the idea from the two-step algorithm, we can decompose the estimates of β_0, β, α and λ into two steps. The first is to estimate β_0 and β with a MNL model which ignores the no-purchase option. We can use all of the available sales data to calculate these estimates and the latter problem can be solved with a standard MCMC method. Given the estimates of β_0 and β, the estimates of α and λ can be calculated with Equation 3.16 and Equation 3.18.

The basic idea behind Bayesian parameter estimation is treating the parameters as random variables. Given prior parameter distributions, which represent the current information we have, the new information is stored in the likelihood function and used to update our confidence in the parameters to give a posterior distribution.

The prior distribution
We assume that the prior of parameters (β_0, β) follows a multivariate normal distribution. Let π_0 denote the prior distribution. We can write the prior as

$$\pi_0(\beta_0, \beta) \propto |A|^{1/2} \exp\{-\frac{1}{2}\left[(\beta_0, \beta) - (\bar{\beta}_0, \bar{\beta})\right]' A \left[(\beta_0, \beta) - (\bar{\beta}_0, \bar{\beta})\right]\},$$

where $(\bar{\beta}_0, \bar{\beta})$ is the mean of the prior distribution and A is the covariance matrix.

If we have no information about the parameters of (β_0, β), we can start our Bayesian inference with a vague prior distribution, e.g., multivariate normal distributions with mean zero and a covariance matrix with no correlation and a high variance. If we have information about parameters before the Bayesian inference, e.g., some survey data, the information can be incorporated into the prior distribution.

The Likelihood Model

Let z_{jt} denote the number of product j which is purchased in time period t; p_{jt} represents the price for product j in time period t; and y_j is the attribute of product j, except the price. Given that the offered product set in time period t is S_t, the likelihood function of the problem without the no-purchase option can be presented by

$$\mathcal{L}(\beta_0, \beta | z, p, y) = \prod_{t=1}^{T} \prod_{j \in S_t} z_{jt} \frac{\exp(\beta_0 p_{jt} + \beta y_j)}{\sum_{i \in S_t} \exp(\beta_0 p_{it} + \beta y_i)}. \tag{3.19}$$

Let $\pi(\beta_0, \beta | z, p, y)$ represent the posterior distributions of β_0 and β, and from Bayes' Theorem, we have

$$\begin{aligned} Posterior &\propto Likelihood \times Prior \\ \pi(\beta_0, \beta | z, p, y) &\propto \mathcal{L}(\beta_0, \beta | z, p, y) \pi_0(\beta_0, \beta) \\ \pi(\beta_0, \beta | z, p, y) &= \frac{\mathcal{L}(\beta_0, \beta | z, p, y) \pi_0(\beta_0, \beta)}{\int \mathcal{L}(\beta_0, \beta | z, p, y) \pi_0(\beta_0', \beta') d(\beta_0', \beta')}. \end{aligned} \tag{3.20}$$

Substituting Equation 3.19, we have

$$\pi(\beta_0, \beta | z, p, y) = \frac{\prod_{t=1}^{T} \prod_{j \in S_t} z_{jt} \frac{\exp(\beta_0 p_{jt} + \beta y_j)}{\sum_{i \in S_t} \exp(\beta_0 p_{it} + \beta y_i)} \pi_0(\beta_0, \beta)}{\int \prod_{t=1}^{T} \prod_{j \in S_t} z_{jt} \frac{\exp(\beta_0' p_{jt} + \beta' y_j)}{\sum_{i \in S_t} \exp(\beta_0' p_{it} + \beta' y_i)} \pi_0(\beta_0', \beta') d(\beta_0', \beta')}.$$

For the MNL model, there is no closed-form solution for the multi-dimensional integral. Therefore, a MCMC algorithm is used to calculate the Bayesian inference.

Markov Chain Monte Carlo Algorithm

For our problem, we cannot obtain a closed-form solution for the posterior distribution of parameters. We use the MCMC algorithm to generate a sample of parameters whose target distribution is the posterior we need. For the MNL model, Metropolis-Hastings sampling (Gamerman, 1997), data augmented Metropolis-Hastings sampling (Scott, 2011), Gibbs sampling (Dellaportas and Smith, 1993) and data-augmentation and Gibbs sampling (Frühwirth-Schnatter and Frühwirth, 2007) have been applied to do the estimation with Bayesian method. We use the Metropolis-Hastings algorithm which was first introduced by Metropolis et al. (1953).

Let $x_k, k = 1, \cdots, K$ denote the points sampled by the Markov Chain where K is the number of iterations. The proposal distribution is given by $q(y|x_k)$ which denotes the conditional probability of proposing a point y given point x_k. Let $a(x_k, y)$ denote the acceptance probability to decide whether to accept the sampled value or not. With these notations, the Metropolis-Hastings algorithm can be shown in Algorithm 4.

Algorithm 4 Metropolis-Hastings

Set $k = 0$, initialise x_0

for each iteration t **do**

  Sample a point z from proposal distribution $q(z|x_k)$

  Sample a uniform(0,1) random variable U

  if U is less than the accept probability $a(x_k, z) = min\{1, \frac{\pi(z)q(x_k|z)}{\pi(x_k)q(z|x_k)}\}$, **then**

    Set $x_{(k+1)} = z$

  else Set $x_{(k+1)} = x_k$

  end if

  Set $k = k + 1$

end for

The initial point of x_0 can be an arbitrary point, since we will set a burn-in period. Any points in the burn-in period will be discarded.

For the initial update, we have vague information about the variables, so an uninformative prior is used as described above. When new information is available, we can update our belief of the choice model by running the MCMC method again with a new prior distribution. Our prior knowledge in each update is the knowledge we had at the end of the last update, which can be represented by the last posterior distribution. So we use the last posterior distribution as the new prior distribution in our Bayesian inference. Information before the last Bayesian update is stored in the prior distribution and information after the last update is included in the likelihood function.

There is no guiding rule on the construction of proposal distribution. However, a proposal distribution with more structure of the problem can converge faster than the one with less structure. We use the multivariate normal distribution as the proposal distribution with the mean at the mode of $\pi(x)$. The covariance matrix of the proposal distribution is a scalar multiplied by the inverse Hessian matrix at the mode that can be written as

$$\left[-\frac{d^2 log\pi(x)}{dx'dx}\right]^{-1}.$$

The scalar is determined by trial and error to ensure that the acceptance rate is about 23.4%, which is the optimal acceptance rate for the random-walk Metropolis.

3.4 Results

In this section, we introduce two examples we used in the tests. Then we test the performance of the EM algorithm without stockout effect, the EM algorithm with stockout effect, the two-step algorithm, and the MCMC method.

3.4.1 Examples with Train data and Simulated data

Example with Train data

In this section, we introduce the sales data from a railway company. We try to estimate the customer preference between trains which have different departure times and departure days. The sales data include information about the train number, origin and destination pair, departure time, arrival time, departure date and the numbers of booked tickets at given snapshot dates. For dates less than one week from the departure date, numbers of bookings are recorded every day. For snapshot dates that are more than one week from the departure date, information is recorded every two to seven days. According to the data, the whole booking period is set to start 84 days before the departure date and we use a time period of 30 seconds. Comparing to the 84 booking days for the trains, this time period is small enough and we can assume that there is at most one arrival in each time period. With this assumption, we can fit the arrival pattern with a Poisson arrival process. We propose a simulation model which mimics the selling system. A Bernoulli process is used to allocate the bookings in weeks or daily snapshots to 30-second time periods.

In the sales data, we only have a snapshot of prices being charged on 21st July 2011. We do not have the number of bookings at each price. In addition, for the trains with different departure dates, the snapshot of prices could be the price offered one day before the departure date, or it could be one week before the departure date. We could not simply compare these prices for the trains. Therefore, we ignore the price as an attribute in the MNL model. This will affect the quality of estimation. A better estimation could be obtained, if we have access to the price information in the whole booking period.

In the estimation, we use data for trains travelling from London to Newcastle and we assume customers only consider the trains departing on 28 consecutive departure days (23rd May 2011 to 19th June 2011). Then, we use the estimated choice model to predict the booking numbers and compare these with the actual numbers of bookings in seven consecutive departure days (20th June 2011 to 26th June 2011). The choice set C_n for customer n includes all trains departing on given days and all trains travelling from London to Newcastle. In this study, departure time consists of four non-overlapping time slots: morning (4am-10am), midday (10am-3pm), teatime (3pm-7pm) and late (7pm-10pm). This setting is based on the categories defined in the data file from the railway company. These non-overlapping time slots could be easily extended into overlapping time slots by adding convex weights to different time slots (See Vulcano et al., 2010). The data we used to estimate represent four weeks. The trains that depart on Monday mornings in different weeks have the same attributes and are treated together. The other trains are handled in the same way. The attributes of the customer choice model are defined in Table 3.1.

Attribute	Description
β_1	Indicator for trains departing on Monday
β_2	Indicator for trains departing on Tuesday
β_3	Indicator for trains departing on Wednesday
β_4	Indicator for trains departing on Thursday
β_5	Indicator for trains departing on Friday
β_6	Indicator for trains departing on Saturday
β_7	Indicator for trains departing on Sunday
β_8	Indicator for Morning train
β_9	Indicator for Midday train
β_{10}	Indicator for Teatime train
β_{11}	Indicator for Late train

Table 3.1: Definition of attributes for train case

The utility of product j can be presented by

$$\begin{aligned} u_j = & \beta_1 x_{j1} + \beta_2 x_{j2} + \beta_3 x_{j3} + \beta_4 x_{j4} + \beta_5 x_{j5} + \beta_6 x_{j6} \\ & + \beta_7 x_{j7} + \beta_8 x_{j8} + \beta_9 x_{j9} + \beta_{10} x_{j10} + \beta_{11} x_{j11}, \end{aligned} \tag{3.21}$$

where β_i is the coefficient for x_i which shows the value of attribute. For example, considering a train departing in Friday morning, we have $x_{j5} = 1$, $x_{j8} = 1$ and all the other attributes have a value of 0. Therefore, we have

$$x_{j7} = 1 - (x_{j1} + x_{j2} + x_{j3} + x_{j4} + x_{j5} + x_{j6})$$

and

$$x_{j11} = 1 - (x_{j8} + x_{j9} + x_{j10}).$$

Therefore, we reformulate Equation 3.21 and re-label the parameters. Equation 3.21 can be presented by

$$\begin{aligned} u_j = &\ \beta_0 + \beta_1 x_{j1} + \beta_2 x_{j2} + \beta_3 x_{j3} + \beta_4 x_{j4} + \beta_5 x_{j5} + \beta_6 x_{j6} \\ &+ \beta_7 x_{j7} + \beta_8 x_{j8} + \beta_9 x_{j9}, \end{aligned} \tag{3.22}$$

and the definition of the parameters can be found in Table 3.1. β_0 denotes the base utility for any purchase option. We use this notation in the two-step algorithm and decompose the estimation of the MNL model. In the first step, we estimate parameters in MNL except β_0, and then we estimate β_0 in the second step.

The parameters for Sunday trains departing in the "Late" time period are omitted in Equation 3.22. We use the Sunday trains which depart in the "Late" time period as reference trains.

Example with simulated data

To test the performance of the estimation algorithms, we apply the algorithms with simulated data and compare the estimation values with the true values which we used to generate the simulated data.

In this example, we generate a simulated data, which has the same setting with the one in Vulcano et al. (2010). The simulated data considers 14 flights which depart on 3 consecutive departure days and departure time is split into 4 overlapping time slots: morning (between 5 a.m. and 11 a.m.), noon (between 9 a.m. and 3 p.m.), afternoon (between 1 p.m. and 7 p.m.), and evening (between 5 p.m. and midnight). Prices of tickets are included as an attribute in this example. A fixed price vector are applied in the simulation. The value of the price vector can be found in Appendix A.1.

Based on discussion with the author, a Bernoulli random variable is used in simulation to determine if there is an arrival in a time period or not. This arrival pattern assumes that there is at most one arrival in each time period. This is valid for a sufficiently small time period and if we ignore group bookings. We follow the setting in Vulcano et al. (2010) and use a time period of 10 minutes with the total booking period is 10 days. The fares used by Vulcano et al. (2010) are not explicit. We take the average of min open fare and max open fare provided in their paper and use the value as the fare in the test. For the test with unlimited inventory level, we assume the products have large initial inventory that cannot be reached, like 10000. For the tests with limited inventory level, we select 4000 as the initial inventory level.

3.4.2 Estimation without stockout effect

In this section, we apply the EM algorithm without stockout effect (Vulcano et al., 2010) over the sale data from a railway company and the simulated data.

1.Example of train data

In this test, we estimate the customer choice behaviour over sale data from a railway company. We apply the EM algorithm from the initial point: $\beta_0 = 0$, $\beta = 0$ and $\lambda = 0$. The result from the EM algorithm is shown in Table 3.2.

Parameter	Description	Est. value
β_0	Base utility for any purchase option	8.17
β_1	Indicator for trains on Monday	-0.06
β_2	Indicator for trains on Tuesday	-0.08
β_3	Indicator for trains on Wednesday	-0.06
β_4	Indicator for trains on Thursday	0.25
β_5	Indicator for trains on Friday	0.33
β_6	Indicator for trains on Saturday	-0.29
β_7	Indicator for Morning train	0.08
β_8	Indicator for Midday train	0.81
β_9	Indicator for Teatime train	0.77
λ	Arrival rate	0.19

Table 3.2: Estimation for the choice model with railway data

Figure 3.1 presents the goodness of fit of the estimation by providing the true observed bookings and predicted bookings.

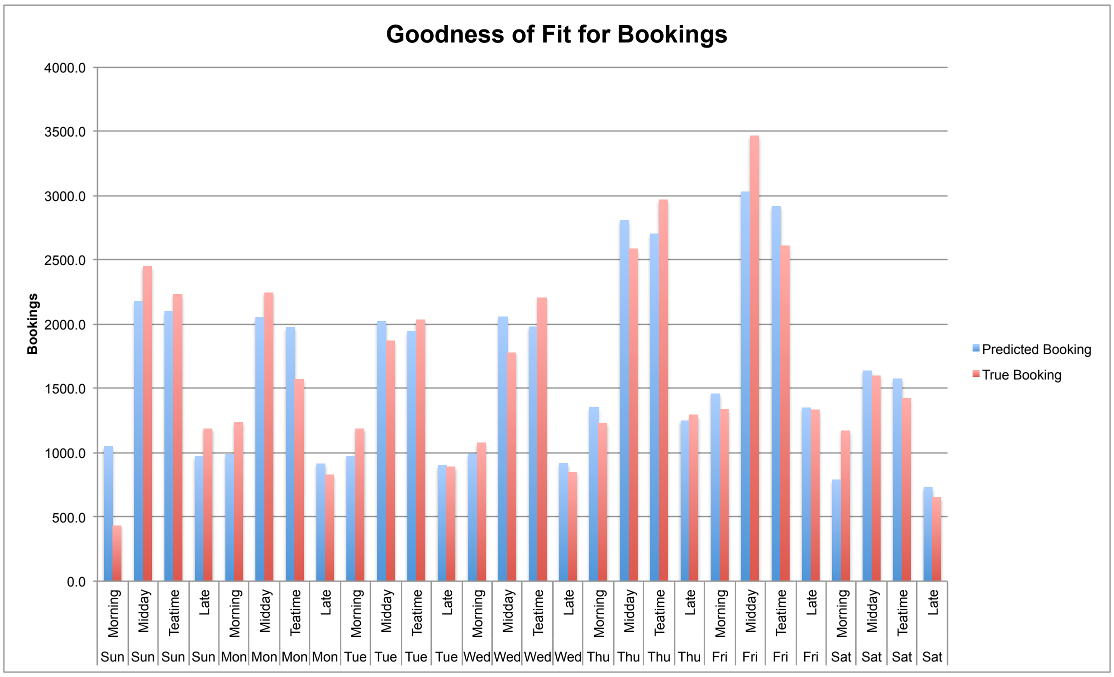


Figure 3.1: Goodness of fit for bookings

We notice that the true bookings on Sunday morning are significantly lower than the prediction. We check the data and find that only three trains depart on Sunday morning, while eight trains depart on Monday morning. We believe that this is the reason for the poor prediction.

Figure 3.2 presents the true bookings over seven consecutive departure days (20th June 2011 to 26th June 2011) and the predicted bookings based on the estimation with the data from 28 consecutive departure days (23rd May 2011 to 19th June 2011). We can find the prediction is not good enough for the trains departing on Sunday. The customer may have a different preference of departure time on Sunday. The structure of the MNL model needed to be reconsidered.

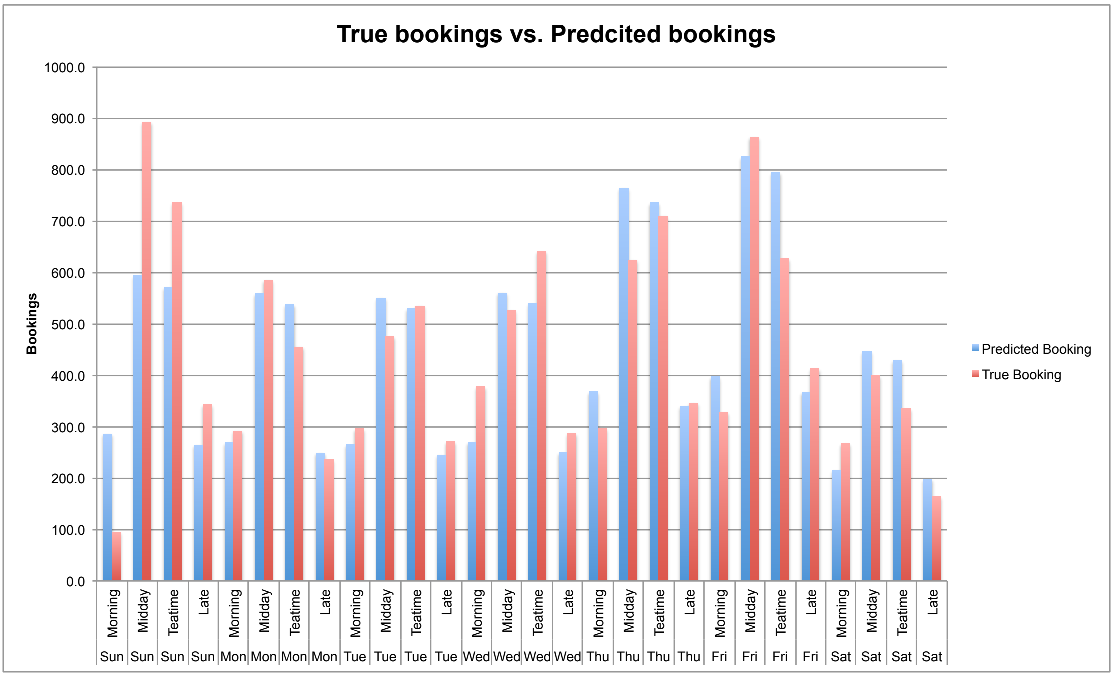


Figure 3.2: True bookings vs. Predicted bookings

2. Example with simulated data

In this section, we test the performance of EM algorithm with simulated data. Table 3.3 gives the estimation of the MNL model from the EM algorithm. The percentage bias between the true values and the estimated values shows that the EM algorithm performs pretty well.

Parameter	Description	True value	Est. value	Percent bias
β_0	Base utility	1.4	1.4981	7.0038
β_1	Base fare	-1	-1.0496	4.9618
β_2	Morning flight	0	-0.0079	
β_3	Noon flight	0.2	0.2139	6.9349
β_4	Afternoon flight	-0.2	-0.2157	7.8338
β_5	Day 1	-0.3	-0.2692	-10.2756
β_6	Day 2	-0.6	-0.6070	1.1656
λ	Arrival rate	0.3	0.2987	-0.4444

Table 3.3: Estimation result for the problem with a simulated data file

Figure 3.3 presents the goodness of fit of the estimation by providing the true expected bookings and predicted bookings from estimation. The difference between the true expected bookings and predicted bookings is negligible.

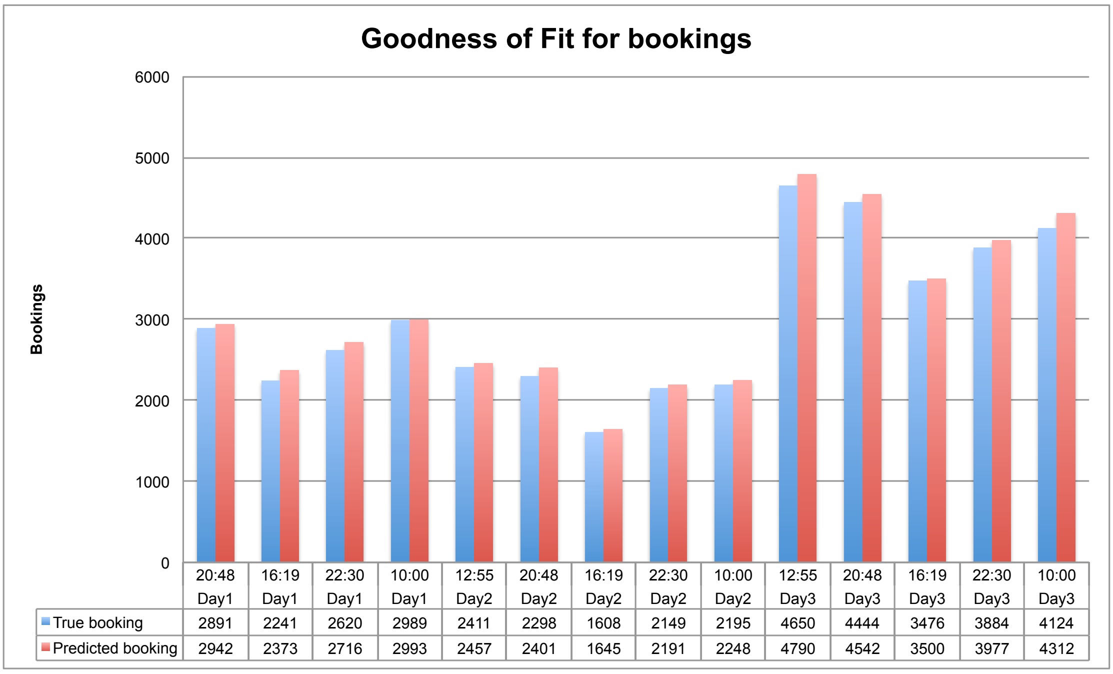


Figure 3.3: Goodness of fit for bookings

Figure 3.4 shows the goodness of fit of the estimation by providing the true expected utilities and predicted utilities.

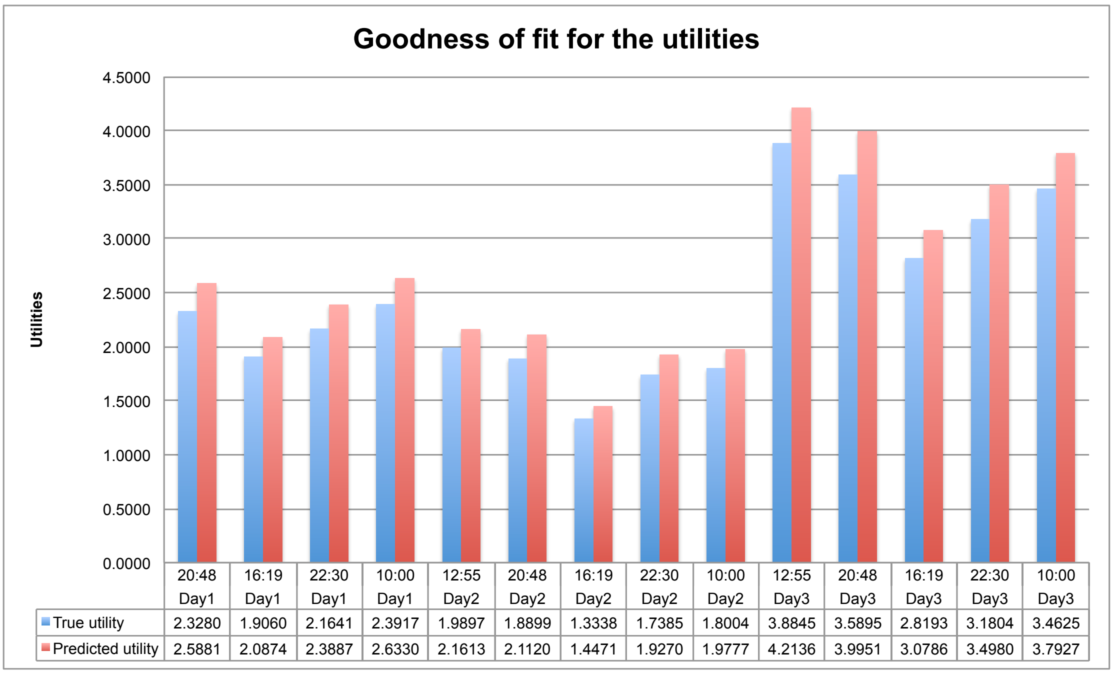


Figure 3.4: Goodness of fit for utilities

The result above is obtained with a fixed initial point $\beta_0 = 1, \beta = 0$ and $\lambda = 0.3$, which is the same one in Vulcano et al. (2010). In Vulcano et al. (2010), they also test with a starting point which has a different value of $\lambda = 0.6$ and state that the estimation is "noticeably good". We test more starting points and show the percentage bias between the true values and estimated values. The data used in the test is the simulated data.

In Figure 3.5, we start with $\beta_0 = 1$ and $\beta = 0$ and change the starting value of λ from 0.1 to 0.9. Figure 3.6 shows the percentage bias for the test with starting point $\beta = 0$ and $\lambda = 0.3$ and the initial point of the β_0 changes from -5 to 5. In Figure 3.7, we keep $\beta_0 = 1$ and $\lambda = 0.3$ and the starting value of β is changed from -5 to 5. We do not present the percentage bias of β_2 because it has a true value of 0. The estimation value of β_2 has a performance that is similar to the other value in β.

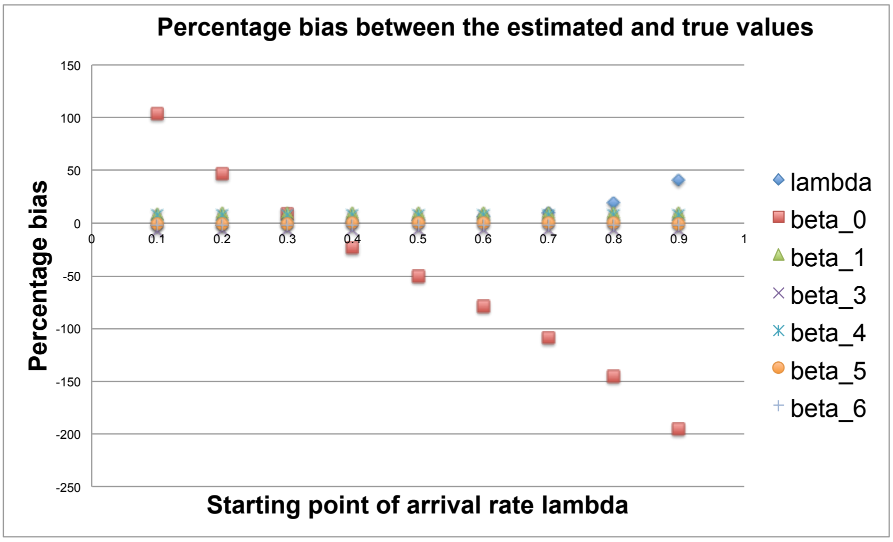


Figure 3.5: Percentage bias with different starting point of arrival rate

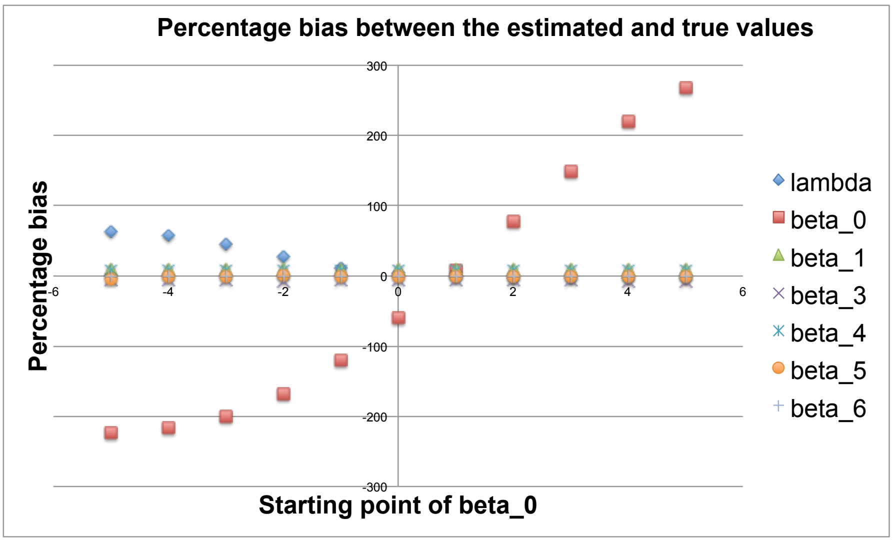


Figure 3.6: Percentage bias with different starting point of β_0

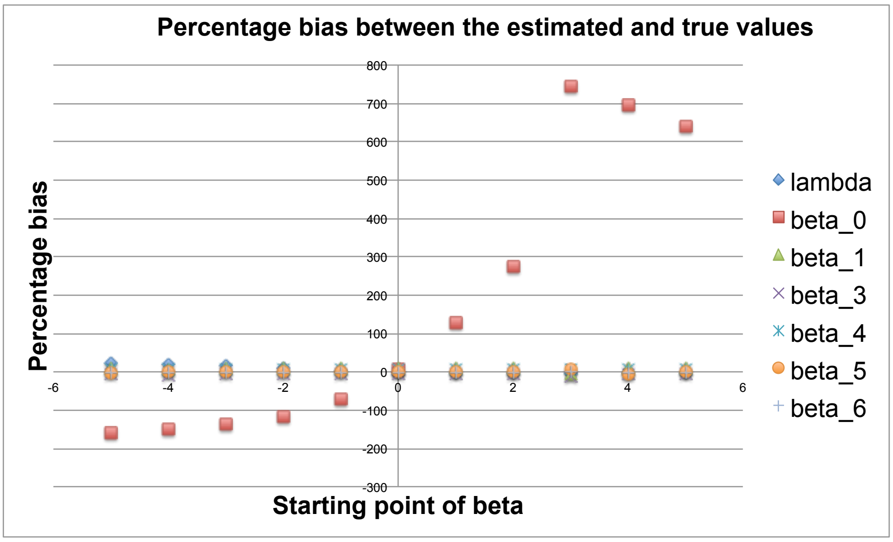
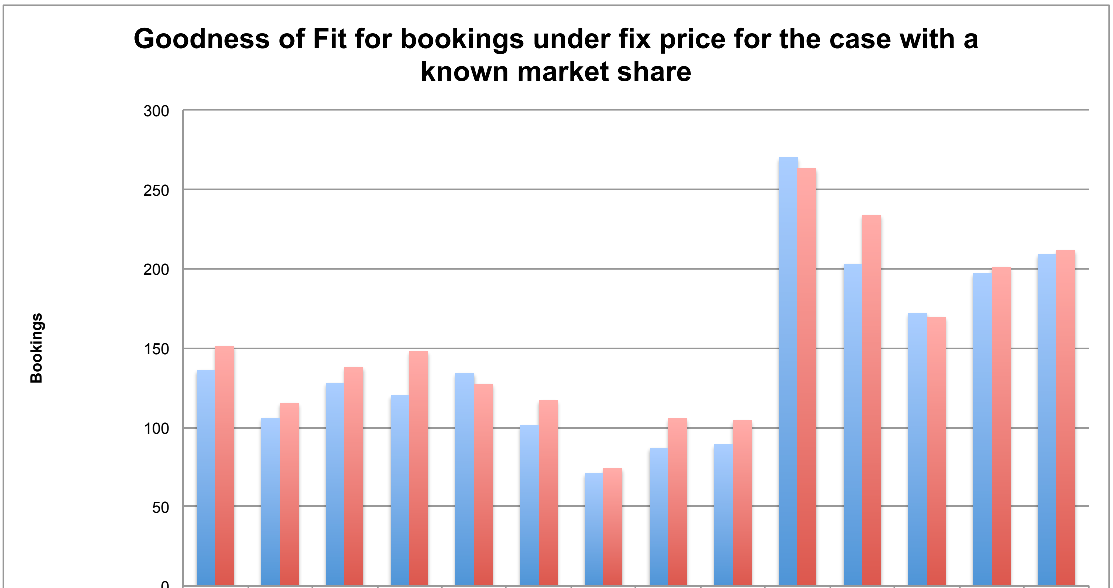


Figure 3.7: Percentage bias with different starting point of β

From these three figures, we can see that the estimates of β are stable with different starting points, and the estimates of λ and β_0 are highly dependent on the starting point. If we regard β as the parameter of a new MNL model without a no-purchase option, then the estimate of β will be an estimate of the MNL model with full information. From Theorem 1 in Newman et al. (2014), the estimate of β is consistent with the estimate of the original problem. Therefore, the estimate of β is stable. The idea in Newman et al. (2014) is to decompose the estimation into two steps. The first step is to estimate β and return a good estimate. The second step is to estimate λ and β_0 and multiple maxima may exist. This problem is generated from the unobservable no-purchase arrivals. Let M denote the number of customers who arrive in the selling system and m denote the number of customers who buy a product from a company. Let q represent the probability that a customer chooses to purchase from this company. We have $m = Mq$. For the problem with incomplete booking data, we can only access the value of m. Both M and q are unknown to us. So any two value pairs which have a fixed ratio M/q can be a solution for the problem. This makes it impossible to exclude some solutions with any estimation method. This problem can be solved by providing M or q. Vulcano et al. (2012) solve this problem by adding information related to market share, and we do the same.

3.4.3 Estimation with stockout effect

In this test, we use the simulated data to test the performance of the EM algorithm which considers two types of missing data. To consider stockout in the problem, we propose a simulation model which mimics the booking system. The customers arrive in a Poisson process and decide which product they will buy or leave the booking system without a purchase based on a choice model. After some product is sold out before the end of the booking period, we exclude the product and recalculate the purchase probability. Figure 3.8 presents the goodness of fit of the estimation by providing the true expected bookings and predicted bookings.



	Flight1	Flight2	Flight3	Flight4	Flight5	Flight6	Flight7	Flight8	Flight9	Flight10	Flight11	Flight12	Flight13	Flight14
True booking	136	106	128	120	134	101	71	87	89	270	203	172	197	209
Predicted booking	151	115	138	148	127	117	74	105	104	263	234	169	201	212

Figure 3.8: Goodness of fit for bookings under fixed price for the case with a known market share

Slightly different from the simulated data in the previous section, we allow the price vector to vary during the booking period. We test how the price policy affects the quality of estimation. Two types of price vector are offered to the customers. The first is the fixed price vector which is generated with uniform distributions and remains the same throughout the selling period. We use the minimum open fares and maximum fares provided in Vulcano et al. (2010) as the bounds of uniform distribution. The values of the bound can be found in Appendix A.1. The second price vector is a random price vector, which is generated from uniform distributions with the bounds of minimum open fare and maximum open fare that are provided in the Appendix A.1. The price vector is updated each day. Only data for the first 50 days are used in the test to have a obvious results. Table 3.4 shows the estimation under the random price vector and fixed price vector. The parameter estimates for the problem with random price vector are better than the estimates with fixed price vector.

		Random Price		Fixed price	
Description	True value	Est.value	%Bias	Est.value	%Bias
base utility	1.40	1.41	0.90	1.47	4.88
price	-1.00	-1.06	5.94	-1.25	24.86
morning	0.00	0.04		-0.13	
noon	0.20	0.22	10.77	0.27	36.78
afternoon	-0.20	-0.16	-21.30	-0.27	35.79
day 1	-0.30	-0.33	9.01	-0.27	-9.42
day 2	-0.60	-0.58	-3.24	-0.64	6.48
arrival rate	0.30	0.30	-1.17	0.29	-4.97

Table 3.4: Estimation and percentage biases

3.4.4 Estimation with the two-step algorithm

In this section, we test the performance of the two-step algorithm described in Section 3.2. First, we compare it with the estimates from the EM algorithm in Vulcano et al. (2012). Then, we show how the data volumes affect the estimation result. Last, we apply the two-step algorithm with random price vectors and a fixed price vector and compare the results to find how the provided price vectors affect the estimation quality. The example we used to test the two-step algorithm is the simulated data in the previous section.

Compare with EM algorithm with stockout effect

First, we compare the results from the two-step algorithm and the EM algorithm in Vulcano et al. (2012). The test is performance with 1000 days which is the same setting used in Vulcano et al. (2010). Table 3.5 shows the estimation and percentage biases from the two-step algorithm and the EM algorithm. We can find that both of the algorithms provide results with good quality. All the estimates have a percentage bias which is less than 10%. Therefore, both of the algorithms can have good estimates of the customer choice behaviour, if enough sales data are provided.

		Two-step algorithm		EM algorithm	
	True value	Est.value	%Bias	Est.value	%Bias
base utility	1.40	1.42	1.12	1.42	1.28
price	-1.00	-1.09	9.35	-1.11	9.57
morning	0.00	0.01		-0.01	
noon	0.20	0.21	6.88	0.19	3.26
afternoon	-0.20	-0.20	-0.93	-0.21	7.42
day 1	-0.30	-0.30	1.39	-0.31	-2.99
day 2	-0.60	-0.61	1.20	-0.59	1.05
arrival rate	0.30	0.30	0.23	0.30	0.19

Table 3.5: Estimates and percentage biases from the two-step algorithm and the EM algorithm

Effect of data volume

Next, we test the quality of estimation with different data volumes. We use 10 minutes as the length of the time period, and the arrival rate is set to 0.3. We test with data generated for 1 day, 10 days, 100 days and 1000 days. For each length of booking period, we run the simulation 10 times and report the percentage bias of the average of estimates.

Table 3.6 presents the bias of the average estimates from the two-step algorithm with different data volumes.

	1 day	10 days	100 days	1000 days
base utility	-10.69	2.73	-1.64	1.01
price	-37.92	34.97	-16.08	7.22
noon	57.50	18.83	5.87	-0.35
afternoon	32.86	-6.03	1.94	1.48
day 1	-4.26	-7.97	5.13	-1.27
day 2	26.42	5.55	4.04	-0.30
arrival rate	-4.62	-2.27	-0.08	0.01

Table 3.6: Biases of average estimates from the two-step algorithm with different data volumes

Comparison of estimation with random price vector and fixed price vector

In this part, we test how the price vector offered affects the quality of estimation. The fixed price vector is generated randomly with uniform distributions and remains unchanged during the test. For the random price vectors, we update the price vector every day with uniform distributions and the price vector keeps unchanged during the day. We use the minimum open fares and maximum fares provided in Vulcano et al. (2010) as the bounds of uniform distribution. In this test, we use 10 days as the length of selling period. Table 3.7 shows the comparison of estimations with random price vectors and fixed price vector. We note that the estimation with random price vectors outperform the estimation with fixed price vector. The results convince us that the price diversity can improve the quality of estimation. Therefore, we need to find a good price strategy than that can estimate the choice model more accurately and efficiently.

		Random Price		Fixed price	
Description	True value	Est.value	%Bias	Est.value	%Bias
base utility	1.40	1.21	13.39	1.26	9.87
price	1.00	-0.88	188.17	-0.02	102.06
noon	0.20	0.30	-48.47	0.11	47.29
afternoon	-0.20	-0.02	92.24	-0.31	-53.79
day 1	-0.30	-0.04	87.50	-0.11	64.52
day 2	-0.60	-0.50	17.02	-0.55	8.47
arrival rate	0.30	0.30	1.52	0.31	-2.48

Table 3.7: Comparison of estimations with random price vectors and fixed price vector

3.4.5 Estimation with the MCMC method

In this section, we use the simulated data with the size of 1000 days to test the performance of the MCMC method. Figure 3.9 shows 3000 iterations from the MCMC method. The burn-in is taken to be the first 2000 iterations. We can observe convergence of the samples from the figure. The running time of the MCMC method is 530.611s (implemented in MATLAB on a OSX 10.9, 2.4 GHz). Table 3.8 shows the means and variances for the normal distributions fitting with the data after the burn-in periods and the percentage bias between the mean value and the true values.

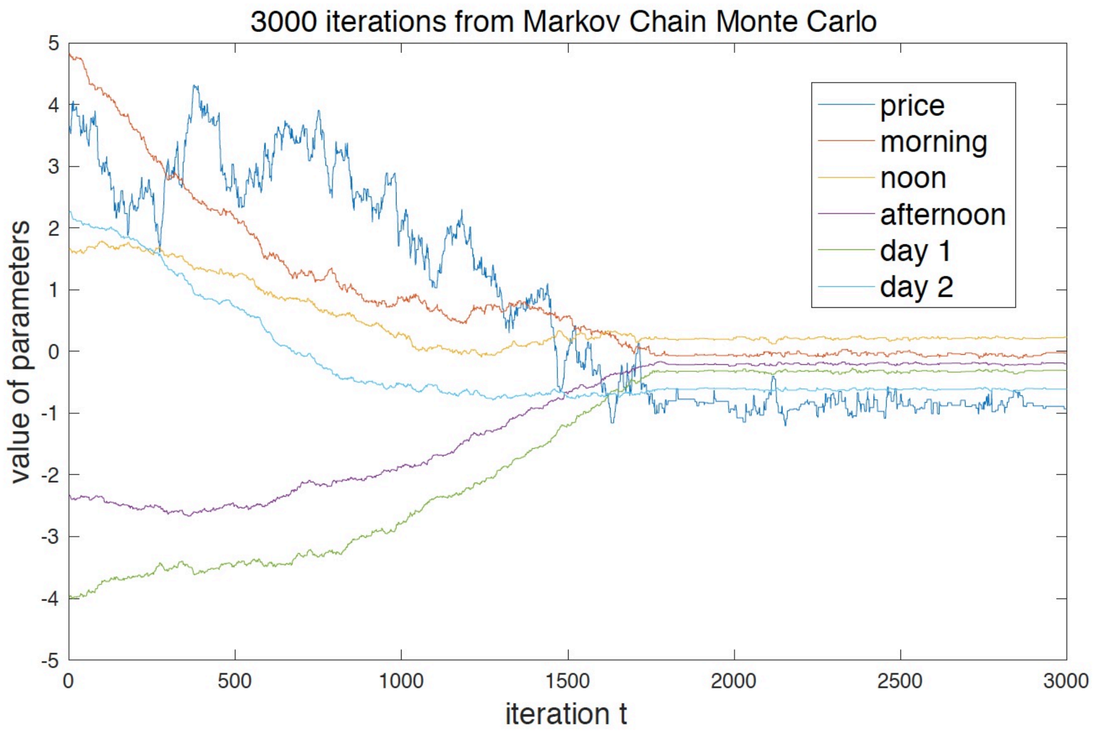


Figure 3.9: 3000 iterations from MCMC sampling

	mean	variance	true value	percentage bias
price	-0.88	0.017901	-1	-12.47
morning	-0.05	0.000654	0	
noon	0.21	0.000330	0.2	6.93
afternoon	-0.20	0.000242	-0.2	0.31
day 1	-0.32	0.000290	-0.3	6.78
day 2	-0.61	0.000233	-0.6	2.05

Table 3.8: Distribution fitting and percentage bias between the mean value and the true value

3.5 Conclusion

In this chapter, we solve the estimation problem with stockout effect with a two-step algorithm and show that the estimation problem can be transformed to a problem that maximises a globally concave function. It is therefore possible to apply maximum likelihood estimation or the MCMC method without worrying about missing data.

The main contribution in Chapter 3 is applying the idea presented in Newman et al. (2014) to decompose the estimation method and solve the problem with a globally concave function. Therefore, we can apply maximum likelihood estimation directly to estimate the unknown parameters with market share information. In addition, a standard Metropolis-Hastings algorithm in the MCMC can be applied to solve the estimation problem with a unique solution. Letham et al. (2015) also use MCMC to solve this problem, but without the market share information, the parameter for base utility is unidentifiable.

The estimation methods are assessed using a simulation study and real data from a train company. Results provide us with a sense of estimation performance and how much data are needed to obtain good estimates. The comparison with estimation methods is also presented. This suggests that, under a fixed price vector and a random price vector, offering a diversity of price vectors can improve the quality of estimation.

The two-step algorithm solves the estimation problem by maximising a globally concave function that can be solved with existing software, and only a unique solution exists. Firms can estimate the demand and choice behaviour more easily with the two-step algorithm than they can with the existing EM algorithms and the unique estimation result helps the firms make smart decisions. We also apply a MCMC method to provide posterior distributions of unknown parameters. With the Bayesian method, prior information, like survey data, can be incorporated in the prior distribution. When new information is available, firms do not need to re-estimate the parameters with all the data available because the sales information before the estimation is stored in the prior distribution. The posterior distribution also provides uncertainty of the unknown parameters to firms. The estimation of customer choice model affords the company a better understanding of product attributes such as day of week, time of day, and price. Another benefit of the estimation is the ability to find the most attractive new products for the company. We do not need to collect sale information for new products or re-estimate the customer choice model.The estimation of customer choice model also helps the company to improve its revenue with a choice-based RM. In the booking period, some products may be sold out before the end of the booking period. The demand for the products which are available can be calculated easily with the known customer choice model.

The limitation in this chapter is the assumption that we have an exogenous estimate of the aggregate market share. However, the estimation of market share is difficult to obtain in practice. In addition, we assume that the changes in our firm's prices have a limited impact on competitors' prices. Therefore, the new market share can be calculated based on our firm's prices only.

Chapter 4

Choice-based Dynamic Pricing

With the development of the Internet, sellers have more flexibility to change the prices of their products. Compared to the traditional RM models with quantity controls, Gallego and van Ryzin (1994) argue that it is more profitable to use pricing methods to control the sales, since the pricing methods control the sales by increasing the price. It is important to solve the pricing problems for practical implementation. The development of the Internet also gives the customers easier access to information about the products. Customers can compare the characteristics of products, which include price information, with little cost. When the customers make their choices from the products that are substitutable, the choice behaviour can be modelled, estimated and used to improve the performance of RM models. In this chapter, we solve a dynamic pricing problem with substitutable products. The products are assumed to be perishable, which means that the value of the product can be neglected after a certain time point. The choice behaviours of the customers are assumed to be homogeneous and modelled with a Multinomial Logit (MNL) model, which could be estimated using methods presented in Chapter 3. Incorporating a choice model into the optimal pricing algorithm, we can improve the expected revenue by solving a choice-based dynamic pricing problem.

In this chapter, we focus on a choice-based dynamic pricing problem of selling fixed inventories of substitutable products. The aim of the problem is to maximise total revenue in a finite time given estimates of the customer choice preferences. The exact solution of the choice-based dynamic pricing problem is impossible to obtain within realistic computational time for a practical problem. To solve the this problem, Zhang and Cooper (2009) propose an approximation algorithm as discussed in Section 4.2. The algorithm decomposes the dynamic pricing problem for multiple products into several one-product dynamic pricing problems. Separable lower bounds and upper bounds for the one-product problem are provided to construct the approximation of the value function. We develop this method further to improve the computational speed of the algorithm.

Sen (2013) proposes an approximation method that is similar to the method in Zhang and Cooper (2009) to solve a dynamic pricing problem with a single product. He approximates the optimal expected revenue with a combination of the lower bounds and upper bounds for the value function, where the lower bounds are approximated by dividing the whole selling period by the number of inventory levels. In contrast to Zhang and Cooper (2009), the upper bound in Sen (2013) is provided by Gallego and van Ryzin (1994). We make use of work in Sen (2013) in developing our method. In this chapter, we do not use the data from the railway company because the information of inventory level is unavailable in the data.

The remainder of this chapter is organised as follows. In Section 4.1, we describe the problem we studied and introduce some notations. Section 4.2 presents an existing approximation algorithm for this pricing problem introduced by Zhang and Cooper (2009). In Section 4.3, we propose a new approximation method which can reduce the computation time significantly. In Section 4.4, we test our method with two examples and compare revenue and computation time with the approximation in Zhang and Cooper (2009).

4.1 Problem Formulation

We study a problem in which a seller offers n perishable products to customers with the aim of maximising the expected revenue. The products are sold in a finite selling period and the product salvage value is set to zero at the end of the selling period. The capacity of the products is determined at the beginning of the selling period and the seller is unable to reorder more products during the selling period. We assume the products offered by the seller are substitutable. That is, customers can shift to other products when their first choice is unavailable or provided at too high a price.

We assume the customers arrive into the selling system following a Poisson process with constant rate λ. It is relatively straightforward to generalise to a non-homogeneous arrival rate by using time periods of varying lengths (Leemis, 1991). Each customer purchases a unit of product based on a certain customer choice preference that is modelled using the MNL model. We divide the whole selling period into T small time periods which are small enough that we can assume that there is at most one arrival in each time period. The first time period is period T, and the last period is period 1.

We denote the allowable price range for product $i \in \{1, \cdots, n\}$ by $\mathbb{P}_i = (r_i^{min}, r_i^{max})$. The price vector offered to the customers is $\mathbf{r} = \{r_1, \cdots, r_n\}$. If a product sells out we set its price to a null price and keep the prices of other products unchanged. The null price is a price which is large enough to ensure that the purchase probability is 0. We assume customers are homogeneous in their preferences. At time t, the probability that a customer chooses product i is denoted by $P_t^i(\mathbf{r})$, and the probability that there is no selling in time period t is denoted by $P_t^0(\mathbf{r})$. Then $P_t^i(\mathbf{r})$ can be calculated by

$$P_t^i(\mathbf{r}) = \frac{\exp(\alpha + \beta_0 r_i + \beta y_i)}{\sum_{j=1}^n \exp(\alpha + \beta_0 r_j + \beta y_j) + 1},$$

where β, β_0 and α are the parameters in MNL model and y_i is a vector of the attributes of product i. The probability that there is no purchase time period t is calculated by

$$P_t^0(\mathbf{r}) = \frac{1}{\sum_{j=1}^n \exp(\alpha + \beta_0 r_j + \beta y_j) + 1}.$$

To formulate a Markov decision process (MDP) for the problem we consider, the components in MDP are shown as follow:

- The state $\mathbf{s} = (s^1, \cdots, s^n)$ is the number of unsold units for each product.
- The action space is given by the allowable prices range $\mathbb{P}_i = (r_i^{min}, r_i^{max})$. Each action is the price vector $\mathbf{r}$ offered to the customers.
- The transition function is the probability that the customer chooses the product which leads the inventory from state s to state s'.
- The reward function is the price of the product which is sold in this time period t.
- In our problem, we set the discount factor to 1.

Let ϵ^i denote one unit of product i. For a given remaining time period t and inventory $\mathbf{s}$, the maximum expected revenue $v_t(\mathbf{s})$ can be calculated by

$$v_t(\mathbf{s}) = \max_{\mathbf{r}} \left\{ \lambda \sum_{i=1}^{n} P_t^i(\mathbf{r}) \left[r_i + v_{t-1}(\mathbf{s} - \epsilon^i) \right] + \left[1 - \lambda + \lambda P_t^0(\mathbf{r}) \right] v_{t-1}(\mathbf{s}) \right\} \forall t, \forall \mathbf{s}, \quad (4.1)$$

$$v_0(\mathbf{s}) = 0.$$

The first term in the equation denotes the revenue for the sale in period t and the second term is the revenue when there is a no-purchase arrival or there is no arrival in period t. This is defined as $\Delta_i v(\mathbf{s}) = v(\mathbf{s}) - v(\mathbf{s} - \epsilon^i)$. We can then rewrite Equation 4.1 as

$$v_t(\mathbf{s}) = \max_{\mathbf{r}} \left\{ \lambda \sum_{i=1}^{n} P_t^i(\mathbf{r}) \left[r_i - \Delta_i v_{t-1}(\mathbf{s}) \right] \right\} + v_{t-1}(\mathbf{s}) \forall t, \forall s. \tag{4.2}$$

Dong et al. (2009) introduce a backward induction algorithm which solves Equation 4.1 with an exact optimal solution. The computational complexity of the algorithm is $O(\|\mathbf{s}\|^n t)$, where $\|\mathbf{s}\| = \max\{s^i : i = 1, \cdots, n\}$. The computational complexity shows a limitation of MDP called the curse of dimensionality. The computation time can increase dramatically and make it impossible to solve in a practical time. Therefore, an approximation method should be applied to solve the problem. A detailed discussion about approximate dynamic pricing can be found in Busoniu et al. (2010).

4.2 Approximation Method in Zhang and Cooper (2009)

To solve the value function given in Equation 4.2, Zhang and Cooper (2009) provide an approximation method. They decompose the original problem with n products into n one-product problems and provide upper bounds and lower bounds of the value functions for the one-product problem; they then combine the upper bounds and lower bounds with a weight parameter to give an approximation of the original value function. The details of the upper bounds and lower bounds are presented in this section. We build on their approximations in the method we have developed in Section 4.3.

Given t remaining time periods, we denote the lower bound of the value function for product i with s unsold units by $\underline{v}_t^i(s)$ and the upper bound of the value function by $\overline{v}_t^i(s)$. Zhang and Cooper (2009) calculate the lower bounds and upper bounds of the value function by

$$\underline{v}_t^i(s) = \max_{r_i} \left\{ \lambda \underline{P}_t^i(r_i) \left[r_i - \Delta \underline{v}_{t-1}^i(s) \right] \right\} + \underline{v}_{t-1}^i(s) \forall t, \forall s, \tag{4.3}$$

$$\underline{v}_0^i(s) = 0$$

and

$$\overline{v}_t^i(s) = \max_{r_i} \left\{ \lambda \overline{P}_t^i(r_i) \left[r_i - \Delta \overline{v}_{t-1}^i(s) \right] \right\} + \overline{v}_{t-1}^i(s) \forall t, \forall s, \tag{4.4}$$

$$\overline{v}_0^i(s) = 0,$$

where $\underline{P}_t^i(\mathbf{r})$ and $\overline{P}_t^i(\mathbf{r})$ denote the lower and upper bounds of purchase probability for product i respectively. For our problem, we model the choice preference with the MNL model, whereas Zhang and Cooper (2009) describe their algorithm for a general choice model.

We here write down the bounds of the purchase probability for an MNL model using the equations (13) and (14) in Zhang and Cooper (2009), which give bounds for a general choice model.

$$\underline{P}_t^i(r_i) = \frac{\exp(\alpha + \beta_0 r_i + \beta y_i)}{\exp(\alpha + \beta_0 r_i + \beta y_i) + \sum_{j \neq i} \exp(\alpha + \beta_0 r_j^{min} + \beta y_j) + 1}. \tag{4.5}$$

$$\overline{P}_t^i(r_i) = \frac{\exp(\alpha + \beta_0 r_i + \beta y_i)}{\exp(\alpha + \beta_0 r_i + \beta y_i) + \sum_{j \neq i} \exp(\alpha + \beta_0 r_j^{max} + \beta y_j) + 1}. \tag{4.6}$$

To approximate the value function of the original problem, a parameter $\theta \in [0, 1]$ is used to combine the upper and lower bounds. The approximation of the value function given in Equation 4.2 and the approximation of the purchase probability are then presented as

$$\tilde{v}_t^i(s) = \sum_{t=1}^{T} \left[\theta \overline{v}_t^i(s) + (1-\theta)\underline{v}_t^i(s)\right], \tag{4.7}$$

$$\tilde{P}_t^i(r_i) = \theta \overline{P}_t^i(r_i) + (1-\theta)\underline{P}_t^i(r_i). \tag{4.8}$$

The optimal price vector is calculated by solving

$$\sum_{i=1}^{n} \max_{r_i} \left\{\lambda \tilde{P}_t^i(r_i) \left[r_i - \Delta \tilde{v}_{t-1}^i(s)\right]\right\}. \tag{4.9}$$

Different values of θ will result in different price vectors. Zhang and Cooper (2009) deal with this by changing the value of θ and finding corresponding price vectors. They then use simulation to determine the quality of each price vector and use this to determine the best θ. The corresponding price vector is recorded as the optimal price strategy. This approach provided in Zhang and Cooper (2009) is summarised below as Algorithm 5.

Algorithm 5 Approximation method from Zhang and Cooper (2009)

Set $\pi^* = 0$ and $v^* = 0$.

for θ move from $\underline{\theta}$ to $\overline{\theta}$ with step size δ **do**

Calculate $\tilde{v}_t$ and $\tilde{P}_t^i$ with Equation 4.7 and Equation 4.8 and use the values in Equation 4.9. The price vector which maximises Equation 4.9 is recorded as π_θ.

Offer the price vector π_θ in a simulation model for l replications. Record the expected total revenue $\hat{v}$

if $\hat{v} > v^*$ **then**

$v^* = \hat{v}$ and $\pi^* = \pi^\theta$

end if

end for

The approximation method in Zhang and Cooper (2009) avoids the "curse of dimensionality" problem. However, the computation time of the approximation method can be impractical with a large number of time periods and a high inventory level. For example, if we need to solve a problem with t time periods and inventory level s for product i, the lower bound $\underline{v}_t^i(s)$ needs to be calculated. We need to find the lower bound $\underline{v}_t^i(s)$ using Equation 4.3 and $\Delta\underline{v}_{t-1}^i(s) = \underline{v}_{t-1}^i(s) - \underline{v}_{t-1}^i(s-1)$; therefore the values of of $\underline{v}_{t-1}^i(s)$ and $\underline{v}_{t-1}^i(s-1)$ are needed. This procedure is the same when we try to find the values of $\underline{v}_{t-1}^i(s)$ and $\underline{v}_{t-1}^i(s-1)$. Figure 4.1 shows the values of bounds that need to be calculated for time period t and inventory level s.

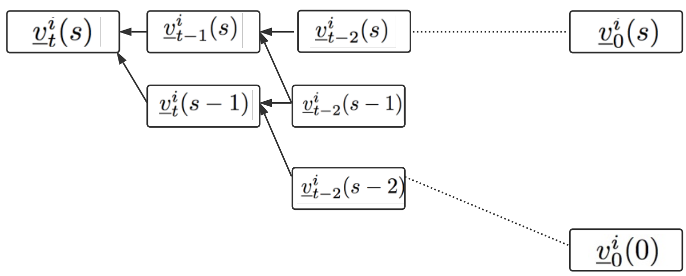


Figure 4.1: The values of bounds need to be calculated for time period t and inventory level s

The numerical study in Zhang and Cooper (2009) tests an example with 1000 time periods, but they do not mention the inventory level. For a problem with high demand, this increases the computation time significantly. In the results section, we measure the computation time for Zhang and Cooper's approximation with different numbers of time periods and inventory levels. This shows that the computation time can be prohibitively long in practical examples.

4.3 New approximation method

To reduce the running time of the approximation method, we propose a new approximation method and term it the 'DVA' method. We term the approximation method, which is proposed in Zhang and Cooper (2009), 'VA' method. The idea of the DVA method is to regard the bounds in the VA method as value functions of a single-product dynamic pricing problem. Therefore, we can propose bounds of the value functions. The lower bound and upper bound in the DVA method can avoid the calculation shown in Figure 4.1 and reduce the running time significantly. The new bounds can be combined in the same way as in the VA method, and thus approximate the original value function. The basic idea of the DVA method is illustrated in Figure 4.2. We describe how to construct the bounds below.

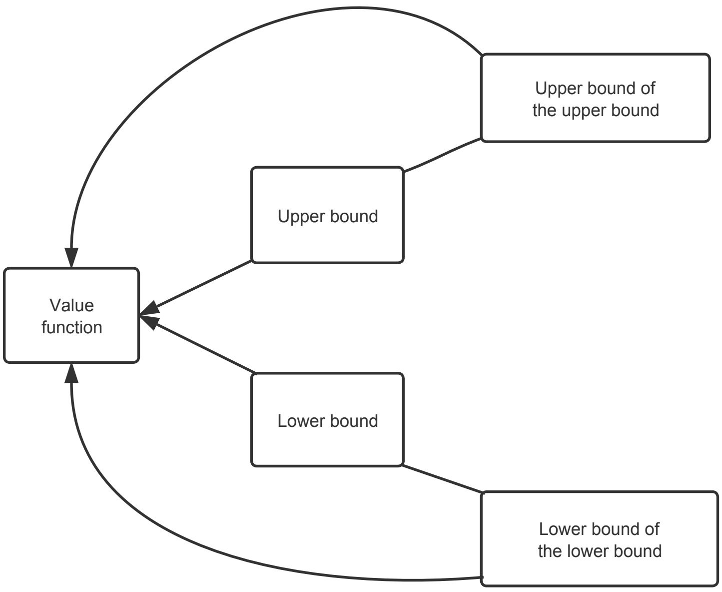


Figure 4.2: The structure of our approximation method

4.3.1 Lower Bounds of the Value function

Let $\underline{v}^i_L(s,t)$ denote the lower bound of the lower bound in the VA method with inventory s at time t. We treat the lower bound in the VA method as a value function of a single-product problem with arrival rate $\lambda \underline{P}^i_t$. Sen (2013) proves that

$$s\underline{v}^i(1, t/s) \leqslant \underline{v}^i(s,t). \tag{4.10}$$

Sen divides the whole selling period t into s periods. The length of each period is t/s. The $\underline{v}^i(1, t/s)$ is the expected revenue for selling one product in a small period with length t/s. Therefore, $s\underline{v}^i(1, t/s)$ is the expected revenue for the problem of selling one item during each small period and ignoring any unsold items from previous periods. We use $\underline{v}^i(s,t)$ to denote the expected revenue for selling s products. There is a positive probability that some items may be unsold in small periods; therefore, the inequality in Equation 4.10 holds.

In Sen (2013), the author assumes a continuous time period, while we use discrete time periods. In the discrete time case, we would use $\lfloor t/s \rfloor$ to denote the time period corresponding to the continuous time t/s, and it is possible to show that

$$\underline{v}^i(1, \lfloor t/s \rfloor) \leqslant \underline{v}^i(1, t/s), \tag{4.11}$$

where $\underline{v}^i(1, \lfloor t/s \rfloor)$ is the lower bound of the revenue for selling one unit of product in a time period with length $\lfloor t/s \rfloor$ and $\underline{v}^i(1, t/s)$ is the lower bound of the revenue for selling one unit of product in a time period with length t/s. There is a non-negative probability that the product may be unsold in time period $\lfloor t/s \rfloor$ and sold in time period $t/s - \lfloor t/s \rfloor$; therefore, the inequality 4.11 holds.

Using this result, we can construct the lower bounds in DVA method as

$$\underline{v}^i_L(s,t) = s\underline{v}^i(1, \lfloor t/s \rfloor). \tag{4.12}$$

With these equations, we do not need to calculate the value of the lower bound shown in Figure 4.1 in the VA method for each inventory level. Only the bounds with inventory level equal to 1 need to be computed. These values can then be used to find the lower bounds in the DVA method fast and easily.

4.3.2 Upper Bound of the Value Function

Let $\overline{v}_U^i(s,t)$ denotes the upper bound of the upper bound in the VA method with inventory s at time t. We treat the upper bound in the VA method as a value function of a single-product problem with arrival rate $\lambda\overline{P}_t^i$.

We define s/t as the run-out rate and the corresponding price is defined as the run-out price. The run-out price can be find with the reverse function of Equation 4.6, which can be shown as

$$r_{runout} = \frac{\log \frac{(s/t\lambda)(\sum_{j\neq i}\exp(\alpha+\beta_0 r_j^{max}+\beta y_j)+1)}{1-s/t\lambda} - \alpha - \beta y_i}{\beta_0}.$$

Theorem 2 in Gallego and van Ryzin (1994) provides an upper bound for a single-product problem. For the problem we solved, $\overline{v}_U^i(s,t)$ can be obtained with following steps.

First, we record the price $\hat{r}_i$, which maximises the equation $\lambda\overline{P}_t^i(r_i)r_i$.

If $s/t < \lambda\overline{P}_t^i(\hat{r}_i)$, we have

$$\overline{v}_U^i(s,t) = s * r_{runout}. \tag{4.13}$$

If $s/t \geqslant \lambda\overline{P}_t^i(\hat{r}_i)$, we have

$$\overline{v}_U^i(s,t) = \lambda\overline{P}_t^i(\hat{r}_i)\hat{r}_i t. \tag{4.14}$$

Theorem 2 in Gallego and van Ryzin (1994) has a constraint that $\lambda\overline{P}^{*i}(r_i)$ is a regular demand function satisfying three assumptions:

1. $\lambda\overline{P}^{*i}(r_i)$ has inverse functions.
2. The revenue function $rev = \lambda\overline{P}^{*i}(r_i)r_i$ is continuous, bounded and concave.
3. There exists a null price which leads the demand to zero.

We use Equation 4.13 and 4.14 as the upper bound in the DVA method, which can be calculated directly for any value of t and s. Unlike the bounds in the VA method, the computation time of the upper bound in the DVA method will not be affected by the value of t and s.

4.3.3 Approximation of the Bounds

Given

$$\underline{v}^i_L(s,t) \leqslant \underline{v}^i(s,t),$$

$$\underline{v}^i(s,t) \leqslant v^i(s,t) \leqslant \overline{v}^i(s,t),$$

and

$$\overline{v}^i(s,t) \leqslant \overline{v}^i_U(s,t),$$

we have

$$\underline{v}^i_L(s,t) \leqslant v^i(s,t) \leqslant \overline{v}^i_U(s,t).$$

We can approximate $v^i(s,t)$ by finding a proper value of parameter θ in the following equation

$$v^i_{app}(s,t) := \theta \overline{v}^i_U(s,t) + (1-\theta)\underline{v}^i_L(s,t). \tag{4.15}$$

Following a similar approximation method in Zhang and Cooper (2009), we calculate $\tilde{P}^i_t(r_i)$ with

$$\tilde{P}^i_t(r_i) = \theta \overline{P}^i_t(r_i) + (1-\theta)\underline{P}^i_t(r_i)$$

and find the optimal price vector by solving

$$\sum_{i=1}^{n} \max_{r_i} \left\{ \lambda \tilde{P}^i_t(r_i) \left[r_i - \Delta v^i_{app}(s,t-1) \right] \right\} \tag{4.16}$$

The price which maximises the function in Equation 4.16 is recorded as the price policy π^θ. Then, we construct a simulation of a selling system. We run the simulation l times and record the average revenue. The θ which returns the highest average revenue is chosen as the parameter to combine the lower and upper bounds. The DVA method is described in Algorithm 6.

Algorithm 6 DVA method

Set $\pi^* = 0$ and $v^* = 0$.

for θ move from $\underline{\theta}$ to $\overline{\theta}$ with step size δ **do**

Calculate $v^i_{app}(s,t)$ and $\overline{P}^i_t$ with

$$v^i_{app}(s,t) := \theta \overline{v}^i_U(s,t) + (1-\theta)\underline{v}^i_L(s,t).$$

$$\tilde{P}^i_t(r_i) = \theta \overline{P}^i_t(r_i) + (1-\theta)\underline{P}^i_t(r_i)$$

Find price vector π_θ by solving

$$\sum_{i=1}^{n} \max_{r_i} \left\{ \lambda \tilde{P}^i_t(r_i) \left[r_i - \Delta v^i_{app}(s,t-1) \right] \right\}$$

Offer the price vector π_θ in a simulation model for l replications. Record the expected total revenue $\hat{v}$

if $\hat{v} > v^*$ **then**

$v^* = \hat{v}$ and $\pi^* = \pi^\theta$

end if

end for

4.4 Results

Here, we consider two examples to test the performances of VA and DVA. The first is a small example from Dong et al. (2009). With this small example, we can obtain the exact solution of optimal price and total revenue in practical time and compare the performance of the approximation methods with the exact solution. The second example has the same setting as the simulated data in Chapter 3 have. The initial inventory level is set to 4000. With this example, we compare the computation time and revenue between the VA and DVA methods.

4.4.1 Three-product example in Dong et al. (2009)

The example in Dong et al. (2009) considers a three-product problem. The customers arrive in the selling system following a Poisson process with an arrival rate of $\lambda = 0.1$. The utilities of products are given by

$$U_1 = 11.75 - r_1,$$

$$U_2 = 9 - r_2,$$

and

$$U_3 = 6.25 - r_3,$$

where r_i is the price of product i.

Figures 4.3, 4.4 and 4.5 show the optimal prices for products with inventory level (2,2,2). We note that the optimal prices from approximation methods and optimal prices from exact methods have a different pattern when the number of remaining time periods is small. Because both of the approximation methods decompose the n-product problem into several one-product problems. We consider an extreme case whereby only one time period is left. For a n-substitutable-products problem, the seller should charge them with same price, because the products are substitutable. The price should be charged depending on the quality of the product.

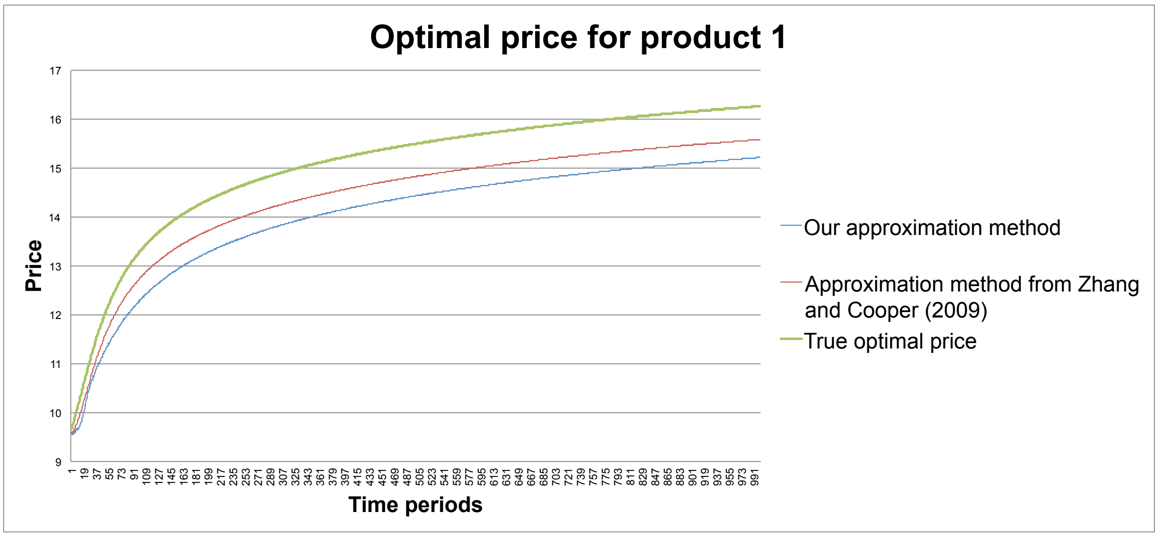


Figure 4.3: Optimal price for product 1

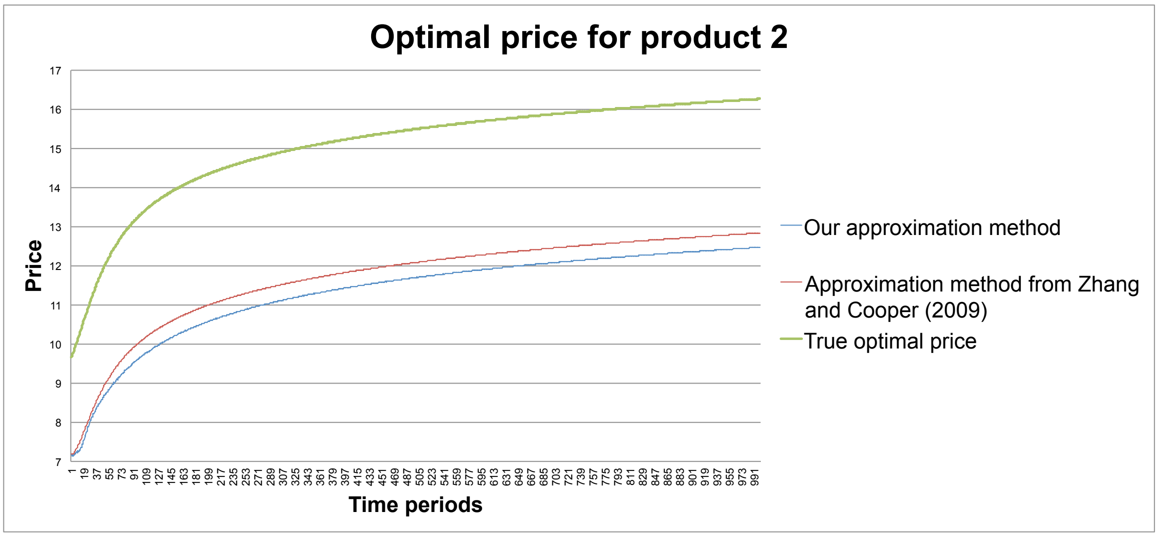


Figure 4.4: Optimal price for product 2

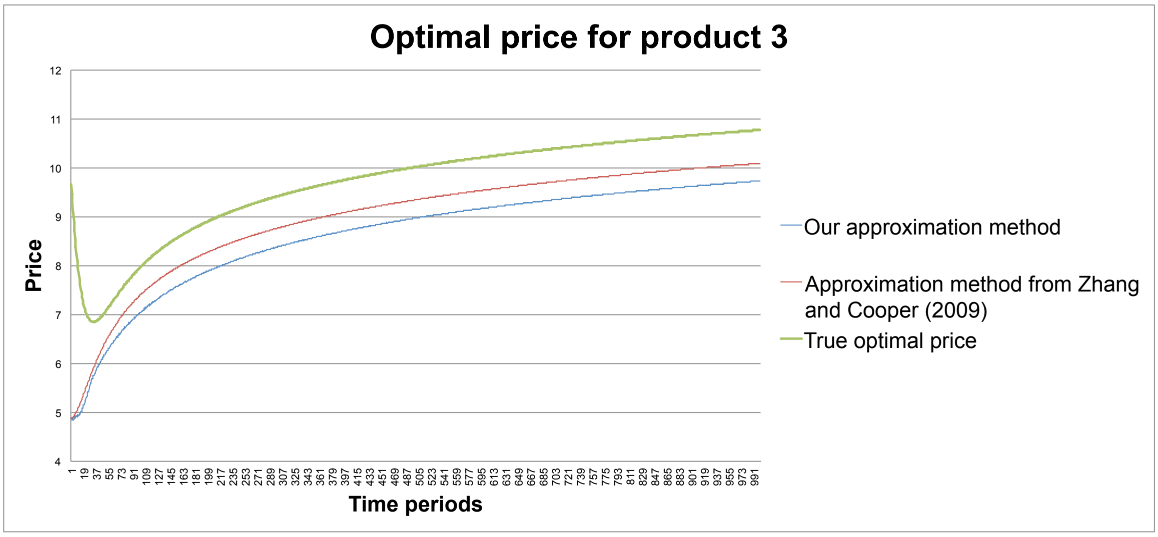


Figure 4.5: Optimal price for product 3

Figure 4.6 shows the optimal revenue under different methods. Both the approximation methods provide near-optimal solutions.

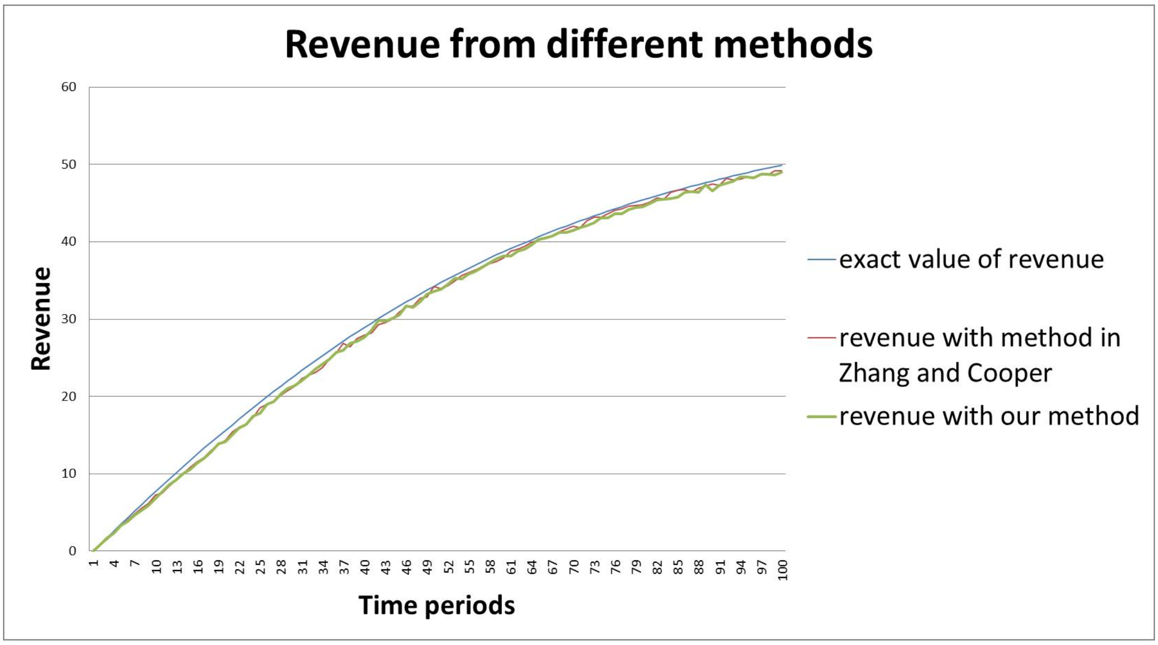


Figure 4.6: Revenues from DVA, VA and exact solution

4.4.2 Example with Simulated Data

In this section, we test the performance of the DVA method and the VA method with the simulated data presented in Chapter 3. We have 14 products and the initial capacity is set to be 10 for each product. The parameters in customer choice model and arrival rate are known in advance. The number of time periods in the selling horizon is range from 100 to 1000. We run the test for 100 times and take the average of the revenue. In addition, we run 100 times of simulations to find the appropriate value of θ. Table **??** shows the revenue from the DVA and the VA methods and the percentage bias between these two methods. Fig 4.7 plots the percentage bias between the DVA and VA methods. The percentage bias is calculated with

$$Percentage\ bias = \frac{Revenue\ from\ the\ DVA\ method - Revenue\ from\ the\ VA\ method}{Revenue\ from\ the\ VA\ method} * 100\%.$$

From the Fig **??** and Table **??**, we can find that the revenue from the DVA and VA methods are pretty close. Most of the time the VA method outperform the DVA method but there is exist the situation that the DVA method returns a better average revenue than the VA method. In the test, the percentage bias falls in 0.7% range when the number of time periods is larger or equal to 300. Fig 4.7 shows the percentage bias between the DVA and VA methods

	DVA	VA	Percentage Bias
100	29.06	30.00	-3.14
200	64.29	67.00	-4.05
300	106.00	106.69	-0.65
400	143.68	143.89	-0.14
500	179.19	179.29	-0.05
600	213.24	213.21	0.02
700	244.78	244.94	-0.06
800	273.46	273.58	-0.04
900	298.30	298.90	-0.20
1000	320.20	321.16	-0.30

Table 4.1: Revenue from the DVA and the VA methods and the percentage bias between these two methods

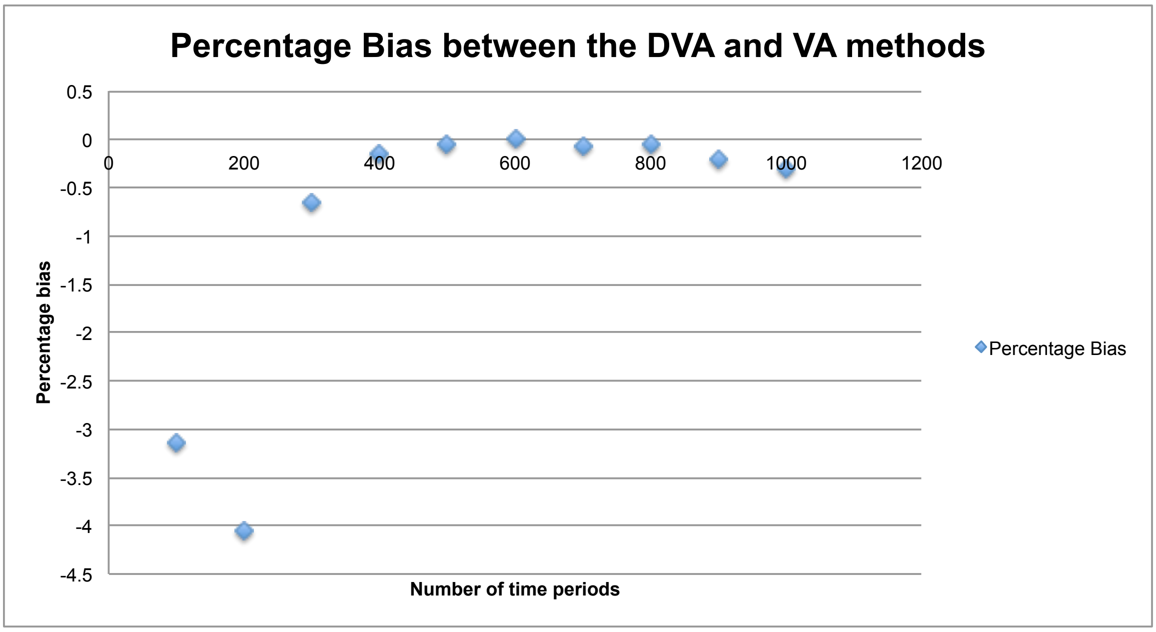


Figure 4.7: Percentage Bias between the DVA and VA methods

4.4.3 Running Time

The following table compares the running time (implemented in MATLAB on a OSX 10.9, 2.4 GHz) of the DVA method and the VA method. In the first test, we consider a

14-product problem with inventory level 100. Different time periods are used in the test. We can find that the DVA method reduces the running time by 89% for an example with 1000 time period and an inventory level 100.

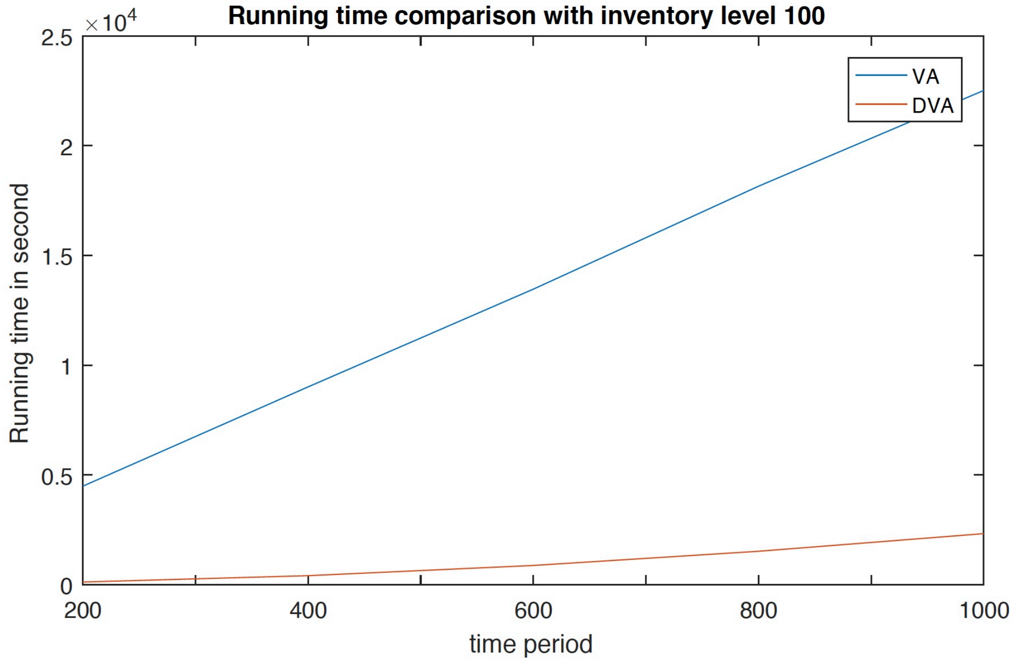


Figure 4.8: Running time comparison with inventory level 100

Figure 4.9 shows the running time for the problem with different inventory levels. In the test, we assume the problem has 100 time periods.

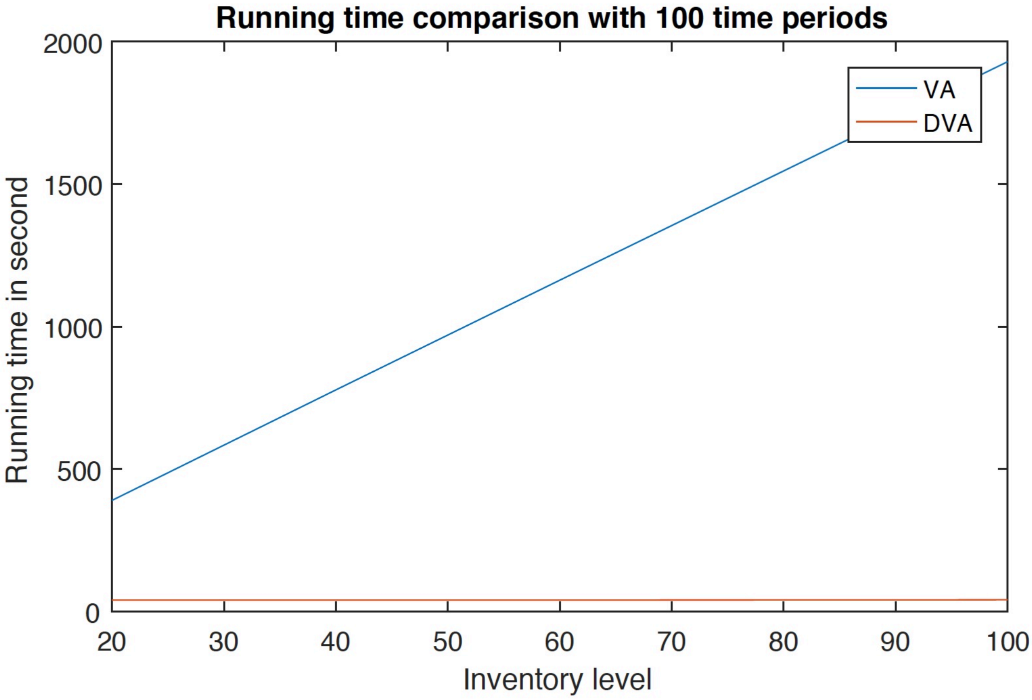


Figure 4.9: Running time under different inventory levels with time period t=10

4.5 Conclusion

Our contribution in this chapter is proposing an approximation method for the choice-based dynamic pricing problem which was shown in our numerical tests to reduce the computational time of the existing approximation method put forward by Zhang and Cooper (2009) by 89% (with 1000 time periods and 100 inventory level). The proposed approximation method can benefit firms that have to solve large-scale choice-based RM problems or those that need to update their pricing policy in a short time.

The existing approximation method decomposes the original n-product dynamic pricing problem into several one-product dynamic pricing problems and provides the upper bounds and lower bounds for the value function of the one-product problem to solve the "curse of dimensionality". For every remaining time period T, the method needs to calculate the value function from T-1 to 1. The computational time can be very large with a large number of time periods. We reduce the computational time by further approximating the upper bounds and lower bounds. Only two values of bounds are needed for any value of time period T.

Firms solving large revenue optimisation problems with high levels of inventory and demand need to be able to optimise prices quickly. These optimisation methods need to be run every night in many cases. Speed of computation is important. Our method speeds it up but still performs well. The approximation method can be applied in transportation area like railway companies.

A limitation of the study is the assumption that each customer purchases at most one product. However, in practice, some customers may choose to purchase more than one product at a time. Incorporating this situation may be another direction of future work. In this study, we construct the lower bound of the value function for a discrete time model from a lower bound for a continuous time model. If we could find a tighter lower bound in the same computation time, we could improve the performance of the approximation method.

Chapter 5

Multi-armed Bandit and Pricing

In this chapter, we consider a dynamic pricing problem with demand and substitution behaviour uncertainty. In the problem, a seller offers a set of substitutable products to a sequence of customers who choose products based on their characteristics. The choice behaviour is homogeneous and can be modelled with a MNL model. The parameters in the MNL model and the customer arrival rate are unknown to the seller. He or she needs to learn the information with a price experiment and find a good pricing policy to maximise the long-term profit. There is a trade-off between learning the information and earning revenue. If the seller spends too much time finding out the information about demand and substitutable behaviour (exploration), he or she will lose the opportunity to earn short-term profit. If the seller spends too little time exploring and offers prices based on poor estimates of demand and substitution behaviour, he or she will lose the opportunity to improve the long-term profit. This is the dilemma of exploration and exploitation.

The trade-off between exploration and exploitation can be solved with pulling policies in a Multi-armed Bandit (MAB) problem. (We describe the underlying ideas of MAB policies in more detail in Chapter 2). In essence, we are solving the same problem that a player faces when playing an array of one-arm bandit machines (or slot machines in a casino); that of deciding which arm to pull in each step. A reward is obtained from an unknown distribution of the arm that is pulled and the aim of the player is to maximise the sum of rewards. This type of problem can be found in many situations, such as clinical trials or online advertising.

For the problem we considered here, the arms are the price vectors we offer to the customers. The MAB algorithm will choose whether to exploit the current estimated the best price vector, or to explore. When exploiting, we use the optimal price vector obtained from solving a dynamic pricing problem (as described in Chapter 4) using the current best estimates of the arrival rate and MNL parameters. When exploring, we use a random price vector that has been generated from a given uniform distribution, as described in Section 5.2. We split the whole selling period into a set of constant-price periods, which may contain multiple time periods. The price vector on offer is only changed when we move to the next constant-price period. After each constant-price period, we collect the sales data and update our belief of demand and substitution information.

In contrast, many studies that consider the dynamic pricing problem with demand uncertainty follow a learning and then earning pattern (Besbes and Zeevi, 2009). Broder and Rusmevichientong (2012) propose a MLE-CYCLE policy that performs the learning and then earning phase in a cycle. Schwartz et al. (2016) apply an Upper Confidence Bound policy to balance the learning and earning phase.

In this chapter, we propose policies in MAB similar to the class of randomised probability matching policies, which choose arms based on a probability that the arms are optimal. A detailed discussion of randomised probability matching policies can be found in Scott (2010). The author also provides numerical comparisons with other policies in MAB. Agrawal and Goyal (2012) provide an analysis of Thompson sampling that is also in the class of randomised probability matching policies.

In each constant-price period, we generate a random price vector from uniform distribution and use this random price vector with a probability, which is a function of the distance between this price vector and the current optimal price vector. Otherwise, the optimal price vector based on the current estimates is offered, which means we perform a exploitation in this constant-price period.

We measure the performance of a policy in terms of regret, which is defined as the expected difference between the revenue obtained using the policy and the revenue of a clairvoyant seller who knows full information and offers the optimal price vector all the time. We also use the percentage of revenue loss to measure the performance of pricing policy, which is defined as the regret divided by the revenue of a clairvoyant seller.

Contributions In this chapter, we propose two pricing policies—RP policy and RP-a policy—which solve the trade-off between estimation and choice-based dynamic pricing. The RP-policy is straightforward and easy to implement with any existing estimation methods and choice-based dynamic pricing policy. RP-a adjusts the RP-policy to incorporate the uncertainty of the estimation. It enables the user to adjust the ratio of exploration and exploitation automatically. Furthermore, we provide numerical results to show the performance of RP and RP-a policies and a comparison with three alternative policies.

In Section 5.1, we describe the problem we solved. The proposed pricing policies are shown in Section 5.2. The numerical results are provided in Section 5.3 and a conclusion is presented in Section 5.4.

5.1 Problem description

In this chapter, we consider a problem of a seller offering n substitutable products to customers over T time periods with a pricing policy. The price vector is fixed during a constant-price period, e.g., one day. The customers arrive following a Poisson process that has an arrival rate of λ. Each of the customers chooses one product based on a certain choice behaviour and we assume the customers have the same choice behaviour that can be modelled with a MNL model. Customers who arrive and purchase a product are recorded in the sales data and the arrivals without a purchase are assumed to be unobservable. In addition, when a customer's first choice is sold out, he/she may choose to leave or purchase another substitutable product. We assume we have an exogenous estimate of the aggregate market share under a certain price vector. When a new price vector is offered to the customer, we assume the other companies will not respond to the price changes and a new market share can be calculated based on the new offered price vector. With this information, the estimation of demand and substitution behaviour can be estimated with methods provided in Chapter 3. In this chapter, we apply the estimation method that we have developed and described in Section 3.2 to estimate the parameters.

Given the estimation of demand and substitution behaviour, we can use the choice-based dynamic pricing algorithm, as described in Chapter 4, to calculate the current optimal price vector. The problem we consider in this chapter can be solved with the structure which is shown in Figure 5.1.

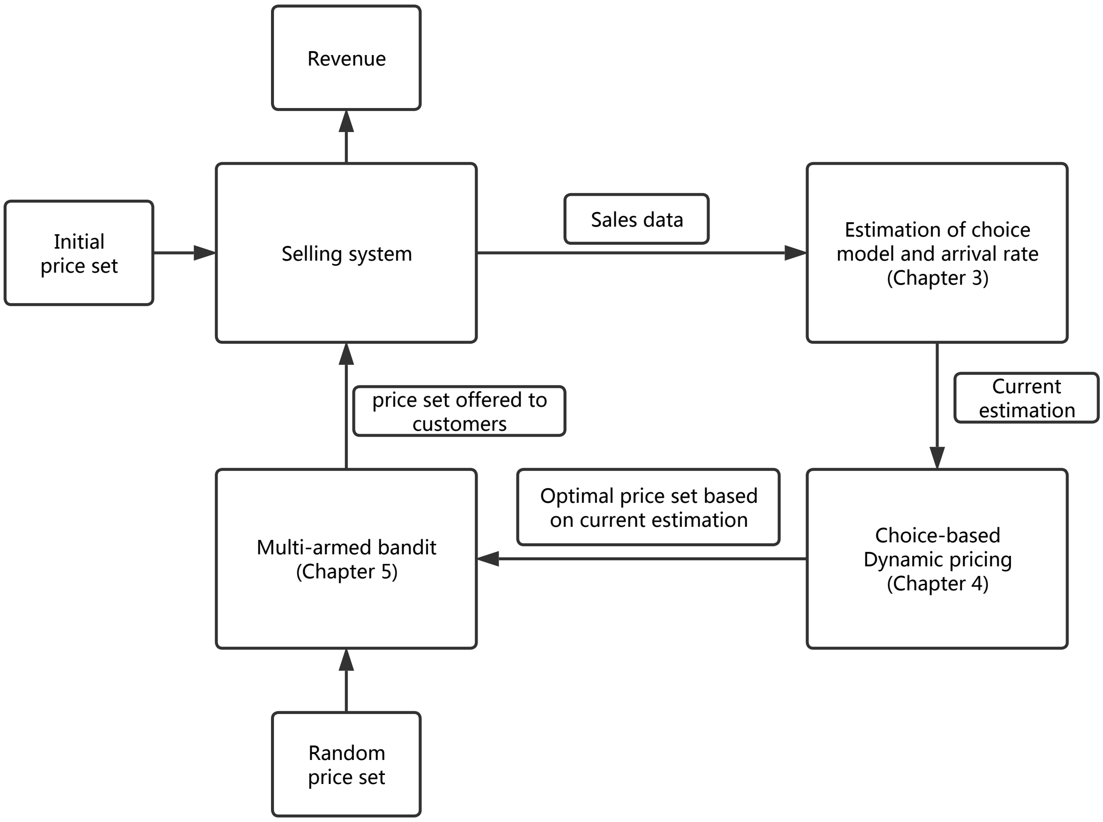


Figure 5.1: The structure of the problem

A straightforward pricing policy would be to apply the optimal price vector based on current estimates of arrival rate and MNL parameters using the sales data from this pricing policy to update the estimates of parameters. The current optimal price vector may not be the actual optimal price vector when the current parameter estimates are of a poor quality. This pricing policy is described as passive learning, myopic pricing or certainty equivalent pricing. In Chapter 3, we apply the two-step algorithm with different price vectors and show that the random price vector provides a better estimation than the fixed price vector does. The test provides evidence that price diversity benefits the quality of estimation that could improve the long-term revenue. Therefore, we need a price policy that can explore the feasible price area.

Keskin and Zeevi (2014) propose an orthogonal pricing policy. For problems with n multiple products, the policy constructs n price vectors based on an orthogonal basis for any given block of n time periods. The prices for each of the products on offer are changed one at a time from the current optimal price vector. As more information is collected, the length of the search distance shrinks and the price vectors in the exploration phase will approach the true optimal price vector. den Boer and Zwart (2013) propose a controlled variance pricing policy that is a certainty equivalent pricing policy with a taboo interval around the price vector that has been offered. We propose a random pricing policy which is more straightforward, easy to implement, and learns about the feasible price area more quickly.

5.2 Pricing policies

5.2.1 RP policy

We propose a pricing policy that samples a random price vector from uniform distributions in each constant-price period. We calculate a probability of selection for the random price vector, which is a function of the distance between this price vector and the current optimal price vector. Our pricing policy is to offer the random price vector with a probability equal to this probability of the selection and hence perform an exploration. Otherwise, the current optimal price vector will be applied and we run an exploitation phase. The uniform distributions ensure that the price vectors sampled are spread evenly in the available price range. If the random price vector is far from the current optimal price vector, the chance of offering it is small. If the distance is small, it is more likely that we will select the random price vector. This probability is effectively used to decide whether to perform an exploration phase or an exploitation phase and maintains the balance between exploration and exploitation. The name of the RP policy comes from the **random** price vector used and matching **probability**. A detailed description of the pricing policy is provided below.

The price vector offered to the customers is $\mathbf{r} = \{r_1, \cdots, r_n\}$. We denote the feasible price interval for product $i \in \{1, \cdots, n\}$ by $\mathbb{P}_i = (r_i^{min}, r_i^{max})$. After that a pricing policy is applied and generate a new price vector to the customer. In this chapter, we propose a pulling strategy in MAB to decide which price vector is offered to the customers. Let $\mathbf{r}_{opt}^t$ denote the optimal price vector based on the estimation in constant-price period t and $\mathbf{r}_{ran}^t$ denote the random price vector sampled in constant-price period t. We measure the distance between two price vectors $\mathbf{r}$ and $\mathbf{r}^*$ with $d(\mathbf{r}, \mathbf{r}^*)$, which is calculated by

$$d(\mathbf{r}, \mathbf{r}^*) = ||\mathbf{r} - \mathbf{r}^*||,$$

where $|| \cdot ||$ is the Euclidean norm. We denote the "maximum distance" of the feasible price area as φ, which is calculated by

$$\varphi = ||(\mathbf{r}^{max} - \mathbf{r}^{min})||.$$

The probability of choosing an exploration phase and offering the sampled random price vector $\mathbf{r}_{ran}$ to customers is set to be

$$\omega = \max\{0, 1 - \frac{d(\mathbf{r}_{ran}, \mathbf{r}_{opt})\tau}{\varphi}\},$$

where τ is a positive parameter that can be tuned by the user. With this parameter, the user can adjust the probability of choosing a random price vector. For users who are happy to spend more time in exploration, they can set τ to a little number. Otherwise, they can set τ to a larger number. We term this pricing policy, which samples a random price vector and matches it with a probability, the RP policy, and describe it in Algorithm 7.

Algorithm 7 Pricing policy RP

In constant-price period $t = 1$, generate a random price vector $\mathbf{r}_{ran}^1$

Go to **Exploration Phase**:

Offer the products with random price vector $\mathbf{r}_{ran}^1$ and record the sales data.

Given the sales data, find the current estimates of parameters with the estimation method.

Find the current optimal price vector $\mathbf{r}_{opt}^2$ with a choice-base dynamic pricing model based on the current estimate.

for $t = 2, 3, \cdots$ **do**

Generate a random price vector $\mathbf{r}_{ran}^t$

Calculate the probability corresponding to the random price vector with

$$\omega^t = \max\{0, 1 - \frac{d(\mathbf{r}_{ran}^t, \mathbf{r}_{opt}^t)\tau}{\varphi}\}. \tag{5.1}$$

Generate a uniformly distributed random number ρ in the interval (0,1).

if $\rho < \omega^t$ **then**,

Go to **Exploration phase**:

Offer the products with random price vector $\mathbf{r}_{ran}^t$ and record the sales data.

Given the sales data, find the current estimates of parameters with the estimation method.

Find the current optimal price vector $\mathbf{r}_{opt}^{t+1}$ with a choice-base dynamic pricing model based on the current estimate.

else

Go to **Exploitation Phase**:

Offer the products with current optimal price vector $\mathbf{r}_{opt}^t$ and record the sales data.

Given the sales data, find the current estimates of parameters with the estimation method.

Find the current optimal price vector $\mathbf{r}_{opt}^{t+1}$ with a choice-base dynamic pricing model based on the current estimate.

end if

end for

5.2.2 RP-a policy

As more sales data are collected, we will have more confidence about the estimation of demand and substitution behaviour. We should spend more time in exploitation than in exploration. Therefore, we adjust the probability with a function which measures the difference between two consecutive estimates. When the estimation becomes more stable, the chance of choosing an exploitation phase will be greater. We name the adjusted RP policy as RP-a policy.

Let $\delta^k = (\alpha^k, \beta_0^k, \beta^k, \lambda^k)$ denote the k^{th} estimates, and the difference between two consecutive estimates can be calculated as

$$||\delta^k - \delta^{k-1}|| = ||(\alpha^k, \beta_0^k, \beta^k, \lambda^k) - (\alpha^{k-1}, \beta_0^{k-1}, \beta^{k-1}, \lambda^{k-1})||.$$

The probability of choosing an exploration phase and offering the sampled random price vector $\mathbf{r}_{ran}$ to customers is set to

$$\omega = \max\{0, 1 - \frac{d(\mathbf{r}_{ran}, \mathbf{r}_{opt})\tau}{\varphi||\delta^k - \delta^{k-1}||}\}. \tag{5.2}$$

We call this pricing policy RP-a, which has a structure similar to the RP policy. The only difference is replace Equation 5.1 with Equation 5.2.

5.3 Numerical Experiments

In this section, we test the performance of the RP policy by presenting the rate of regret. Then we compare our RP policy, RP-a policy and three alternative policies and measure the performance with percentage revenue loss.

The example we use here is similar to the simulated data used in Chapter 3 to test the performance of the two-step algorithm. The only difference is that we keep the price vector unchanged over the course of one day. If a product sells out we set its price to a null price and keep the prices of other products unchanged. We assume that the estimation and dynamic pricing algorithms are run overnight and generate the new optimal price vector based on the current estimation. The seller sells 14 substitutable products to customers. The maximum prices for the products are set to three and the minimum prices for the products are set to zero.

In Figure 5.2, we present the logarithm of the average regret of the RP policy versus $log(t)$ over five independent runs, where t denotes the time period. We can find from the figure that the results have a linear form and the slope of the line, which fits the logarithm of regret, is 0.41. This empirical result shows that the RP policy achieved $\Theta(T^{0.41})$ order of regret.

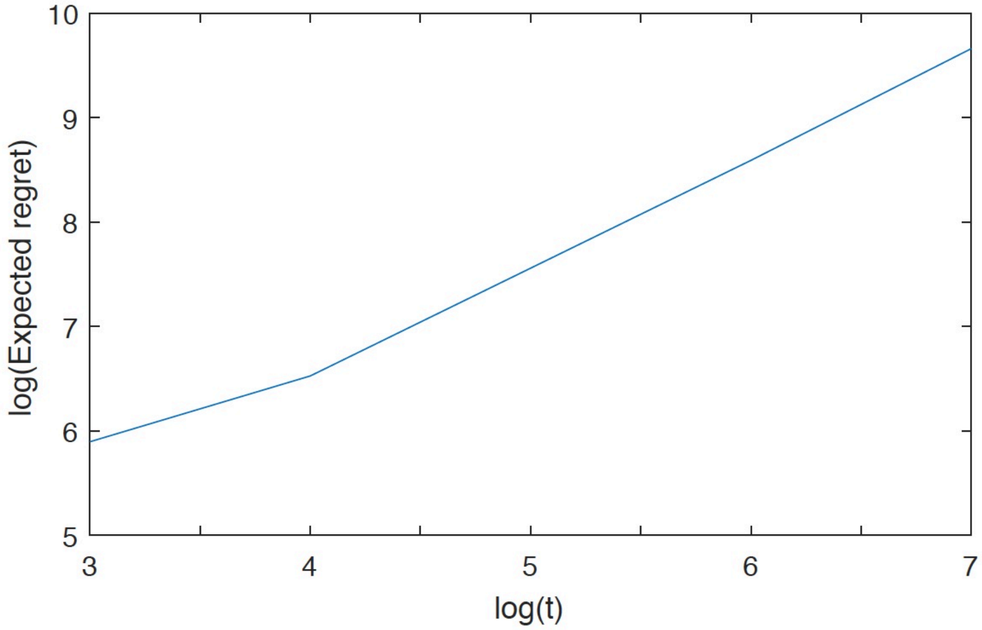


Figure 5.2: Average regret of RP policy based on five iterations

Then, we compare our policies with three other policies. We describe the policies below.

1. **ϵ-first Policy**

ϵ-first is a simple pulling policy which does the pure exploration first and then moves to the exploitation phase. It is applied in the studies that have a learning and then an earning structure. For a problem with T time periods, ϵ-first spends ϵT periods in the exploration phase and uses the remaining time periods to do the exploitation. The value of ϵ is determined by the user. For example, if the user decides to spend half of the whole period to explore the unknown parameters, he can set the $\epsilon = 0.5$.

For a T periods problem, the ϵ-first strategy for our problem can be described below.

Algorithm 8 ϵ-first policy

for $t = 1, 2, \cdots, \lfloor \epsilon T \rfloor$ **do**

Generate a random price vector $\mathbf{r}_{ran}^{t}$.

Offer the products with random price vector $\mathbf{r}_{ran}^{t}$ and record the sales data.

end for

Given the sales data, find the current estimates of parameters with the estimation method.

Find the current optimal price vector $\mathbf{r}_{opt}$ with a Choice-base dynamic pricing model based on the current estimates.

for $t = \lfloor \epsilon T \rfloor + 1, \cdots, T$ **do**

Offer the products with current optimal price vector $\mathbf{r}_{opt}$ and record the sales data.

end for

2. **CYCLE Policy**

The ϵ-first policy has a major shortcoming in that the sales data collected from the exploitation phase cannot be used in the exploration phase. Broder and Rusmevichientong (2012) propose a MLE-CYCLE policy to improve the total revenue by operating the exploration phase and exploitation phase in a cycle. The exploration price vectors are generated randomly from uniform distributions. To compare with other policies, we stop the CYCLE policy when the total time period reaches T.

Algorithm 9 CYCLE policy

Set $t = 0$.

Given K exploration price vectors $\mathbf{r}_1, \cdots, \mathbf{r}_K$,

while $t < T$ **do**

  for $c = 1, 2, \cdots$ **do**

    for $k = 1, 2, \cdots$,K **do**

      Offer the products with random price vector $\mathbf{r}_k$ and record the sales data.

      $t = t + 1$.

    end for

    Given the sales data, find the current estimates of parameters with the estimation method.

    Find the current optimal price vector $\mathbf{r}^c_{opt}$ with a Choice-base dynamic pricing model based on the current estimates.

    for $l = 1, \cdots$,c **do**

      Offer the products with current optimal price vector $\mathbf{r}^c_{opt}$ and record the sales data.

      $t = t + 1$.

    end for

  end for

end while

3. **Passive learning policy**

Passive learning policy is a policy without an exploration phase. It always offers the current optimal price vector to customers. The sales data from the exploitation phase are used to estimate the unknown parameters. The initial price vector is generated randomly from uniform distribution. To draw comparisons with other policies, we stop the passive learning policy when the total time period reach T. We describe the policy below.

Algorithm 10 Passive learning

In time period $t = 1$,

Offer the products with given price vector $\mathbf{r}$ and record the sales data.

Given the sales data, find the current estimates of parameters with the estimation method.

Find the current optimal price vector $\mathbf{r}_{opt}^{2}$ with a choice-base dynamic pricing model based on the current estimate.

for $t = 2, 3, \cdots, T$ **do**

Offer the products with current optimal price vector $\mathbf{r}_{opt}^{t}$ and record the sales data.

Given the sales data, find the current estimates of parameters with the estimation method.

Find the current optimal price vector $\mathbf{r}_{opt}^{i+1}$ with a choice-base dynamic pricing model based on the current estimate.

end for

The performance of the pricing policies are measured with *Percentage Revenue Loss*, which is calculated as the difference between the revenue we obtained under the policy and the optimal revenue we could have achieved if the parameters were known in advance, divided by the optimal revenue. Let Rev_{opt} denote the total optimal revenue obtained with full information about the demand and substitution behaviour and Rev denote the total revenue obtained with a pricing policy. The percentage revenue loss can be calculated by

$$\text{Percentage Revenue Loss} = \frac{\text{Rev}_{opt} - \text{Rev}}{\text{Rev}_{opt}}. \tag{5.3}$$

Table 5.1 shows the comparison of the percentage revenue loss of different pricing policies. The first line is the number of price changes. In our experiment, we keep the price vector unchanged during a day. In the test, the value of ϵ is set to 0.5. For our policies, we set $\tau = 2$. From the table, we note that all the percentage revenue losses are decreasing with the number of price changes. Our RP-a policy provides the best results compared to the others. All the policies return a better revenue than the ϵ-first policy; that is the learning and the earning model. At least, the results show that ϵ-first can be costly, if the user does not set the parameter ϵ properly. For the problem we consider, the passive learning policy provides relatively good results.

	100	200	300	400	500
ϵ-first	47.6	43.7	42.3	41.8	40.9
CYCLE	35.8	35.2	34.5	22.9	21.3
PASSIVE	9.0	7.4	6.9	6.1	6.0
RP	36.0	31.8	29.4	28.8	27.6
RP-a	8.7	7.1	5.9	5.7	5.5

Table 5.1: Comparison of percentage revenue loss of different pricing policies

From previous test, the passive learning policy provides relative good results. The performance of RP and RP-a policy depends on the choice of adjustable parameter τ. To find the effect of adjustable parameter in RP policy, we apply the policy with different values of adjustable parameter and compare the results with the passive learning policy. Due to the runtime of policies, we scale down the problem and use 10 days as the length of sale period. The other setting is the same to the previous test. We run the policies for 30 times and take the average of percentage revenue loss. Since we have scaled down the problem and only 10 days are considered. The estimation results are not good enough to become stable. We do not apply the RP-a policy for this test.

The following table shows the average of percentage revenue loss with different values of adjustable parameter τ and the percentage revenue loss with the passive learning policy, which has the best performance among the benchmark policies in previous test.

policy	percentage revenue loss
$\tau = 0.5$	23.99
$\tau = 1$	19.74
$\tau = 1.5$	18.84
$\tau = 2$	18.67
passive learning	22.21

Table 5.2: Comparison of percentage revenue loss of the passive learning policy and RP policy under different values of adjustable parameter

From the table, we can find that the performance of RP policy depends on the choice of adjustable parameter. Even most of the choices return better results than the passive learning policy, there still exist the value of parameter which return a worse result than the passive learning policy. Therefore, we should choose the value of parameter with caution.

5.4 Conclusion

In this chapter, we propose two pricing policies, RP and RP-a, that balance the exploration of demand and substitution behaviour and exploitation of short-term revenue. The pricing policies are straightforward and easy to implement in practice. They facilitate the user to combine any existing estimation methods and choice-based dynamic pricing methods which have been applied in the users' selling system. The RP-a policy automatically adjusts the ratio of exploration and exploitation by the state of estimates. We perform the numerical studies to show that the logarithm of the average regret of RP policy versus $log(t)$ has a linear form. This empirical result shows that the RP policy achieved ?(T 0.41) order of regret. We also compare the RP and RP-a policies with three alternative policies; ϵ-first, CYCLE and Passive learning. The RP-a policy performs better than the other policies. These two new policies that we describe here will benefit businesses that are introducing a new set of substitutable products into the market.

Online learning has been widely used in the Internet retailing sector and many of these ideas can be useful to traditional RM users, e.g., transportation. These methods are particularly appropriate when new products are being launched. Our pricing policy in MAB is easy to implement and can combine any existing estimation methods and pricing methods that are already applied in firms.

Chapter 6

Conclusion and Future Work

In this chapter, we conclude the contributions in this thesis on the study of the choice-based dynamic pricing problem with demand and substitution uncertainty. In Section 6.1, we summarise the findings of each chapter. In Section 6.2, we set out some future directions that emerge from this thesis.

Revenue management is an important application that helps the seller to make price or quantity decisions and improve their revenue. We focus here on price control, one category of the decisions that have been applied in many industries. While the traditional revenue management model does not consider customer choice behaviour, there is an increasing demand to incorporate more complicated demand models and further improve the revenue. Most of the studies which take customer choice behaviour into account assume that the choice model is known before the optimisation phase. Where choice behaviour is estimated, the trade-off between estimation and optimisation is rarely discussed; the exception being recent papers by Broder and Rusmevichientong (2012), Harrison et al. (2012) and Schwartz et al. (2016). However, in many applications, the parameters in the choice model are hard to estimate and the total revenue can be improved by adjusting the estimation phase and optimisation phase.

The aim of the project is finding a method for optimising the prices when arrival and choice behaviour are uncertain. There are three main research challenges in the project. The first is estimating the parameters of a customer choice model with incomplete data. Our contribution to this challenge is solving the problem with stockout effect with a two-step algorithm or a simple MCMC method. The second challenge is reducing the computational time associated with finding the optimal prices to charge for a choice-based RM problem. We propose a new approximation method that can be used for large instances. The third challenge is developing a pricing strategy that balances the trade-off between estimation and exploitation in choice-based RM. We propose two algorithms in MAB to improve the revenue by balancing the exploration phase and the exploitation phase automatically.

6.1 Findings

In this thesis, we solve the estimation problem with stockout effect which has previously been solved with the EM method. We show that the problem can be transferred to a problem that maximises a globally concave function. Therefore, we can apply maximum likelihood estimation directly to estimate the unknown parameters. In addition, a standard Metro-Hasting algorithm in MCMC can be applied to solve the estimation problem with a unique solution. The estimation methods are assessed using a simulation study and real data from a train company. Results provide us a sense of estimation performance and how much data are needed to have good estimates. The comparison with the EM method is also presented. The comparison of estimation under a fix price vector and a random price vector gives us the idea that offering price vectors with diversity can improve the quality of estimation. Compared to the EM methods that are needed to perform the optimisation process iteratively, the two-step algorithm only needs to maximise a globally concave function and can be applied with existing software.

After we get the estimation of choice model, we propose an approximation method for the choice-based dynamic pricing problem which was shown in our numerical tests to reduce the computational time of the existing approximation method put forward by Zhang and Cooper (2009). With the numerical results, we show that the new approximation method results in revenues that are close to the existing method and can reduce the computation time by 89% (with 1000 time periods and 100 inventory level). The approximation method can benefit firms that solving large-scale choice-based RM problems or those that need to update their pricing policy in a short time.

Last, we propose RP policy and RP-a policy in MAB to balance the estimation phase and the optimisation phase. A numerical study shows that the performance of the RP policy achieves a regret of order $T^{0.41}$. The RP-a policy outperforms three existing policies ϵ-first Policy, CYCLE Policy and Passive learning Policy. From a practical point of view, our policies can combine any existing estimation methods and optimisation methods that are already being applied in a firm. The firm that has adopted a learning and then earning pattern or passive learning pattern can improve their profit by adopting a RP-a policy.

6.2 Limitations and Future work

In Chapter 3, we estimate the demand and substitution behaviour with stockout effect. We assume that we have an exogenous estimate of the aggregate market share. However, the estimation of market share is difficult to obtain in practice. Future work could consider using click-through data and getting rid of the need of market share. In addition, we assume the changes in our firm's prices having a limited impact on other competitors' prices, therefore, the new market share can be calculated based on our firm's prices only. Considering the situation that other competitors will respond to the changes in our firm's prices can be another direction of future work.

In Chapter 4, we treat the trains with same origin-destination pair as substitutable products and ignore the network effect. Zhang and Lu (2013) consider a dynamic pricing problem in network revenue management. Du et al. (2016) study a pricing problem with network effect. In the paper, they apply a MNL model to consider customer choice behaviour. Another research direction can be incorporating dynamic pricing for network revenue management with other choice models. Another limitation in the study is the assumption that each customer purchases at most one product. However, some customers may choose to purchase more than one product at a time in practice. Incorporating this situation may be another direction of future work. In this study, we construct the lower bound of the value function for a discrete time model from a lower bound for a continuous time model. We should find a tighter lower bound in the same computation time and improve the performance of the approximation method.

In Chapter 5, the regret of the pricing policies is only shown with numerical study. A theoretical analysis of the policies should be one direction of future work. In our study, we apply random price vectors to provide the diversity of price vector. We control the use of random price vectors with a probability function. It is worth to investigate the different ways of generating price vectors. In addition, a more thorough computational study of the effectiveness of the random price vector should be another direction of future work.

In addition, we only use the MNL model to estimate customer choice behaviour. We assume that all the customers have the same choice behaviour and that the choice behaviour is time-invariant. These assumptions can be reconsidered and lead to a future direction. Li and Huh (2011) and Gallego and Wang (2014) study a multiple-products pricing problem with a Nested Logit model. There is a research gap to incorporate the heterogeneity of the customer choice model in dynamic pricing problems.

References

[1] Daniel A Ackerberg and Marc Rysman. Unobserved product differentiation in discrete choice models: Estimating price elasticities and welfare effects. Technical report, National Bureau of Economic Research, 2002.

[2] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *COLT*, pages 39–1, 2012.

[3] Yalcin Akcay, Harihara Prasad Natarajan, and Susan H Xu. Joint dynamic pricing of multiple perishable products under consumer choice. *Management Science*, 56(8):1345–1361, 2010.

[4] Victor F Araman and René Caldentey. Dynamic pricing for nonperishable products with demand learning. *Operations research*, 57(5):1169–1188, 2009.

[5] Neeraj Arora and Joel Huber. Improving parameter estimates and model prediction by aggregate customization in choice experiments. *Journal of Consumer Research*, 28(2):273–283, 2001.

[6] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[7] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*, pages 322–331. IEEE, 1995.

[8] Yossi Aviv and Amit Pazgal. Pricing of short life-cycle products through active learning. *Under revision for Management Science*, 2002.

[9] Yossi Aviv and Amit Pazgal. A partially observed markov decision process for dynamic pricing. *Management Science*, 51(9):1400–1416, 2005.

[10] M Ben-Akiva and SR Lerman. Discrete choice analysis: Theory and application to travel demand, 1985.

[11] Steven T Berry. Estimating discrete-choice models of product differentiation. *The RAND Journal of Economics*, pages 242–262, 1994.

[12] Omar Besbes and Alp Muharremoglu. On implications of demand censoring in the newsvendor problem. *Management Science*, 59(6):1407–1424, 2013.

[13] Omar Besbes and Assaf Zeevi. Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research*, 57(6):1407–1420, 2009.

[14] Omar Besbes and Assaf Zeevi. On the (surprising) sufficiency of linear models for dynamic pricing with demand learning. *Management Science*, 61(4):723–739, 2015.

[15] Chandra R Bhat. Quasi-random maximum simulated likelihood estimation of the mixed multinomial logit model. *Transportation Research Part B: Methodological*, 35(7):677–693, 2001.

[16] Gabriel Bitran and René Caldentey. An overview of pricing models for revenue management. *Manufacturing & Service Operations Management*, 5(3):203–229, 2003.

[17] Gabriel R Bitran and Susana V Mondschein. Periodic pricing of seasonal products in retailing. *Management Science*, 43(1):64–79, 1997.

[18] Josef Broder and Paat Rusmevichientong. Dynamic pricing under a general parametric choice model. *Operations Research*, 60(4):965–980, 2012.

[19] Juan José Miranda Bront, Isabel Méndez-Díaz, and Gustavo Vulcano. A column generation algorithm for choice-based network revenue management. *Operations Research*, 57(3):769–784, 2009.

[20] Lucian Busoniu, Robert Babuska, Bart De Schutter, and Damien Ernst. *Reinforcement learning and dynamic programming using function approximators*, volume 39. CRC press, 2010.

[21] Nicolò Cesa-Bianchi and Paul Fischer. Finite-time regret bounds for the multi-armed bandit problem. In *ICML*, pages 100–108. Citeseer, 1998.

[22] Yiwei Chen and Vivek F Farias. Simple policies for dynamic pricing with imperfect forecasts. *Operations Research*, 61(3):612–624, 2013.

[23] CSM Currie, RCH Cheng, and HK Smith. Dynamic pricing of airline tickets with competition. *Journal of the Operational Research Society*, 59(8):1026–1037, 2008.

[24] Ricardo A Daziano and Denis Bolduc. Incorporating pro-environmental preferences towards green automobile technologies through a bayesian hybrid choice model. *Transportmetrica A: Transport Science*, 9(1):74–106, 2013.

[25] P Dellaportas and Adrian FM Smith. Bayesian inference for generalized linear and proportional hazards models via gibbs sampling. *Applied Statistics*, pages 443–459, 1993.

[26] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[27] Arnoud V den Boer. Dynamic pricing with multiple products and partially specified demand distribution. *Mathematics of operations research*, 39(3):863–888, 2014.

[28] Arnoud V den Boer. Dynamic pricing and learning: historical origins, current research, and new directions. *Surveys in operations research and management science*, 20(1):1–18, 2015.

[29] Arnoud V den Boer and Bert Zwart. Simultaneously learning and optimizing using controlled variance pricing. *Management Science*, 60(3):770–783, 2013.

[30] Lingxiu Dong, Panos Kouvelis, and Zhongjun Tian. Dynamic pricing and inventory control of substitute products. *Manufacturing & Service Operations Management*, 11(2):317–339, 2009.

[31] Chenhao Du, William L Cooper, and Zizhuo Wang. Optimal pricing for a multinomial logit choice model with network effects. *Operations Research*, 64(2):441–455, 2016.

[32] Wedad Elmaghraby and Pınar Keskinocak. Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. *Management Science*, 49(10):1287–1309, 2003.

[33] Vivek F Farias, Srikanth Jagabathula, and Devavrat Shah. A nonparametric approach to modeling choice with limited data. *Management Science*, 59(2):305–322, 2013.

[34] Awi Federgruen and Aliza Heching. Combined pricing and inventory control under uncertainty. *Operations research*, 47(3):454–475, 1999.

[35] Youyi Feng and Guillermo Gallego. Optimal starting times for end-of-season sales and optimal stopping times for promotional fares. *Management Science*, 41(8):1371–1391, 1995.

[36] Youyi Feng and Guillermo Gallego. Perishable asset revenue management with markovian time dependent demand intensities. *Management Science*, 46(7):941–956, 2000.

[37] Youyi Feng and Baichun Xiao. A continuous-time yield management model with multiple prices and reversible price changes. *Management Science*, 46(5):644–657, 2000.

[38] Sylvia Frühwirth-Schnatter and Rudolf Frühwirth. Auxiliary mixture sampling with applications to logistic models. *Computational Statistics & Data Analysis*, 51(7):3509–3528, 2007.

[39] Guillermo Gallego and Ming Hu. Dynamic pricing of perishable assets under competition. *Management Science*, 60(5):1241–1259, 2014.

[40] Guillermo Gallego, Garud Iyengar, R Phillips, and Abha Dubey. Managing flexible products on a network. 2004.

[41] Guillermo Gallego and Garrett Van Ryzin. Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management science*, 40(8):999–1020, 1994.

[42] Guillermo Gallego and Garrett Van Ryzin. A multiproduct dynamic pricing problem and its applications to network yield management. *Operations Research*, 45(1):24–41, 1997.

[43] Guillermo Gallego and Ruxian Wang. Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Operations Research*, 62(2):450–461, 2014.

[44] Dani Gamerman. Sampling from the posterior distribution in generalized linear mixed models. *Statistics and Computing*, 7(1):57–68, 1997.

[45] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, 1984.

[46] John Gittins, Kevin Glazebrook, and Richard Weber. *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.

[47] John Gittins and D Jones. Progress in statistics. *Amsterdam: North-Holland*, 2:9, 1974.

[48] Ronald L Goettler and Ron Shachar. Spatial competition in the network television industry. 2001.

[49] Joachim Grammig, Reinhard Hujer, and Michael Scheidler. Discrete choice modelling in airline network management. *Journal of Applied Econometrics*, 20(4):467–486, 2005.

[50] Peng Guo, Baichun Xiao, and Jun Li. Unconstraining methods in revenue management systems: Research overview and prospects. *Advances in Operations Research*, 2012, 2012.

[51] J Michael Harrison, N Bora Keskin, and Assaf Zeevi. Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science*, 58(3):570–586, 2012.

[52] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[53] Stephane Hess, Kenneth E Train, and John W Polak. On the use of a modified latin hypercube sampling (mlhs) method in the estimation of a mixed logit model for vehicle choice. *Transportation Research Part B: Methodological*, 40(2):147–163, 2006.

[54] Chris C Holmes, Leonhard Held, et al. Bayesian auxiliary variable models for binary and multinomial regression. *Bayesian analysis*, 1(1):145–168, 2006.

[55] Leslie Pack Kaelbling. *Learning in embedded systems*. MIT press, 1993.

[56] N Bora Keskin and Assaf Zeevi. Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Operations Research*, 62(5):1142–1167, 2014.

[57] WM Kinkaid and Donald A Darling. An inventory pricing problem: technical note. 1962.

[58] Sumit Kunnumkal and Huseyin Topaloglu. A new dynamic programming decomposition method for the network revenue management problem with customer choice behavior. *Production and Operations Management*, 19(5):575–590, 2010.

[59] Tze Leung Lai. Adaptive treatment allocation and the multi-armed bandit problem. *The Annals of Statistics*, pages 1091–1114, 1987.

[60] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[61] Lawrence M Leemis. Nonparametric estimation of the cumulative intensity function for a nonhomogeneous poisson process. *Management Science*, 37(7):886–900, 1991.

[62] Benoît Leloup and Laurent Deveaux. Dynamic pricing on the internet: Theory and simulations. *Electronic Commerce Research*, 1(3):265–276, 2001.

[63] Benjamin Letham, Lydia M Letham, and Cynthia Rudin. Bayesian inference of arrival rate and substitution behavior from sales transaction data with stockouts. *arXiv preprint arXiv:1502.04243*, 2015.

[64] Guang Li, Paat Rusmevichientong, and Huseyin Topaloglu. The d-level nested logit model: Assortment and price optimization problems. *Operations Research*, 63(2):325–342, 2015.

[65] Hongmin Li and Woonghee Tim Huh. Pricing multiple products with the multinomial logit and nested logit models: Concavity and implications. *Manufacturing & Service Operations Management*, 13(4):549–563, 2011.

[66] Qian Liu and Garrett Van Ryzin. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management*, 10(2):288–310, 2008.

[67] R.. Ducan Luce. *Individual Choice Behavior a Theoretical Analysis*. John Wiley and sons, 1959.

[68] Constantinos Maglaras and Joern Meissner. Dynamic pricing strategies for multiproduct revenue management problems. *Manufacturing & Service Operations Management*, 8(2):136–148, 2006.

[69] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[70] Hazel A Morrow-Jones, Elena G Irwin, and Brian Roe. Consumer preference for neotraditional neighborhood characteristics. *Housing Policy Debate*, 15(1):171–202, 2004.

[71] Andrés Musalem, Marcelo Olivares, Eric T Bradlow, Christian Terwiesch, and Daniel Corsten. Structural estimation of the effect of out-of-stocks. *Management Science*, 56(7):1180–1197, 2010.

[72] Jeffrey P Newman, Mark E Ferguson, Laurie A Garrow, and Timothy L Jacobs. Estimation of choice-based models using sales data from a single firm. *Manufacturing & Service Operations Management*, 16(2):184–197, 2014.

[73] Richard M Ratliff, B Venkateshwara Rao, Chittur P Narayan, and Kartik Yellepeddi. A multi-flight recapture heuristic for estimating unconstrained demand from airline bookings. *Journal of Revenue and Pricing Management*, 7(2):153–171, 2008.

[74] W Zachary Rayfield, Paat Rusmevichientong, and Huseyin Topaloglu. Approximation methods for pricing problems under the nested logit model with price bounds. *INFORMS Journal on Computing*, 27(2):335–357, 2015.

[75] Christian Robert and George Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2013.

[76] Michael Rothschild. A two-armed bandit theory of market pricing. *Journal of Economic Theory*, 9(2):185–202, 1974.

[77] Paat Rusmevichientong and Huseyin Topaloglu. Robust assortment optimization in revenue management under the multinomial logit choice model. *Operations Research*, 60(4):865–882, 2012.

[78] Steven L Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

[79] Steven L Scott. Data augmentation, frequentist estimation, and the bayesian analysis of multinomial logit models. *Statistical Papers*, 52(1):87–109, 2011.

[80] Alper Şen. A comparison of fixed and dynamic pricing policies in revenue management. *Omega*, 41(3):586–597, 2013.

[81] Zuo-Jun Max Shen and Xuanming Su. Customer behavior modeling in revenue management and auctions: A review and new research opportunities. *Production and operations management*, 16(6):713–728, 2007.

[82] Minsuk Suh and Goker Aydin. Dynamic pricing of substitutable products with limited inventories under logit demand. *IIE transactions*, 43(5):323–331, 2011.

[83] Kalyan Talluri. A finite-population revenue management model and a risk-ratio procedure for the joint estimation of population size and parameters. *Available at SSRN 1374853*, 2009.

[84] Kalyan Talluri and Garrett Van Ryzin. Revenue management under a general discrete choice model of consumer behavior. *Management Science*, 50(1):15–33, 2004.

[85] Kalyan T Talluri and Garrett J Van Ryzin. *The theory and practice of revenue management*, volume 68. Springer Science & Business Media, 2006.

[86] Pouya Tehrani, Yixuan Zhai, and Qing Zhao. Dynamic pricing under finite space demand uncertainty: a multi-armed bandit with dependent arms. *arXiv preprint arXiv:1206.5345*, 2012.

[87] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[88] Kenneth E Train. *Discrete choice methods with simulation.* Cambridge university press, 2009.

[89] Garrett van Ryzin and Gustavo Vulcano. Computing virtual nesting controls for network revenue management under customer choice behavior. *Manufacturing & Service Operations Management*, 10(3):448–467, 2008.

[90] Garrett van Ryzin and Gustavo Vulcano. An expectation-maximization algorithm to estimate a general class of non-parametric choice models. *preprint*, 2011.

[91] Garrett van Ryzin and Gustavo Vulcano. A market discovery algorithm to estimate a general class of nonparametric choice models. *Management Science*, 61(2):281–300, 2014.

[92] Joannes Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European conference on machine learning*, pages 437–448. Springer, 2005.

[93] Gustavo Vulcano, Garrett Van Ryzin, and Wassim Chaar. Om practice-choice-based revenue management: An empirical study of estimation and optimization. *Manufacturing & Service Operations Management*, 12(3):371–392, 2010.

[94] Gustavo Vulcano, Garrett Van Ryzin, and Richard Ratliff. Estimating primary demand for substitutable products from sales transaction data. *Operations Research*, 60(2):313–334, 2012.

[95] Christopher John Cornish Hellaby Watkins. *Learning from delayed rewards*. PhD thesis, University of Cambridge England, 1989.

[96] Chieh-Hua Wen and Frank S Koppelman. The generalized nested logit model. *Transportation Research Part B: Methodological*, 35(7):627–641, 2001.

[97] CF Jeff Wu. On the convergence properties of the em algorithm. *The Annals of statistics*, pages 95–103, 1983.

[98] Xinan Yang, Arne K Strauss, Christine SM Currie, and Richard Eglese. Choice-based demand management and vehicle routing in e-fulfillment. *Transportation science*, 50(2):473–488, 2014.

[99] Dan Zhang and Daniel Adelman. An approximate dynamic programming approach to network revenue management with customer choice. *Transportation Science*, 43(3):381–394, 2009.

[100] Dan Zhang and William L Cooper. Revenue management for parallel flights with customer-choice behavior. *Operations Research*, 53(3):415–431, 2005.

[101] Dan Zhang and William L Cooper. Pricing substitutable flights in airline revenue management. *European Journal of Operational Research*, 197(3):848–861, 2009.

[102] Dan Zhang and Zhaosong Lu. Assessing the value of dynamic pricing in network revenue management. *INFORMS Journal on Computing*, 25(1):102–115, 2013.

[103] Wen Zhao and Yu-Sheng Zheng. Optimal dynamic pricing for perishable assets with nonhomogeneous demand. *Management science*, 46(3):375–388, 2000.