Return Rate Prediction

Erik Karlén and Caspar Welin

Master's thesis
2017:E46

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Centrum Scientiarum Mathematicarum

LUND UNIVERSITY

MASTER THESIS

Return Rate Prediction

Authors:
Erik KARLÉN
Caspar WELIN

Supervisors:
Johan LINDSTRÖM
Leon ANDERSSON
Johannes ROSENBERG

Centre for Mathematical Sciences

June 26, 2017

Lund University

Abstract

Faculty of Engineering

Centre for Mathematical Sciences

Return Rate Prediction

by Erik KARLÉN & Caspar WELIN

Product quality is a major concern for all companies. Predicting the lifetime return rate allows for identification of products with atypically high return rates. Such products might otherwise not have been detected before it is too late to take any action to reduce the return rate. Furthermore, predicting a product's return rate allows for estimation of the company's expected cost of handling returns.

This thesis explores multiple methods for predicting a product's lifetime return rate. One of the methods investigated utilizes a mixture cure model for the time to return distribution and includes a parameter for the probability of return. Different time to return distributions were used in the model; the negative binomial distribution and the Weibull distribution. The parameters of the model were estimated using variations of the Expectation Maximization algorithm.

A simple way of estimating the lifetime return rate is by simply dividing the number of observed returns with the number of observed sales, called the aggregated return rate. All methods tested outperformed the aggregated return rate and the most accurate method proved to be the cure mixture model using a negative binomial distribution.

Acknowledgements

We would like to thank our supervisor Johan Lindström at the Centre for Mathematical Sciences, LTH, for his guidance and valuable advice.

We also want to express our gratitude to Axis Communications for the idea of the thesis and for allowing us to use their resources and data. Furthermore we want to thank our supervisor Leon Andersson and co-supervisors Johannes Rosenberg and Carl Oreborg at Axis Communications for their support and useful comments.

Contents

List of Abbreviations

API	**A**pplication **P**rogramming **I**nterface
ARR	**A**ggregated **R**eturn **R**ate
CDF	**C**umulative **D**istribution **F**unction
DLM	**D**istributed **L**ag **M**odel
ECM	**E**xpectation **C**onditional **M**aximization
ECME	**E**xpectation **C**onditional **M**aximization **E**ither
EM	**E**xpectation **M**aximization
iid	**i**ndependent and **i**dentically **d**istributed
KM	**K**aplan-**M**eier estimator
LRR	**L**ifetime **R**eturn **R**ate
NB	**N**egative **B**inomial distribution
PDF	**P**robability **D**ensity **F**unction
PIA	**P**roduct **I**nformation **A**PI
pmf	**p**robability **m**ass **f**unction
RMA	**R**eturn **M**aterial **A**uthorization
Wbl	**W**ei**b**ull distribution

List of Symbols

t	current time
p	return rate
$\boldsymbol{\theta}$	distribution parameters
ρ	PDF or pmf of the time to return distribution
R	CDF of the time to return distribution
$\mathcal{L}$	Likelihood function
ℓ	log-likelihood function
Q	Expectation of the complete log-likelihood function
$t_i^{(o)}$	Observed time to return for unit i
$\mathbf{t}^{(o)}$	Vector containing $t_i^{(o)}, i = 1, \ldots, m$
$T_i^{(u)}$	Stochastic variable representing the unobserved return time for unit i
$\mathbf{T}^{(u)}$	Vector containing $T_i^{(u)}, i = m+1, \ldots, n$
s_i	Period when unit i was sold
S	Survival function of the time to return distribution
•	$\mathbf{t}^{(o)}, \mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}, p^{(j)}, \boldsymbol{\theta}^{(j)}$
n	number of units sold
m	number of units returned
$\mathbf{x}$	observed data

1 Introduction

Faulty units is a problem that all companies that sell physical products have to deal with. Selling such units may lead to dissatisfied customers so usually companies will have a return policy to make up for it. Commonly the defective unit will be either replaced or repaired. This whole process can cost a significant amount of money for a company if they have many faulty units. It is therefore in the company's best interest to minimize the number of faulty units sold.

To reduce return rates it is important to as early as possible in a product's life span (the time from when units start being sold to when no more units are returned or sold) find out if there is a problem with the product which may lead to an unusually high number of returns. Detecting if there is a problem with a certain product is not simple. One way is by looking at the sold and returned units and use information about them to try to predict what the lifetime return rate (LRR) will be. The lifetime return rate is simply the number of units returned divided by the number of units sold at the end of a product's life span.

Getting an estimation of the number of eventual returns of a product is also of great interest for a company. This information can be used to estimate the expected cost of handling the returns and give an idea of e.g. the number of spare parts that should be kept in store. By multiplying the predicted return rate with the number of sold units one acquires an estimation of the expected number of returns.

The purpose of this master's thesis is to analyze real data for different products and try to predict the lifetime return rate as early as possible in the life span of those products. Several different methods for predicting the LRR were evaluated and are compared in this report.

This project was carried out in cooperation with Axis Communications AB. They mainly design and produce network cameras and are interested in the return rates of their products. Axis believes that predicting the LRR far in advance will enable them to take preemptive actions to decrease the return rate. They allowed us to use the data they have collected for their products, stored in their Return Material Authorization (RMA), Product Information API (PIA) and Industrial and Financial Systems (IFS) databases.

A very simple but naive way of estimating the lifetime return rate of a product is by taking the total number of returns obtained so far and divide it by the total number of sales obtained so far. This is called the aggregated return rate (ARR) and may give a decent prediction at the end of a product's life span, since only a few more sales and returns will be observed. However, early in a product's life span the ARR is likely to underestimate the LRR. This is because the returns are delayed relative to the sales time, meaning it can take a long time for the aggregated return rate to stabilize. By then it is often too late to do anything to decrease the return rate since the product most likely is late in its life span.

Alternatives to ARR have been suggested, and include an approach using Bayesian estimation of a distributed lag model as done by Toktay, Wein, and Zenios [25], Krapp, Nebel, and Srivastava [15], and Clottey, Benton, and Srivastava [5]. This method does not take advantage of item-level information, i.e. information about the sales and return date of each individual unit being analyzed. Instead it uses what is known as period-level information, i.e. only the total amount of sales and returns in each time period is used.

A model using item-level information was tested by Toktay, Wein, and Zenios [25] and the return rate was predicted by calculating the maximum likelihood parameter estimations using the expectation maximization (EM) algorithm. Similarly, Balakrishnan and Pal [2] use a cure rate model to estimate the cure rate by solving the maximum likelihood using the EM-algorithm and computes standard errors with the observed Fisher information matrix. However, they apply the model to clinical data to estimate the number of patients that are cured by a treatment in contrast to using it for predicting a product's return rate.

It is also possible to include other information in prediction models in the form of covariates, this is done in a paper by Hess and Mayhew [11]. They use a split adjusted hazard model with item-level information about the sold and returned units from an apparel company. Wu et al. [26] use a similar approach and model the cure rate using logistic regression, allowing for covariates, and then solve the resulting estimation problem using the EM-algorithm.

In this thesis we investigate and compare different methods for predicting the lifetime return rate and expand on previously proposed methods. We first introduce concepts in survival analysis and a method based on the Kaplan-Meier estimator. Then cure models and the model used for a unit's time to return are described. This is followed by some less successful ways of predicting the return rate. A method which uses the EM-algorithm and variations of the EM-algorithm is then presented. Furthermore the calculations and variance calculations for a negative binomial and Weibull time to return distribution when using the EM-algorithm are shown. The results of using the methods analyzed are presented for a few products.

2 Data

The data used in this project was obtained from three of Axis' databases (IFS, RMA and PIA). The IFS database contains sales information e.g. when a unit was sold together with a unique serial number. The return data was fetched from the RMA-database which contains e.g. when a specific unit was returned together with its serial number. The PIA-database was used to identify the product type associated with each unit's serial number.

2.1 Issues with the data

Working with data collected from real life, i.e. non-generated data, nearly always introduces some issues. This section contains the main difficulties that we encountered when working with the data that was provided to us. We also describe how we chose to handle these problems.

2.1.1 Load times in supply chain

Unfortunately we did not have access to the exact date when a unit was delivered to a customer but rather the date at which the unit was shipped to a distributor. This means that the time to return modelled here includes an unknown amount of time during which the unit was stored at a distributor storage.

The time the unit spends at the distributor storage changes the shape of the distribution of the time to return. Hence our observed time to return distribution is not simply the time between the customer receiving the unit and the customer returning the unit. Instead it is the sum of what can be seen as two stochastic variables, the time the unit spent at the distributor storage T_d and the time the customer owned the unit before returning it T_c. We can therefore write the observed time to return T for a single unit as

$$T = T_d + T_c. \tag{2.1}$$

Modeling T_d and T_c independently to acquire an expression for the distribution of T will often be too complicated. An alternative to modeling both T_d and T_c is to model the sum T directly from the observed data.

2.1.2 Missing data

Most of the data in the databases is entered manually and there are inevitably some errors. The more common errors include lack of sales dates or return dates that

predate the sales date. In section 5.3 we discuss some potential extensions of our model to handle these errors. Entries with such errors are rare and we have therefore decided not to include them in our analysis.

2.2 Choosing products with accurate lifetime return rate

To be able to evaluate how accurate the predictions are, it is necessary to know what the true lifetime return rates are. The only way of measuring the true lifetime return rate for a product is by looking at the aggregated return rate at the end of the product's life span. Hence for many of Axis' products it is not possible to measure the true lifetime return rate since they are still being sold and/or returned. When choosing what products to analyze it is therefore important to make sure an accurate lifetime return rate has been obtained and can be used for comparisons. Consequently we want to choose products that have not received any new returns or sales for a long time. However the problem with this is that some of Axis' older products lack return data from early in their life span. These products can therefore not be used to evaluate our predictions. As a result, the products we analyzed were carefully selected to make sure their data and lifetime return rates were available.

3 Methodology

This thesis investigates several different methods for predicting the return rate of Axis' products. Including both simple and more advanced methods. For example, a simple but naive way is by taking the total amount of returns obtained so far and divide it by the total amount of sales obtained so far, i.e. the aggregated return rate (ARR). Some more advanced methods that we have tested are Kaplan-Meier estimations and expectation maximization algorithms which are described in detail in this chapter.

When analyzing subjects which may experience an event being investigated after an unknown amount of time, one is often interested in a subject's time to event. This is simply the time it takes for a subject to experience the event. In our case the event being investigated is the return of a unit and we are therefore interested in the time to return for the unit being analyzed.

3.1 Censoring

Often when dealing with time to event data some of the observations are incomplete. By incomplete we mean that the exact time of the event is not known, only that the event will happen in some interval of time. This is called censoring. There are several types of censoring e.g. right censored, left censored and interval censored.

When data is right censored the only observed information is that the time of event occurs after the censoring time. E.g. in a clinical study of the mortality of patients with some particular disease the censoring time is typically the end of the study. This means that for patients which have died during the study the exact lifetime is known but for patients still alive at the end of the study, i.e. the censoring time point, it is only known that death will occur after the censoring.

Similarly left censored is when the event has occurred before some censoring time which in this case could be that death occurred before the start of the study. Interval censoring is when the event is known to occur in an interval which may happen e.g. when patients are only observed periodically, such as at yearly check ups. [13]

The type of censoring that is of interest in this study is right censoring. This is due to the fact that we have either observed a time of return for a unit or the unit had not yet been returned at the time of observation. In this case it is known that the unit will either be returned after the time of observation or not at all.

3.2 Survival function

When dealing with right-censored data, it is not ideal to use normal statistical methods for the analysis [see e.g. 1]. Instead there are certain tools and concepts for dealing with such data. One of these concepts which is useful in our analysis is the so called survival function. In a population of subjects to which a specific event may happen the survival function simply tells you the proportion of the population that is expected to have experienced the event at a certain time. If the survival function is denoted S and the random time at which the event occurs is T, then the survival function can be written as

$$S(t) = P(T > t) = 1 - P(T \leq t). \tag{3.1}$$

As the above equation shows, the survival function is the complement of the cumulative distribution function for T. An example of a survival function curve is shown in figure 3.1. Normally the survival function will go towards zero as t increases, this is however not always the case. For events that may not necessarily occur for the entire population being investigated the survival function will tend to a non-zero value as time approaches infinity. In such cases it is useful to use a cure model as described in Section 3.4.

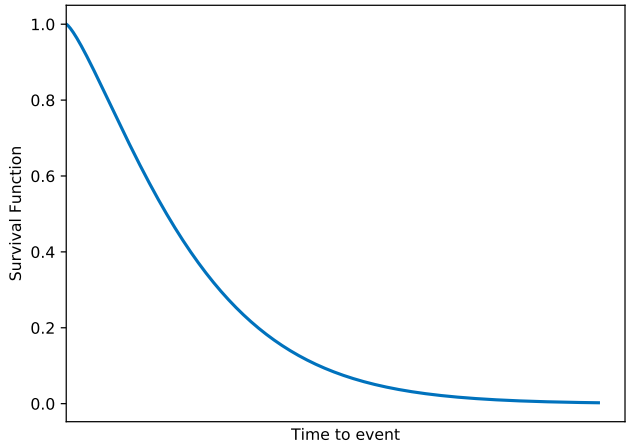

FIGURE 3.1: Example of a survival function curve.

3.3 Kaplan-Meier estimator

A common non-parametric way of estimating the survival function from censored survival data is with the Kaplan-Meier estimator [12]. This estimator gives a step function estimation of the survival curve up to the time of the last uncensored data point. Let us say we want to calculate the Kaplan-Meier estimation of the survival function at time t. To do this we need to know the number of subjects that can still be affected by the event of interest just before time t, let this be denoted by $Y(t)$. Moreover, let $T_1 < T_2 < \ldots$ denote the times where at least one event of interest has occurred. Finally we let d_j be the number of tied survival times at time T_j. The

Kaplan-Meier estimator of the survival function $\hat{S}(t)$ can now be written as

$$\hat{S}(t) = \prod_{T_j \leq t} \left(1 - \frac{d_j}{Y(T_j)}\right). \tag{3.2}$$

To calculate the variance σ of a Kaplan-Meier estimator there are a a few different possibilities; a common choice is Greenwood's formula [1]:

$$\sigma^2(t) = \hat{S}(t)^2 \sum_{T_j \leq t} \frac{d_j}{Y(T_j)(Y(T_j) - d_j)}. \tag{3.3}$$

3.4 Cure models

A cure model is a type of model which can be useful for certain censored survival data [see 14]. Such models may be used effectively to describe cases where there is a chance that some of the subjects will never experience the event being investigated. E.g. the portion of people that will be cured and not die from a certain illness, or in our case units that are never returned. In data where a cure model is appropriate, the survival function will tend to a non-zero value called the "cure rate". One way of finding out whether a cure model is a decent model for a given data set is by looking at a Kaplan-Meier plot. If the Kaplan-Meier estimator plot seems to tend to a non-zero value as the time increases a cure model might be appropriate.

One type of cure model is the mixture cure model. It considers two groups, one consisting of cured people and another of those who are not cured. The cured group will never experience the event whereas the non-cured group will experience the event eventually. The survival function of an event that follows a cure model is

$$S(t) = P(T > t) = 1 - p + pS_u(t) \tag{3.4}$$

where $1 - p$ is the cure rate and p is consequently the proportion of subjects that will never be cured and S_u is the survival function for those subjects. In figure 3.2 an example of the survival function when using a cure model is shown.

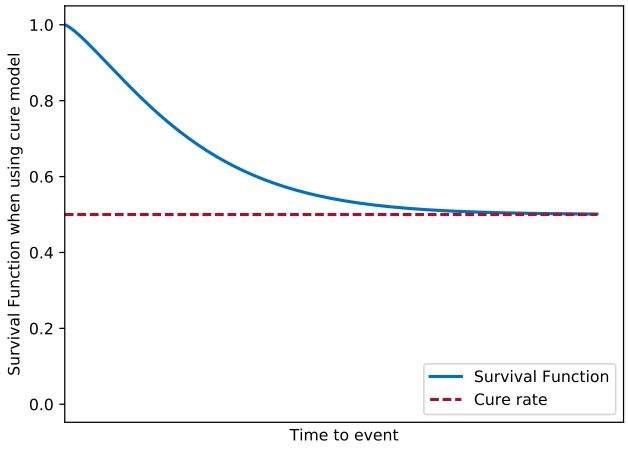

FIGURE 3.2: Example of a survival function when using a cure model. The dashed red line represents the cure rate.

3.4.1 Estimating the return rate using the Kaplan-Meier estimator

In a cure model it may be interesting to estimate the proportion of subjects that will never be cured, in our case a product's lifetime return rate, i.e. p. Maller and Zhou [20] propose using the last value of the Kaplan-Meier function as an estimation of p.

$$\hat{p} = 1 - \hat{S}(T_n) \tag{3.5}$$

where T_n is the largest observed or censored time to return. This estimation is asymptotically consistent if the maximum observable time to return is less than or equal to the maximum obtainable censored time to return. Maller and Zhou also show that, with modest conditions on the censored data, the estimation is asymptotically normal.

3.5 Returns model

For the other methods we used to predict the return rate we needed a model of the time to return. Following the notation used by Toktay, Wein, and Zenios [25] we denote the current time t and we assume that n and m units have been sold and returned by time t, respectively. The units are indexed such that units $i = 1, \ldots, m$ have been returned and a time to return has been observed, these times are denoted $t_i^{(o)}$. Let s_i denote the time when the i:th unit was shipped for $i = 1, 2, \ldots, n$. The aggregated return rate for time t can now be expressed as

$$p_{ARR} = \frac{m}{n}. \tag{3.6}$$

An example of what a plot of the ARR for an entire product life span typically looks like is shown in figure 3.3.

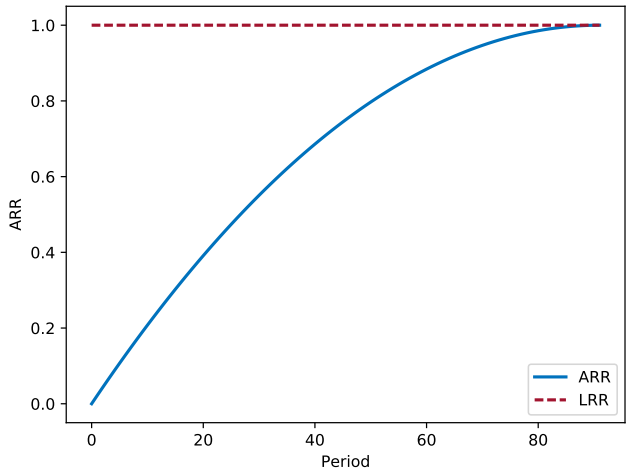

FIGURE 3.3: Example of an ARR plot.

For units $i = m + 1, \ldots, n$ our only observations are that the time to return is greater than $t - s_i$ i.e. the observations are right censored. Items that are never returned are defined to have an infinite time to return. The elapsed time between the shipping

time and the return time, i.e. the time to return, for product i is denoted T_i, these are assumed to be iid.

Since some units are never returned a mixture cure model is suitable to describe the distribution for T_i. This is because the distribution must have some probability mass at $T_i = \infty$ as well as probability mass at $T_i < \infty$. The probability that a unit is ever returned is p and the distribution for the time to return given that the unit will be returned is denoted ρ with parameters $\boldsymbol{\theta}$. This gives the following mixture cure model for T_i.

$$P(T_i = t_i) = (1-p)\delta(t_i = \infty) + p \cdot \rho(t_i|\boldsymbol{\theta}) \tag{3.7}$$

where the δ is the Kronecker delta function in the case of a discrete time to return distribution ρ and the Dirac delta in the case of a continuous time to return distribution.

The probability of observing a time to return greater than $t - s_i$ is

$$P(T_i > t - s_i) = \sum_{l=t-s_i+1}^{\infty} (1-p)\delta(l = \infty) + p \cdot \rho(l|\boldsymbol{\theta}) = 1 - p + pS_\rho(t - s_i|\boldsymbol{\theta}) \tag{3.8}$$

for a discrete distribution where S_ρ is the survival function corresponding to ρ and

$$P(T_i > t - s_i) = \int_{t-s_i}^{\infty} ((1-p)\delta(\tau = \infty) + p \cdot \rho(\tau|\boldsymbol{\theta}))d\tau = 1 - p + pS_\rho(t - s_i|\boldsymbol{\theta}) \tag{3.9}$$

for a continuous distribution.

3.6 Distributed Lag Model

One way of predicting the lifetime return rate is by using a distributed lag model (DLM) which uses period-level information. A DLM is a time series model which describes the value of a stochastic variable in a certain period as a linear combination of past values of an independent stochastic variable [3]. In the case of product returns, the number of returns will depend on the number of sales in previous and current periods. If m_t and n_t represent the number of returns and sales in period t, respectively, and β_k represents the probability of return of a product after k periods then the DLM can be written as

$$m_t = \beta_0 n_t + \beta_1 n_{t-1} + \ldots + \beta_{t-1} n_1 + e_t, \quad t = 1, 2, \ldots, T. \tag{3.10}$$

Modeling the probabilities β_k requires a distribution for the time to return. One possibility is to choose

$$\beta_k = p \cdot \rho(k|\boldsymbol{\theta}). \tag{3.11}$$

The parameters can then be estimated by assuming a distribution for the error terms e_t. This leads to a distribution for the parameters. Estimations of the parameters can then calculated by either maximizing the distribution, i.e. finding the mode, or by calculating the expected values of the distribution. In general this is a difficult problem which may require numerical methods such as Markov chain Monte Carlo algorithms to solve. We used a DLM for predicting the return rate but we decided

to not include any results from using it due to the difficulty in solving the problem numerically and the poor predictions.

3.7 Prediction using survival function

Another way of predicting the lifetime return rate is based on the survival function in (3.4). Solving this equation for p gives

$$p = \frac{1 - S(t)}{1 - S_u(t)} \tag{3.12}$$

which means that p can be estimated if we know both survival functions S and S_u. These functions can be estimated in many different ways, e.g. they can be estimated using the Kaplan-Meier estimator or by parameterization and maximizing likelihood. When the functions S and S_u are known p can be estimated by either taking the values of S and S_u for a specific time, or as the average of values for multiple times. This is similar to the method described in section 3.4.1 but requires an estimation of S_u. These methods give similar results when estimating S_u using maximum likelihood for the observed returns and therefore we decided to not include the results from this method.

3.8 Maximum likelihood parameter estimation

Maximum likelihood parameter estimation is a general method for finding parameters of a model given data. The likelihood is the probability of observing the data that was observed given the model [24]. Finding the parameters that maximizes this value i.e. makes the probability of observing the data given the model as large as possible is the maximum likelihood parameter estimation method.

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, P(\mathbf{x}|\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \mathcal{L}(\boldsymbol{\theta}|\mathbf{x}) \tag{3.13}$$

In some cases one might have some prior information of what parameter values are more likely. This can be summarized in a prior distribution for the parameters. The posterior distribution i.e. the distribution for the parameters given the data using a prior distribution is then given by Bayes' theorem. This way of estimating parameters using a prior distribution is known as maximum a posteriori parameter estimation.

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, P(\boldsymbol{\theta}|\mathbf{x}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, P(\mathbf{x}|\boldsymbol{\theta})P(\boldsymbol{\theta}) \quad \text{since} \quad P(\boldsymbol{\theta}|\mathbf{x}) \propto P(\mathbf{x}|\boldsymbol{\theta})P(\boldsymbol{\theta}) \tag{3.14}$$

As can be seen from (3.14) the maximum a posteriori parameter estimate is equal to the maximum likelihood estimate when the prior distribution is uniform, i.e. $P(\boldsymbol{\theta})$ is constant.

3.8.1 Maximum likelihood for the returns model

The observed data consists of the observed times to return $t_i^{(o)}, i = 1 \dots m$, collected in the vector $\mathbf{t}^{(o)}$, and the knowledge that $T_i^{(u)} > t - s_i,\ i = m+1 \dots n$, where t is the current time. This can be written in vector form as $\mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}$ where $\mathbf{s}$ is a vector containing all sales times $s_i,\ i = m+1, \dots, n$ and $\mathbf{1}$ is a vector consisting of all ones. This gives the likelihood for the observed data

$$\begin{aligned}\mathcal{L}(p, \boldsymbol{\theta} | \mathbf{t}^{(o)}, \mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}) &= \prod_{i=1}^{m} p \cdot \rho(t_i^{(o)} | \boldsymbol{\theta}) \prod_{i=m+1}^{n} (1 - p + p(1 - R(t - s_i | \boldsymbol{\theta}))) = \\ &= p^m \prod_{i=1}^{m} \rho(t_i^{(o)} | \boldsymbol{\theta}) \prod_{i=m+1}^{n} (1 - pR(t - s_i | \boldsymbol{\theta}))\end{aligned} \tag{3.15}$$

where R is the CDF of ρ.

Finding the parameters $p, \boldsymbol{\theta}$ that maximize this likelihood is a problem that generally can not be solved analytically. It is possible to find a closed form expression for the p that maximizes this likelihood in the case of equal sales times s_i for all products. Since this is never the case in real data numerical methods is required to the find the MLE.

Maximizing (3.15) can be done by using standard numerical optimization techniques such as the Newton-Raphson method or Quasi-Newton methods. The problem of using Newton-Raphson is that unless one has an accurate initial value for the parameters the method has the tendency to find saddle points and local minimi as often as local maxima [21].

Quasi-Newton methods fix this problem by approximating the Hessian and forcing it to always be negative definite. However Quasi-Newton methods can still be problematic to use in statistical applications since the initial approximation, often an identity matrix, may be badly scaled to the problem. This could lead to the algorithm overshooting or undershooting the maximum of the objective function in the current step direction [17].

We have chosen to solve the problem of finding the MLE estimates by using the EM-algorithm [7]. The EM-algorithm is commonly applied to problems with incomplete-data, e.g. censored data. But the EM-algorithm is also suitable in cases where the incompleteness of the data is not due to censoring [21]. An example of this is mixture distributions where the incomplete data corresponds to some variable indicating which distribution the observed value originates from. Hence the EM-algorithm is particularly well suited to our problem, we have incomplete-data due to censoring as well as a mixture distribution.

3.9 Expectation Maximization

The EM-algorithm is an iterative method for finding maximum likelihood estimates of parameters of a statistical model given incomplete data. The EM-algorithm maximizes the quantity [10]

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j)}) = \mathbb{E}\left[\log \mathcal{L}(\boldsymbol{\theta}|\mathbf{Y})|\mathbf{x}, \boldsymbol{\theta}^{(j)}\right] \tag{3.16}$$

where $\boldsymbol{\theta}^{(j)}$ is the estimated maximizer for iteration $j, (j = 0, 1, \ldots)$. The variable $\mathbf{Y}$ contains all the data i.e. all the observed and unobserved data. $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j)})$ is the expected value of the log likelihood for all of the data conditioned on the observed data $\mathbf{x}$. The EM-algorithm is initiated by choosing $\boldsymbol{\theta}^{(0)}$ and then alternates between an **E**xpectation and a **M**aximization step.

1. **E step:** Compute $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j)})$ using the observed data and best guess of the parameters so far
2. **M step:** Update parameter guess by maximizing $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j)})$ with respect to $\boldsymbol{\theta}$ i.e. $\boldsymbol{\theta}^{(j+1)} = \operatorname{argmax}_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j)})$
3. Return to step 1 unless a stopping criterion has been met.

An iteration of the EM-algorithm is guaranteed to increase the likelihood, see McLachlan and Krishan [21]

$$\mathcal{L}(\boldsymbol{\theta}^{(j+1)}|\mathbf{x}) \geq \mathcal{L}(\boldsymbol{\theta}^{(j)}|\mathbf{x}) \tag{3.17}$$

A normal stopping criterion that could be used is to stop the iterations when $\boldsymbol{\theta}^{(j+1)} - \boldsymbol{\theta}^{(j)}$ is sufficiently small. This could, however, lead to long computational times and may be combined with a maximum number of steps allowed.

3.9.1 EM-algorithm with a general distribution

To use the EM-algorithm we need the likelihood function for the complete data. This consists of the observed times to return $\mathbf{t}^{(o)}$, the unobserved conditional times to return $t_i^{(u)}, i = m+1 \ldots n$, collected in the vector $\mathbf{t}^{(u)}$, and the latent variables $z_i, i = 1, \ldots, n$, collected in the vector $\mathbf{z}$. The latent variables z_i take the value 1 if product i is ever returned and 0 otherwise which means that $z_i = 1,\ i = 1, \ldots, m$. The complete likelihood function $\mathcal{L}(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{t}^{(u)}, \mathbf{z})$ becomes the following

$$\begin{aligned}\mathcal{L}(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{t}^{(u)}, \mathbf{z}) &= \prod_{i=1}^{n} p^{z_i} \cdot \rho(t_i|\boldsymbol{\theta})^{z_i}(1-p)^{(1-z_i)} = \\ &= p^m \prod_{i=1}^{m} \rho(t_i^{(o)}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} p^{z_i} \cdot \rho(t_i^{(u)}|\boldsymbol{\theta})^{z_i}(1-p)^{(1-z_i)}\end{aligned} \tag{3.18}$$

Taking the logarithm yields the log-likelihood function

$$\ell(p,\boldsymbol{\theta}|\mathbf{t}^{(o)},\mathbf{t}^{(u)},\mathbf{z}) = \log\mathcal{L}(p,\boldsymbol{\theta}|\mathbf{t}^{(o)},\mathbf{t}^{(u)},\mathbf{z}) = m\log(p) + \sum_{i=1}^{m}\log(\rho(t_i^{(o)}|\boldsymbol{\theta}))+$$
$$+\log(p)\sum_{i=m+1}^{n} z_i + \sum_{i=m+1}^{n} z_i\log(\rho(t_i^{(u)}|\boldsymbol{\theta})) + \log(1-p)\sum_{i=m+1}^{n}(1-z_i). \quad (3.19)$$

To calculate the Q function in (3.16) we replace the missing and latent data by stochastic variables $T_i^{(u)}$ and Z_i, collected in the vectors $\mathbf{T}^{(u)}$ and $\mathbf{Z}$, and take the expected value of $\ell(p,\boldsymbol{\theta}|\mathbf{t}^{(o)},\mathbf{T}^{(u)},\mathbf{Z})$ with respect to $\mathbf{T}^{(u)}$ and $\mathbf{Z}$ given the observed data.

$$Q(p,\boldsymbol{\theta}|p^{(j)},\boldsymbol{\theta}^{(j)}) = \mathbb{E}\left[\ell(p,\boldsymbol{\theta})|\,\mathbf{t}^{(o)},\mathbf{T}^{(u)} > t\cdot\mathbf{1}-\mathbf{s},p^{(j)},\boldsymbol{\theta}^{(j)}\right] =$$
$$= m\log(p) + \sum_{i=1}^{m}\log(\rho(t_i^{(o)}|\boldsymbol{\theta})) + \log(p)\sum_{i=m+1}^{n}\mathbb{E}[Z_i|\mathbf{t}^{(o)},\mathbf{T}^{(u)} > t\cdot\mathbf{1}-\mathbf{s},p^{(j)},\boldsymbol{\theta}^{(j)}]+$$
$$+\sum_{i=m+1}^{n}\mathbb{E}[Z_i\log(\rho(T_i^{(u)}|\boldsymbol{\theta}))|\mathbf{t}^{(o)},\mathbf{T}^{(u)} > t\cdot\mathbf{1}-\mathbf{s},p^{(j)},\boldsymbol{\theta}^{(j)}]+$$
$$+\log(1-p)\sum_{i=m+1}^{n}(1-\mathbb{E}[Z_i|\mathbf{t}^{(o)},\mathbf{T}^{(u)} > t\cdot\mathbf{1}-\mathbf{s},p^{(j)},\boldsymbol{\theta}^{(j)}]) \quad (3.20)$$

From now on we will denote the conditionary set $\left\{\mathbf{t}^{(o)},\mathbf{T}^{(u)} > t\cdot\mathbf{1}-\mathbf{s},p^{(j)},\boldsymbol{\theta}^{(j)}\right\}$ by $\bullet$ to make the calculations easier to follow. The term $\mathbb{E}[Z_i|\bullet]$, which we from now on denote by π_i, can be evaluated using the cumulative distribution function R of ρ, the current time t and the sales time s_i of the i:th unit. The probability that one has not yet seen a return of unit i is a sum of two probabilities. Either the unit will never be returned, probability $1-p$ or it will be returned after time t, with probability $p\cdot(1-R(t-s_i))$. Hence the expected value is given by

$$\pi_i^{(j)} = \mathbb{E}[Z_i|\bullet] = 1\cdot P(Z_i=1|\bullet) + 0\cdot P(Z_i=0|\bullet) = P(Z_i=1|\bullet). \quad (3.21)$$

Using the definition of conditional probability and that the units' times to return are independent gives

$$\pi_i^{(j)} = P(Z_i=1|\mathbf{t}^{(o)},\mathbf{T}^{(u)} > t\cdot\mathbf{1}-\mathbf{s},p^{(j)},\boldsymbol{\theta}^{(j)}) =$$
$$= P(Z_i=1|T_i^{(u)} > t-s_i,p^{(j)},\boldsymbol{\theta}^{(j)}) = \frac{P(Z_i=1,T_i^{(u)} > t-s_i|p^{(j)},\boldsymbol{\theta}^{(j)})}{P(T_i^{(u)} > t-s_i|p^{(j)},\boldsymbol{\theta}^{(j)})} =$$
$$= \frac{P(T_i^{(u)} > t-s_i|Z_i=1,p^{(j)},\boldsymbol{\theta}^{(j)})P(Z_i=1|p^{(j)},\boldsymbol{\theta}^{(j)})}{\sum_{k=0}^{1}P(T_i^{(u)} > t-s_i|Z_i=k,p^{(j)},\boldsymbol{\theta}^{(j)})P(Z_i=k|p^{(j)},\boldsymbol{\theta}^{(j)})} =$$
$$= \frac{(1-R(t-s_i|\boldsymbol{\theta}^{(j)}))p^{(j)}}{1-p^{(j)}+p^{(j)}(1-R(t-s_i|\boldsymbol{\theta}^{(j)}))}. \quad (3.22)$$

The second to last term in (3.20) can be evaluated using the law of total expectation

$$\begin{aligned}
&\mathbb{E}[Z_i \log(\rho(T_i^{(u)}|\boldsymbol{\theta}))|\bullet] = \mathbb{E}\left[\mathbb{E}[Z_i \log(\rho(T_i^{(u)}|\boldsymbol{\theta}))|\bullet, Z_i]\middle|\bullet\right] = \\
&= \sum_{k=0}^{1} k \cdot \mathbb{E}[\log(\rho(T_i^{(u)}|\boldsymbol{\theta}))|\bullet, Z_i = k] P(Z_i = k|\bullet) = \\
&= P(Z_i = 1|\bullet)\mathbb{E}[\log(\rho(T_i^{(u)}|\boldsymbol{\theta}))|Z_i = 1, \bullet] = \pi_i^{(j)} \cdot \mathbb{E}[\log(\rho(T_i^{(u)}|\boldsymbol{\theta}))|Z_i = 1, \bullet].
\end{aligned} \tag{3.23}$$

We want to find the value of p that maximizes Q, this can be done analytically by differentiating (3.20) with respect to p which gives

$$\begin{aligned}
\frac{\partial Q(p, \boldsymbol{\theta}|p^{(j)}, \boldsymbol{\theta}^{(j)})}{\partial p} &= \frac{m}{p} + \frac{1}{p}\sum_{i=m+1}^{n} \pi_i^{(j)} + \frac{m-n}{1-p} + \frac{1}{1-p}\sum_{i=m+1}^{n} \pi_i^{(j)} = \\
&= \frac{1}{1-p}\left(\frac{1}{p}\left(m + \sum_{i=m+1}^{n} \pi_i^{(j)}\right) - n\right)
\end{aligned} \tag{3.24}$$

Setting this equal to zero and solving for p gives

$$\frac{1}{1-p}\left(\frac{1}{p}\left(m + \sum_{i=m+1}^{n} \pi_i^{(j)}\right) - n\right) = 0 \implies p = \frac{m}{n} + \frac{1}{n}\sum_{i=m+1}^{n} \pi_i^{(j)} \tag{3.25}$$

i.e. the updating formula for p is

$$p^{(j+1)} = \frac{m}{n} + \frac{1}{n}\sum_{i=m+1}^{n} \pi_i^{(j)} \tag{3.26}$$

Differentiating (3.20) with respect to $\boldsymbol{\theta}$ and setting equal to zero yields

$$\begin{aligned}
\frac{\partial Q(p, \boldsymbol{\theta}|p^{(j)}, \boldsymbol{\theta}^{(j)})}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{m} \frac{\partial \log(\rho(t_i^{(o)}|\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}} + \\
+ \sum_{i=m+1}^{n} \pi_i^{(j)} \cdot \mathbb{E}\left[\frac{\partial \log(\rho(T_i^{(u)}|\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}}\middle| Z_i = 1, \bullet\right] = 0
\end{aligned} \tag{3.27}$$

which can not be solved analytically for all distributions ρ.

3.9.2 Negative binomial distribution

The negative binomial distribution is a discrete distribution of the number of failures in a sequence of independent and identically distributed (iid) Bernoulli trials with probability q of success before r successes have occurred. It is therefore easy to see that a special case of the negative binomial distribution is the geometric distribution and that it is obtained with $r = 1$ and $p = 1 - q$.

The negative binomial (NB) distribution is defined by the following equations. [9]

$$\begin{aligned} & T \in \text{NB}(r,q), \quad r>0, \quad q \in [0,1] \\ \text{pmf:} \quad & P(T=k) = \begin{cases} \binom{k+r-1}{k}(1-q)^r q^k & \text{if } r \in \mathbb{Z} \\ \frac{\Gamma(k+r)}{k!\,\Gamma(r)}(1-q)^r q^k & \text{if } r \in \mathbb{R} \end{cases} \\ \text{CDF:} \quad & P(T \leq k) = 1 - I_q(k+1,r) \end{aligned} \tag{3.28}$$

where I_q is the regularized incomplete beta function defined as

$$I_q(a,b) = \frac{B(q;a,b)}{B(1;a,b)}, \quad B(q;a,b) = \int_0^q t^{a-1}(1-t)^{b-1}dt. \tag{3.29}$$

and

$$\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx, \quad \Gamma(z+1) = z\Gamma(z). \tag{3.30}$$

See the appendix A.1 for proof that the regularized incomplete beta function can be used to calculate the CDF of a negative binomial distribution.

It is not possible to obtain an expression for all parameters when using a negative binomial distribution. We can get p from (3.26) but an expression for q is possible only if r is fixed. In the following calculations we therefore assume that r is fixed i.e. $r^{(j+1)} = r^{(j)}$.

The expression for π_i when using a negative binomial distribution can be obtained from (3.22) and (3.28) yielding

$$\pi_i^{(j)} = P(Z_i = 1|\bullet) = \frac{p^{(j)} I_{q^{(j)}}(t - s_i + 1, r^{(j)})}{1 - p^{(j)} + p^{(j)} I_{q^{(j)}}(t - s_i + 1, r^{(j)})} \tag{3.31}$$

and inserting this into (3.26) gives the updating expression for p.

The updating formula for q can be calculated by using (3.27) with a NB(r,q) distribution.

$$\begin{aligned} \frac{\partial Q(p,q|p^{(j)},q^{(j)},r^{(j)})}{\partial q} = -\frac{r^{(j)}\,m}{1-q} + \frac{1}{q}\sum_{i=1}^{m} t_i^{(o)} - \frac{r^{(j)}}{1-q}\sum_{i=m+1}^{n}\pi_i^{(j)} + \\ + \frac{1}{q}\sum_{i=m+1}^{n}\pi_i^{(j)} \cdot \mathbb{E}\left[T_i^{(u)}\middle| Z_i = 1, \bullet\right] = 0 \end{aligned} \tag{3.32}$$

We define the expected time to return given that the unit not yet has been returned and will be returned as

$$\begin{aligned} & \mu_i^{(j)} = \mathbb{E}\left[T_i^{(u)}\middle| Z_i = 1, \bullet\right] = \mathbb{E}\left[T_i^{(u)}|Z_i = 1, T_i^{(u)} > t - s_i, p^{(j)}, q^{(j)}, r^{(j)}\right] = \\ & = \sum_{l=0}^{\infty} l \cdot P(T_i^{(u)} = l | Z_i = 1, T_i > t - s_i, p^{(j)}, q^{(j)}, r^{(j)}) = \\ & = \sum_{l=0}^{\infty} l \cdot \frac{P(T_i^{(u)} = l, T_i > t - s_i | Z_i = 1, p^{(j)}, q^{(j)}, r^{(j)})}{P(T_i > t - s_i | Z_i = 1, p^{(j)}, q^{(j)}, r^{(j)})} \end{aligned} \tag{3.33}$$

Inserting the negative binomial distribution (3.28) and simpifying the sum (see appendix A.2) gives

$$\begin{aligned}&\frac{1}{I_{q^{(j)}}(t-s_i+1,r^{(j)})}\sum_{l=t-s_i+1}^{\infty} l\cdot\frac{\Gamma(l+r^{(j)})(1-q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})}=\\&=\frac{r^{(j)}q^{(j)}}{(1-q^{(j)})I_{q^{(j)}}(t-s_i+1,r^{(j)})}\begin{cases}I_{q^{(j)}}(t-s_i,r^{(j)}+1), & \text{if } t-s_i>0\\ 1, & \text{if } t-s_i=0.\end{cases}\end{aligned} \tag{3.34}$$

Solving (3.32) for q yields the following updating formula

$$q^{(j+1)}=\frac{\sum_{i=1}^{m}t_i^{(o)}+\sum_{i=m+1}^{n}\mu_i^{(j)}\cdot\pi_i^{(j)}}{r^{(j)}m+\sum_{i=1}^{m}t_i^{(o)}+\sum_{i=m+1}^{n}(\mu_i^{(j)}+r^{(j)})\cdot\pi_i^{(j)}}. \tag{3.35}$$

As said before we have assumed that r is fixed such that $r^{(j+1)}=r^{(j)}$. If we want to let r vary between iterations a different method is suitable to maximize $Q(p,q,r|p^{(j)},q^{(j)},r^{(j)})$. Two such methods are Expectation Conditional Maximization and Expectation Conditional Maximization Either, see Sections 3.9.4 and 3.9.5.

3.9.3 Weibull distribution

The Weibull distribution was named after Waloddi Weibull who investigated it in detail in 1951. It is a continuous distribution which is often used to describe the failure times of different kinds of components. The distribution is defined by two parameters λ and k known as the shape and scale parameters, respectively. A special case of the Weibull distribution is obtained with $k=1$ which makes it reduce to an exponential distribution. For values where $k<1$ the probability density function is monotonically decreasing and if $k>1$ the density function is first increasing and then decreasing.

The Weibull (Wbl) distribution is defined by the following equations. [16]

$$\begin{aligned}&T\in\text{Wbl}(\lambda,k),\quad \lambda>0,\quad k>0\\ \text{PDF:}\quad &f_T(t)=\begin{cases}\frac{k}{\lambda}\left(\frac{t}{\lambda}\right)^{k-1}e^{-(t/\lambda)^k} & \text{if } t\geq 0\\ 0 & \text{if } t<0\end{cases}\\ \text{CDF:}\quad &F_T(t)=\begin{cases}1-e^{-(t/\lambda)^k} & \text{if } t\geq 0\\ 0 & \text{if } t<0\end{cases}\end{aligned} \tag{3.36}$$

Similarly to the case with a negative binomial distribution, it is not possible to obtain an expression for all parameters when using a Weibull distribution so we therefore let the k parameter be fixed i.e. $k^{(j+1)}=k^{(j)}$.

An expression for $\pi_i^{(j)}$ with a Weibull distribution is obtained by using (3.36) and (3.22) which gives

$$\pi_i^{(j)}=P(Z_i=1|\bullet)=\frac{p^{(j)}e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}}{1-p^{(j)}+p^{(j)}e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}}. \tag{3.37}$$

Using this $\pi_i^{(j)}$ in (3.26) yields the updating formula for p.

To calculate λ we use (3.27) and (3.36) which gives

$$\frac{\partial Q(p,\lambda|p^{(j)},\lambda^{(j)},k^{(j)})}{\partial\lambda} = -\frac{k^{(j)}\,m}{\lambda} + \frac{k^{(j)}}{\lambda^{k^{(j)}+1}}\sum_{i=1}^{m}(t_i^{(o)})^{k^{(j)}} - \frac{k^{(j)}}{\lambda}\sum_{i=m+1}^{n}\pi_i^{(j)} + \\ + \frac{k^{(j)}}{\lambda^{k^{(j)}+1}}\sum_{i=m+1}^{n}\pi_i^{(j)}\cdot\mathbb{E}\left[(T_i^{(u)})^{k^{(j)}}\,\middle|\,Z_i=1,\bullet\right] = 0 \quad (3.38)$$

We define the expected value of the time to return to the power of $k^{(j)}$ given that the unit has not yet been returned and will be returned as

$$\begin{aligned}\mu_i^{(j)} &= \mathbb{E}\left[\left(T_i^{(u)}\right)^{k^{(j)}}\,\middle|\,Z_i=1,\bullet\right] = \\ &= \int_0^\infty \tau^{k^{(j)}}\cdot P(T_i=\tau|Z_i=1,T_i>t-s_i,p^{(j)},k^{(j)},\lambda^{(j)})d\tau = \\ &= \int_0^\infty \tau^{k^{(j)}}\cdot\frac{P(T_i=\tau,T_i>t-s_i|Z_i=1,p^{(j)},k^{(j)},\lambda^{(j)})}{P(T_i>t-s_i|Z_i=1,p^{(j)},k^{(j)},\lambda^{(j)})}d\tau = \\ &= \int_{t-s_i}^\infty \tau^{k^{(j)}}\cdot\frac{k^{(j)}\tau^{k^{(j)}-1}e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}}e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}}d\tau\end{aligned} \quad (3.39)$$

The calculation of this integral is shown in appendix A.3 and the result is

$$\mu_i^{(j)} = \int_{t-s_i}^\infty \tau^{k^{(j)}}\cdot\frac{k^{(j)}\tau^{k^{(j)}-1}e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}}e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}}d\tau = (t-s_i)^{k^{(j)}} + (\lambda^{(j)})^{k^{(j)}} \quad (3.40)$$

The updating formula for λ now becomes

$$\lambda^{(j+1)} = \left(\frac{\sum_{i=1}^m (t_i^{(o)})^{k^{(j)}} + \sum_{i=m+1}^n \mu_i^{(j)}\cdot\pi_i^{(j)}}{m+\sum_{i=m+1}^n \pi_i^{(j)}}\right)^{1/k^{(j)}}. \quad (3.41)$$

If we also want k to vary we must use other algorithms, similarly to r for the negative binomial distribution.

3.9.4 Expectation Conditional Maximization

One of the advantages of the EM-algorithm is that the maximization step only involves the complete likelihood. This likelihood is in most cases simple and the maximization can be performed analytically. However there exist cases where the maximization of the complete likelihood is complicated and can not be done analytically. One must then either maximize the complete likelihood numerically or abandon the EM-algorithm and try to maximize the observed likelihood directly using some other numerical optimization method.

In many of these complicated cases it is possible to maximize the complete likelihood analytically when conditioned on some function of the other parameters. Meng and Rubin [22] developed the Expectation Conditional Maximization (ECM) algorithm which utilizes that conditional maximization is often much simpler and can be done

analytically. This algorithm can be seen as a special case of the cyclic coordinate ascent method used in numerical optimization [21].

Formally the M-step in the EM-algorithm is replaced by B conditional maximization steps. If $\boldsymbol{\theta}^{(j+b/B)}$ denotes the value of $\boldsymbol{\theta}$ on the b:th conditional maximization step of the $j+1$:th iteration then $\boldsymbol{\theta}^{(j+b/B)}$ is chosen as

$$\boldsymbol{\theta}^{(j+b/B)} = \underset{\theta}{\operatorname{argmax}}\, Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j)}) \quad \text{subject to} \quad \mathbf{g}_b(\boldsymbol{\theta}) = \mathbf{g}_b(\boldsymbol{\theta}^{(j+(b-1)/B)}) \tag{3.42}$$

where C = $\{\mathbf{g}_b(\boldsymbol{\theta}), b = 1, \ldots, B\}$ is a set of preselected vector functions. [21, 22]

If $\boldsymbol{\theta}$ is divided into B subvectors $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1^{\mathrm{T}}, \ldots, \boldsymbol{\theta}_B^{\mathrm{T}}\}^{\mathrm{T}}$ then it is in many applications natural to let the b:th conditional maximization step be the maximization of $\boldsymbol{\theta}_b$ with the other $B-1$ subvectors fixed at their current values [21]. In this case $\mathbf{g}_b$ is a vector containing all subvectors except $\boldsymbol{\theta}_b$.

Meng and Rubin [22] prove that the ECM algorithm preserves all the desirable properties of the EM-algorithm, e.g. the likelihood is always non-decreasing with each iteration i.e.

$$\mathcal{L}(\boldsymbol{\theta}^{(j+1)}|\mathbf{x}) \geq \mathcal{L}(\boldsymbol{\theta}^{(j)}|\mathbf{x}) \tag{3.43}$$

3.9.5 Expectation Conditional Maximization Either

The Expectation Conditional Maximizatoin Either (ECME) algorithm developed by Liu and Rubin [18] is essentially the same algorithm as the ECM-algorithm. The difference is that in some of the conditional maximization steps the actual observed likelihood is maximized instead of the complete likelihood. The monotone convergence property (3.43) also holds for the ECME algorithm. But only if the conditional maximization steps that act on the Q-function (Expectation of the complete likelihood) are performed before the steps that act on the observed likelihood [21].

3.9.6 Using ECM and ECME

The ECM and ECME algorithm allow us to maximize the likelihood with respect to all the distribution parameters for the distributions (Negative binomial and Weibull).

As we discussed earlier in section 3.9.2 the problem with having a varying r is that the expression for q depends on r. Thus the Q-function would have to be maximized with respect to both q and r for which it would not be possible to find an analytic expression. Hence this would have to be done numerically in each iteration. By using the ECM-algorithm we can maximize with respect to q conditional on $r = r^{(j)}$, thus we still use the analytic expression for $q^{(j+1)}$ (3.35) as in the case when r is constant.

Unfortunately it is not possible to derive an expression for $r^{(j+1)}$, even when conditioned on $p = p^{(j+1)}$, $q = q^{(j+1)}$ using (3.42). Another problem is that the expectation of $\log(\Gamma(T_i^{(u)} + r))$ must be calculated numerically by summation for each new r. This is not a problem when maximizing Q with respect to p or when maximizing Q with respect to q conditional on $r = r^{(j)}$ since this term vanishes when

differentiating. However this term must be calculated numerically when maximizing Q with respect to r which would take too much time. But using the ECME-algorithm solves the problem. In the third conditional maximization step where we maximize Q with respect to r conditioned on $p = p^{(j+1)},\ q = q^{(j+1)}$ we simply instead maximize the observed likelihood with respect to r and again conditioned on $p = p^{(j+1)},\ q = q^{(j+1)}$. This must still be done numerically but we evade the intractable expectation of $Z_i \log(\Gamma(T_i^{(u)}+r))$. The problem of maximizing the observed likelihood with respect to r holding p and q constant is simply a one-dimensional search and hence can be solved without too much trouble despite the complexity of the observed likelihood.

Similarily with the Weibull distribution as discussed in section 3.9.3 the expression for λ depends on k which makes it invalid in the case of a varying k. Also the expression for μ_i derived in (3.39) is only valid when $k = k^{(j)}$. As discussed in section 3.9.3 this holds when k is constant. But by using the ECM and ECME algorithms it is possible for this to hold even for varying k. By simply performing the maximization of Q with respect to λ before maximizing with respect to k makes both (3.39) and (3.41) valid since the maximization with respect to λ is then performed conditioned on $k = k^{(j)}$.

Similar to the negative binomial distribution it is not possible to derive an expression for $k^{(j+1)}$. It is only possible to write the expectation of $(T_i^{(u)})^k$ on closed form when $k = k^{(j)}$, see (3.39), thus this would have to be calculated numerically for each new k. Therefore it is much more efficient to use the ECME-algorithm and replace the Q-function in the last conditional maximization step with the observed likelihood as was done with the negative binomial distribution. Maximizing this numerically with respect to k holding p and λ constant is again only a simple one-dimensional search.

3.9.7 Accelerating the EM-algorithm

One problem with the EM-algorithm is that the convergence can be quite slow. This can be remedied by acceleration of the EM-algorithm. As pointed out by Lange [17] it is likely that no accelerated version of the EM-algorithm can match the stability and simplicity of the original EM-algorithm.

We have chosen to implement an acceleration method proposed by Louis [19]. The algorithm proposed by Louis is similar to the Aitken acceleration method [21]. This method first produces a parameter estimation $\boldsymbol{\theta}_{EMA}^{(j+1)}$ using the EM-algorithm and the previous parameter values $\boldsymbol{\theta}_A^{(k)}$. The parameter estimation $\boldsymbol{\theta}_{EMA}^{(j+1)}$ produced by the EM-algorithm is then used in an acceleration formula to obtain the final parameter estimation $\boldsymbol{\theta}_A^{(j+1)}$ for the $j+1$:th iteration.

The formula for $\boldsymbol{\theta}_A^{(j+1)}$ using the Aitken acceleration method is given by

$$\boldsymbol{\theta}_A^{(j+1)} = \boldsymbol{\theta}_A^{(j)} + (I - J(\boldsymbol{\theta}_A^{(j)}))(\boldsymbol{\theta}_{EMA}^{(j+1)} - \boldsymbol{\theta}_A^{(j)}) \tag{3.44}$$

where I is the identity matrix and $J(\boldsymbol{\theta})$ is the Jacobian of the map from $\boldsymbol{\theta}^{(j)}$ to $\boldsymbol{\theta}^{(j+1)}$. The Jacobian of this map is intractable to calculate analytically in many cases. Louis

[19] proposes estimating the Jacobian by using the following

$$J(\boldsymbol{\theta}^{(j)}) = I - Q''(\boldsymbol{\theta}^{(j)})^{-1}\mathcal{L}''(\boldsymbol{\theta}^{(j)}) \tag{3.45}$$

where $\mathcal{L}''$ is the Hessian of the observed likelihood

$$\mathcal{L}''(\boldsymbol{\theta}|X) = \frac{\partial^2 \mathcal{L}(\boldsymbol{\theta}|X)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^{\mathrm{T}}} \tag{3.46}$$

and Q'' is the hessian of the expected value of the complete likelihood

$$Q''(\boldsymbol{\theta}) = \frac{\partial^2 Q(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^{\mathrm{T}}} \tag{3.47}$$

This gives the updating formula for $\boldsymbol{\theta}$ as

$$\boldsymbol{\theta}_A^{(j+1)} = \boldsymbol{\theta}_A^{(j)} + \mathcal{L}''(\boldsymbol{\theta}_A^{(j)})^{-1}Q''(\boldsymbol{\theta}_A^{(j)})(\boldsymbol{\theta}_{EMA}^{(j+1)} - \boldsymbol{\theta}_A^{(j)}) \tag{3.48}$$

This algorithm has the possibility of creating the effect of an infinite number of iterations [19].

The approximation proposed by Louis is only valid local to the MLE. Hence the acceleration method should not be used until some EM-iterations have been performed so that the parameter estimations when starting the acceleration are sufficiently close to the MLE.

It can be shown that the Louis' acceleration is exactly the same as applying the Newton-Raphson method to find the zero of the difference

$$\delta(\boldsymbol{\theta}) = M(\boldsymbol{\theta}) - \boldsymbol{\theta} \tag{3.49}$$

where $M(\boldsymbol{\theta})$ is the map from $\boldsymbol{\theta}^{(j)}$ to $\boldsymbol{\theta}^{(j+1)}$ obtained with the EM-algorithm [21]. This is precisely what we are trying to achieve with the EM-algorithm, i.e. we want to iterate until $\|\boldsymbol{\theta}^{(j+1)} - \boldsymbol{\theta}^{(j)}\| = \|M(\boldsymbol{\theta}^{(j)}) - \boldsymbol{\theta}^{(j)}\| \leq \epsilon$ where ϵ should be as close to zero as possible and optimally equal to zero.

Unfortunately it is not possible to use the acceleration when having a varying r in the negative binomial distribution and varying k in the Weibull distribution. This is because as explained in section 3.9.6 there are expectations dependent on r and k which are intractable to calculate when they are non-constant. Hence it is not possible to find $Q''(\boldsymbol{\theta})$ on closed form when r and k are varying.

3.10 Parameter variance estimation

3.10.1 Fisher information matrix

In the general sense, Fisher information is a way of measuring how much information is contained about a parameter in a data set. One of its use-cases is the ability to approximate variances for estimated parameters. This is interesting to us since we want to estimate how accurate our predictions of the return rate are.

Let X be a stochastic variable with PDF $f(x|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is a column vector of k parameters. We define

$$\lambda(x|\boldsymbol{\theta}) = \log(f(x|\boldsymbol{\theta})) \tag{3.50}$$

and assume that λ is twice differentiable with respect to $\boldsymbol{\theta}$. With a few assumptions on X and f, described in detail by [6], we can now define the Fisher information matrix $\mathcal{I}(\boldsymbol{\theta})$ as the $k \times k$ matrix with element on row i and column j as the following.

$$\mathcal{I}_{i,j}(\boldsymbol{\theta}) = \mathbb{E}\left[\left(\frac{\partial}{\partial\theta_i}\lambda(x|\boldsymbol{\theta})\right)\left(\frac{\partial}{\partial\theta_j}\lambda(x|\boldsymbol{\theta})\right)\middle|\,\boldsymbol{\theta}\right] \tag{3.51}$$

It is possible to show that, under certain regularity conditions, the elements of the Fisher information matrix may also be written as:

$$\mathcal{I}_{i,j}(\boldsymbol{\theta}) = -\mathbb{E}\left[\left(\frac{\partial^2}{\partial\theta_i\partial\theta_j}\lambda(x|\boldsymbol{\theta})\right)\middle|\,\boldsymbol{\theta}\right] \tag{3.52}$$

i.e. as minus the expectation of the Hessian of λ.

3.10.2 Estimating variance for maximum likelihood estimation

When solving a problem using maximum likelihood estimation it is important to know the accuracy of the solutions. This is where the Fisher information matrix is useful. Assume that $\hat{\boldsymbol{\theta}}$ is a maximum likelihood estimator for $\boldsymbol{\theta}$ with n samples. It can then be shown that under regular conditions $\hat{\boldsymbol{\theta}}$ will be approximately normally distributed with mean $\boldsymbol{\theta}$ and covariance matrix $(\mathcal{I}(\boldsymbol{\theta}))^{-1}$ for large values of n. In this case $\mathcal{I}(\boldsymbol{\theta})$ is calculated using (3.52) with the log-likelihood of the observed data $\ell(\boldsymbol{\theta}|\mathbf{x})$, where $\mathbf{x}$ is the observed data, instead of $\lambda(x|\boldsymbol{\theta})$. This means that the Fisher information matrix can be used to approximate the variances of the estimated parameters in a maximum likelihood estimation.

When actually trying to estimate the variance of a maximum likelihood estimation, it may not be optimal to use the standard Fisher information matrix. Instead, as argued by [8], it may be better to use what is commonly referred to as the observed Fisher information matrix $I(\boldsymbol{\theta})$. The matrix' elements are defined by

$$I_{i,j}(\boldsymbol{\theta}) = -\left(\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ell(\boldsymbol{\theta}|\mathbf{x})\right). \tag{3.53}$$

As is shown, the observed Fisher information matrix is the negative Hessian of the log-likelihood function ℓ. Obtaining approximations of the variances now simply requires us to insert the estimated parameter $\hat{\boldsymbol{\theta}}$ into the observed Fisher information matrix and taking the inverse. The estimated covariance matrix Σ for a maximum likelihood estimation can therefore be expressed as:

$$\Sigma = -\left[\left(\frac{\partial^2}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}\ell(\boldsymbol{\theta}|\mathbf{x})\right)^{-1}\right]_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \tag{3.54}$$

With the covariance matrix known it is simple to calculate confidence intervals for the estimated parameters $\hat{\boldsymbol{\theta}}$ since they are approximately normally distributed around $\boldsymbol{\theta}$.

Calculation of the Hessian of the observed log-likelihood for a general time to return distribution and the negative binomial and Weibull distribution are shown in appendix A.4.

3.11 Prior distributions

A prior distribution can be used to take the prior knowledge of the parameters into account, i.e. our beliefs of which parameter values are more likely before observing any data, as described in section 3.8. Since we had some prior knowledge of which lifetime return rates that were more likely it was appropriate to use a prior for the p-parameter. When introducing the EM-algorithm earlier the likelihood function was used but the EM-algorithm can also be used to find maximum a posteriori parameter estimates [4].

As can be seen from (3.18) the conjugate prior for p, i.e. the prior that gives the posterior distribution the same functional form as the prior, is the beta distribution. The PDF of the beta distribution is given by [9]

$$f_P(p) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(1;\alpha,\beta)} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1} \tag{3.55}$$

where the Gamma function $\Gamma(z)$ is defined in (3.30).

The posterior complete distribution for a general time to return distribution becomes

$$\begin{aligned}
&\mathcal{P}(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{t}^{(u)}, \mathbf{z}, \alpha, \beta) = \\
&= \prod_{i=1}^{n} p^{z_i} \cdot \rho(t_i|\boldsymbol{\theta})^{z_i}(1-p)^{(1-z_i)} \cdot \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1} = \\
&= p^m \prod_{i=1}^{m} \rho(t_i^{(o)}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} p^{z_i} \cdot \rho(t_i^{(u)}|\boldsymbol{\theta})^{z_i}(1-p)^{(1-z_i)} \cdot \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}.
\end{aligned} \tag{3.56}$$

By using a conjugate prior the functional form of the p dependency is not changed as mentioned earlier, hence the posterior complete distribution can still be maximized analytically. Taking the logarithm of (3.56) and differentiating with respect to p and solving the resulting equation, as done in section 3.9.1, results in the updating formula for p as

$$p^{(j+1)} = \frac{m + \sum_{i=m+1}^{n} \pi_i^{(j)} + \alpha - 1}{n + \alpha + \beta - 2}. \tag{3.57}$$

The beta distribution with the parameters set to $\alpha = 1, \beta = 1$ is equal to a uniform distribution on $[0, 1]$ and it can be seen that (3.57) is equal to (3.26) in that case. The parameters α and β of the prior distribution were selected by fitting a beta distribution to Axis' previous return rates. The updating formulas for the parameters $\boldsymbol{\theta}$ are not affected by the prior since the prior is only a function of p.

The updated variance calculation for a general time to return distribution when using a beta distribution prior on p is shown in appendix A.5.

When using the ECME-algorithm for finding the estimates of all the parameters, i.e. including r and k in the case of negative binomial and Weibull respectively, we used a prior on r and k to improve the prediction. Since r and k have to be positive it is reasonable to use a distribution for which the support of the PDF is the real positive axis. We decided to use the log-normal distribution for which the PDF is [9]

$$f_X(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-(\log(x)-\mu)^2/2\sigma^2} \tag{3.58}$$

where the parameters μ and σ were selected based on return times of previous products.

3.12 Measuring error

To be able to evaluate the performance of a method we need some measure of how well the lifetime return rate was predicted. There are multiple ways of constructing such error measures and the error measure that we decided to use is based on the absolute error between our predictions and the lifetime return rate. If our prediction of the return rate for period $i = 0 \dots T$, where T is the final period, is denoted $\hat{p}_i$ and the lifetime return rate is p then the error measure used is

$$e = \frac{1}{p(T+1)} \sum_{i=0}^{T} |p - \hat{p}_i|. \tag{3.59}$$

This can be interpreted as the area between the LRR and the predicted return rate divided by the area of the rectangle between the LRR and the t-axis as is shown in figure 3.4.

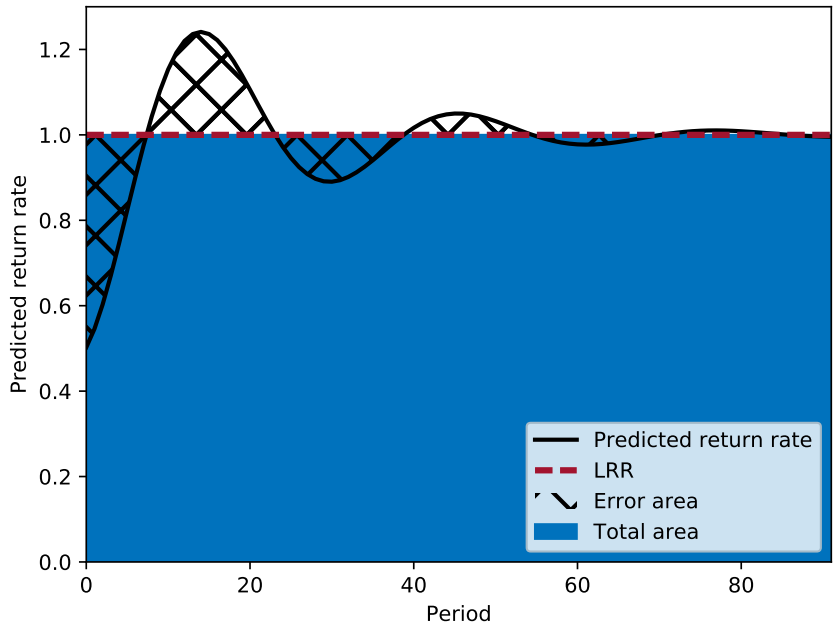

FIGURE 3.4: A plot of how the error can be interpreted as area.

4 Results

The products used for tests are called products A-H. In this chapter we first show the results of using different methods to predict the return rate on the same product (product A). Then we show the results of using different methods on products B-H. We evaluate the results and then apply the most successful method on the products B-H, shown in Appendix A.8. Each product's life span is divided into smaller time periods each of which is 30 days long and the return rate is calculated for each period until no more returns are obtained. The first period that the return rate is calculated for is the first period where a return occurred.

Unless stated otherwise, all plots have been scaled to make the lifetime return rate equal to 1. This has been done to avoid revealing business sensitive information.

In addition, for all results which used either the EM-algorithm or the ECME-algorithm we have used a prior on p unless stated otherwise. As explained in Section 3.11 we used a beta distribution prior with parameters selected by fitting a beta distribution to Axis' previous return rates.

4.1 Aggregated Return Rate

Using the aggregated return rate for estimating the lifetime return rate of product A yields the result shown in figure 4.1.

4.2 Kaplan-Meier Estimator

When using the last value of a Kaplan-Meier estimator as the predicted return rate, as described in section 3.4.1, we obtained the result shown in figure 4.2. The plot also shows the 95% confidence intervals for the predicted return rate. As can be seen in the plot, a problem with this method are the confidence intervals for the first few periods. These do not cover the LRR in the beginning since these confidence intervals are only valid for the last value of the Kaplan-Meier estimator and not the LRR.

4.3 Negative binomial distribution with constant r-parameter

A negative binomial distribution was used with the EM-algorithm to predict the return rate. When using the standard EM-algorithm we had to let the r-parameter be constant. We therefore first show the results of finding the optimal value of the r-parameter and then the results when using the optimal value for product A.

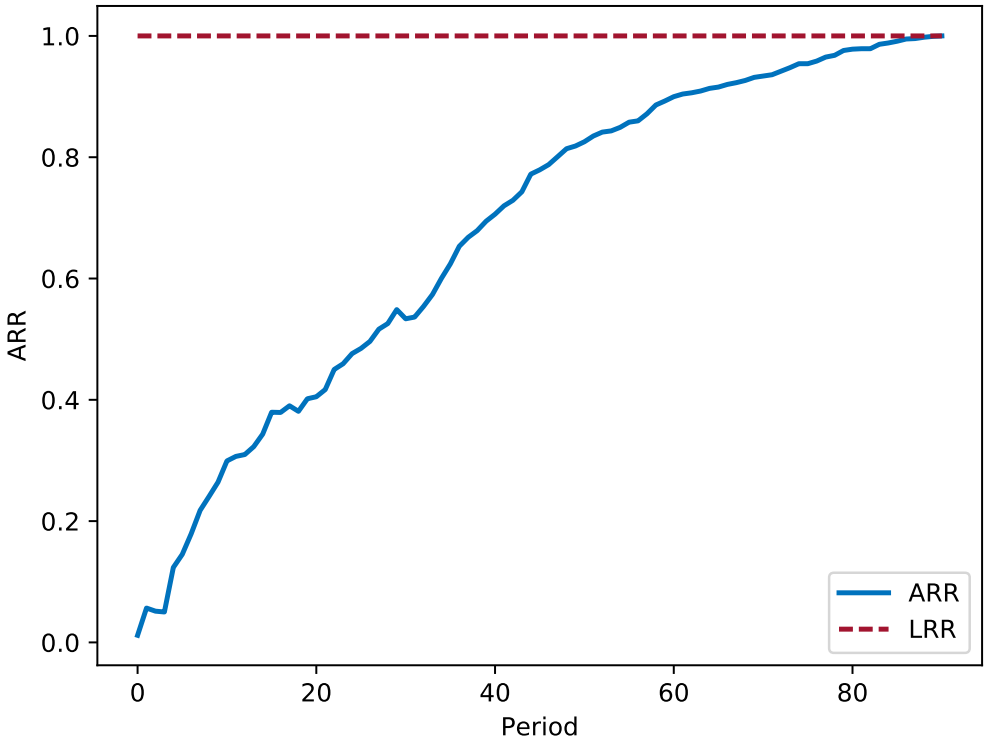

FIGURE 4.1: Aggregated return rate and lifetime return rate for product A

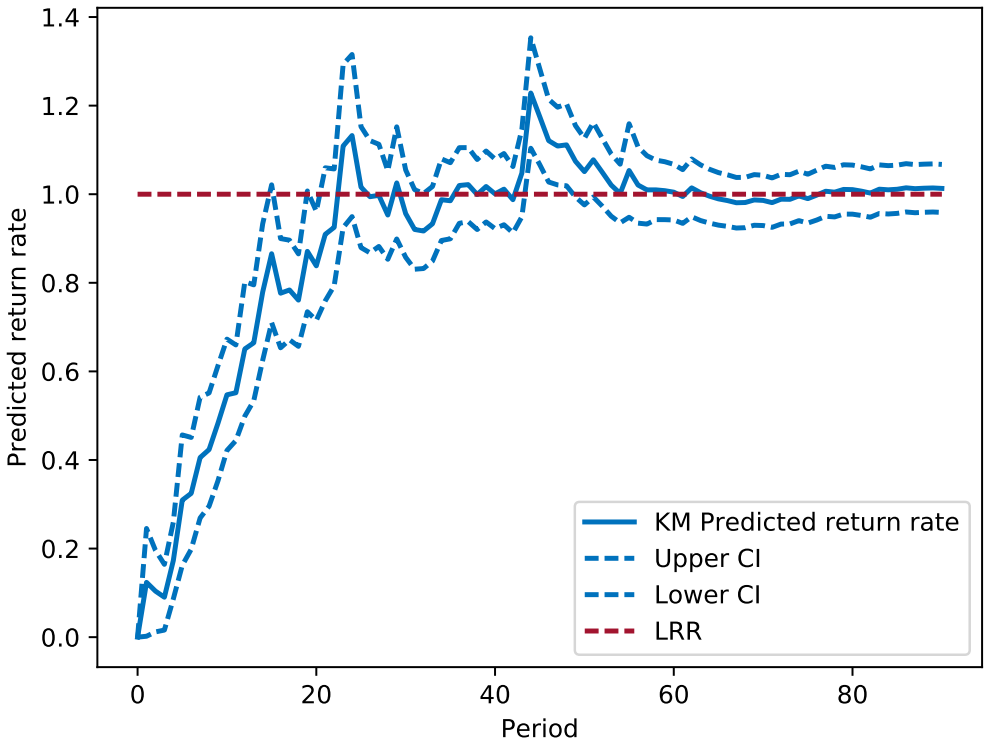

FIGURE 4.2: Predicted return rate for product A and confidence intervals using the last value of the Kaplan-Meier estimator.

4.3.1 Finding the optimal r-value

To find the optimal value of the r-parameter, tests were carried out on products B-H. This was done by starting on an r-value of 1.0 corresponding to a geometrical distribution and then running the EM-algorithm for each product with a lifetime return rate. Then the value of r was incremented by 0.05 and the procedure was repeated. The error value for each r was calculated by using our error measure for each product and then adding them together to form a single error value for each r. The results are shown in figure 4.3. From this analysis an optimal value of $r = 2.05$ was yielded which was used in the subsequent tests with constant r. The result of using the EM-algorithm with the negative binomial distribution with this value of r is shown in figure 4.4.

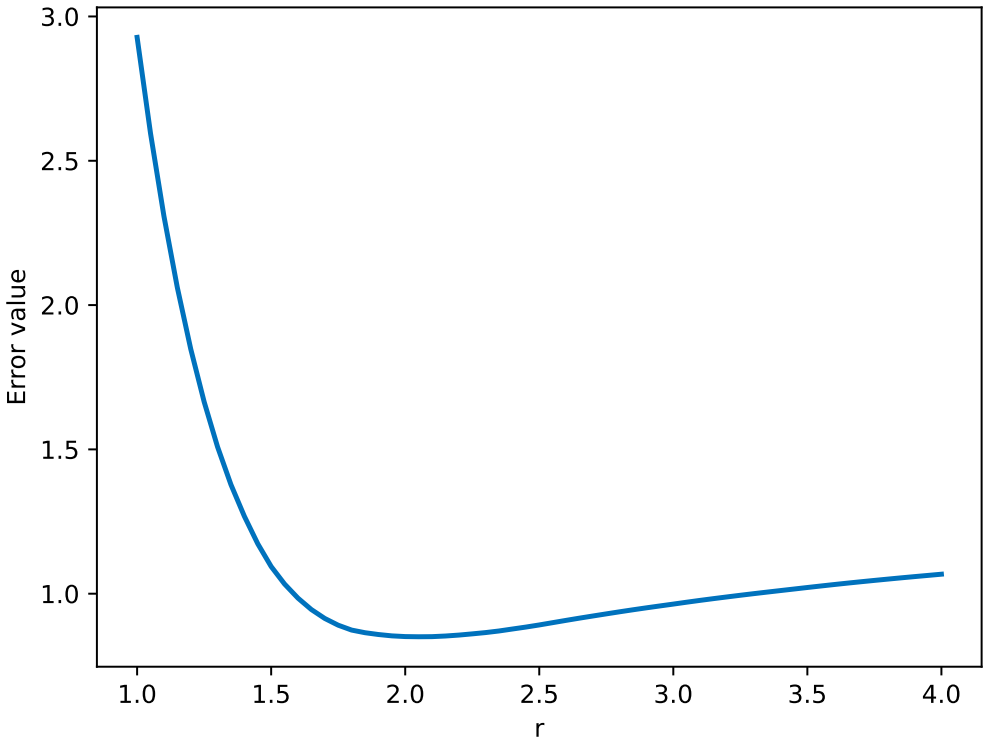

FIGURE 4.3: Measured error values for different constant values of r

4.4 Generated Data

We generated data to establish how quickly and accurately the ECME algorithm could find the correct parameters of a distribution. The data generated was supposed to emulate how real data could look for an Axis product. We therefore used a Poisson distribution with constant mean to simulate the number of sales per 30 day period and a return rate $p = 0.01$ and negative binomial distribution with parameters $r = 1.3$ and $q = 0.85$ to simulate the returns. For real products the number of sales per period does not in general follow a Poisson distribution with constant mean, rather the number of sales per period varies during the product's lifetime.

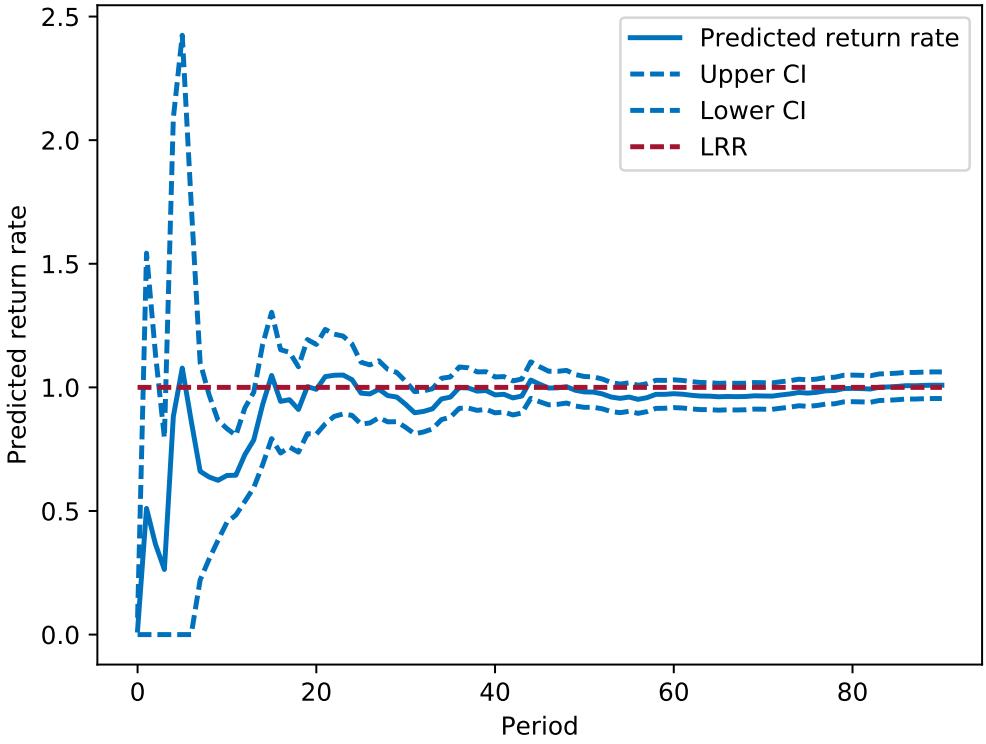

FIGURE 4.4: Predicted return rate for product A using the EM-algorithm with negative binomial distribution with constant r together with approximate 95% confidence intervals.

Usually the number of sales per period increases for a few years and then it decreases, but modeling the sales with a constant mean appears to be sufficiently accurate.

In figures 4.5 - 4.7 the results from running the ECME algorithm with a negative binomial distribution without a prior on p for 20 different generated data sets are shown. In the figures we see that all parameters are correctly estimated in later periods when there is plenty of data. However, as can be seen in figure 4.5 the return rate is heavily overestimated in the beginning periods when there is an insufficient amount of data. To remedy this, we applied the prior on p which was fitted for Axis' products, as described previously, and the results are shown in figures 4.8 - 4.10. With a prior on p the estimations for the predicted return rate are much more reasonable when there is an insufficient amount of data and it does not noticeably affect the results for later periods. Unfortunately the estimations for the other parameters q and r are still unreasonable for the first few periods. But since we are only interested in accurately predicting the return rate this is not a major concern.

4.5 Negative binomial distribution with non-constant r-parameter

Using the ECME algorithm, as described in Section 3.9.6, allowed us to let the r-parameter vary between periods. The results obtained from this method when tested on product A are depicted in figure 4.11. Confidence intervals were not calculated when using this method.

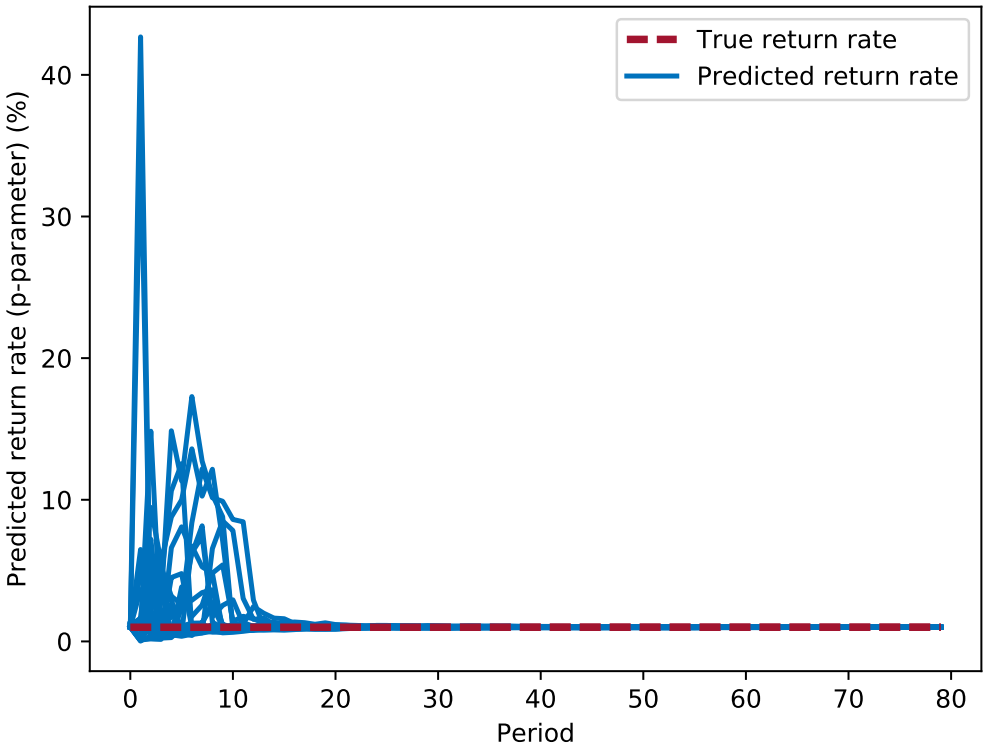

FIGURE 4.5: Predicted return rate using ECME with a negative binomial distribution with non-constant r on generated data with a true return rate of 1%.

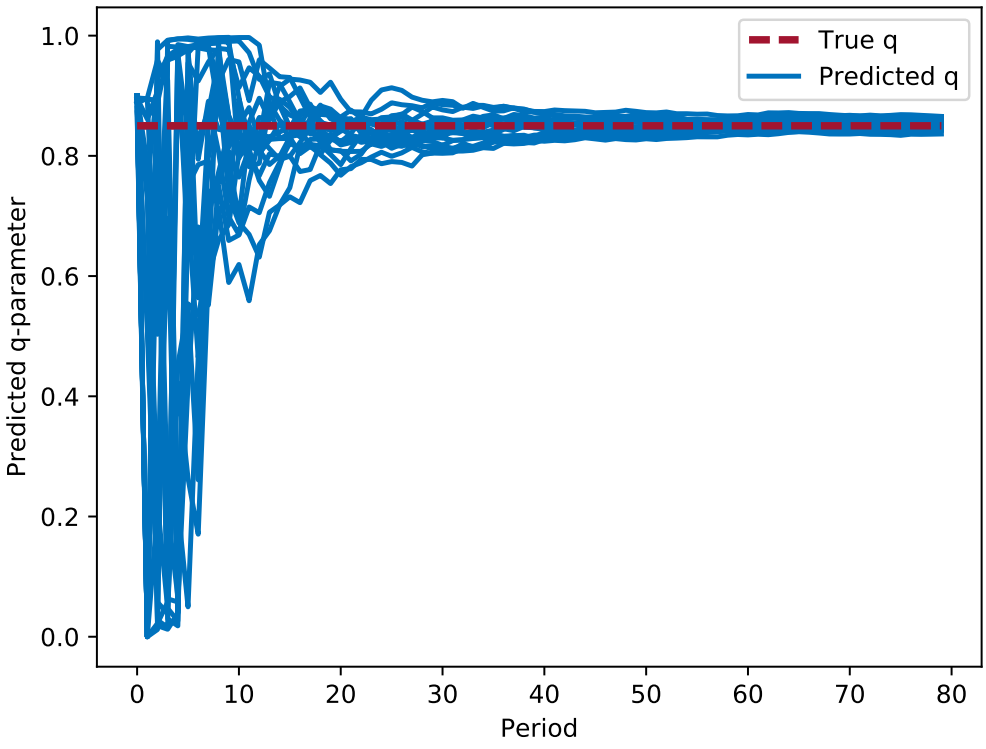

FIGURE 4.6: Predicted q-parameter using ECME with a negative binomial distribution with non-constant r on generated data with a true q-parameter value of 0.85.

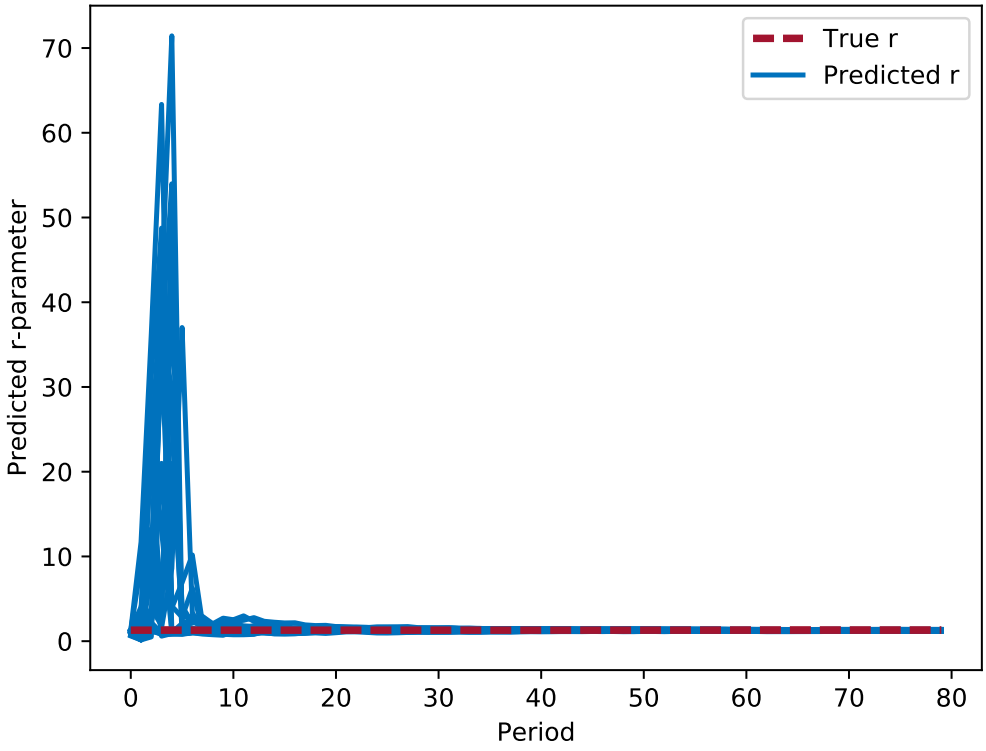

FIGURE 4.7: Predicted r using ECME with a negative binomial distribution with non-constant r on generated data with a true r-parameter value of 1.3.

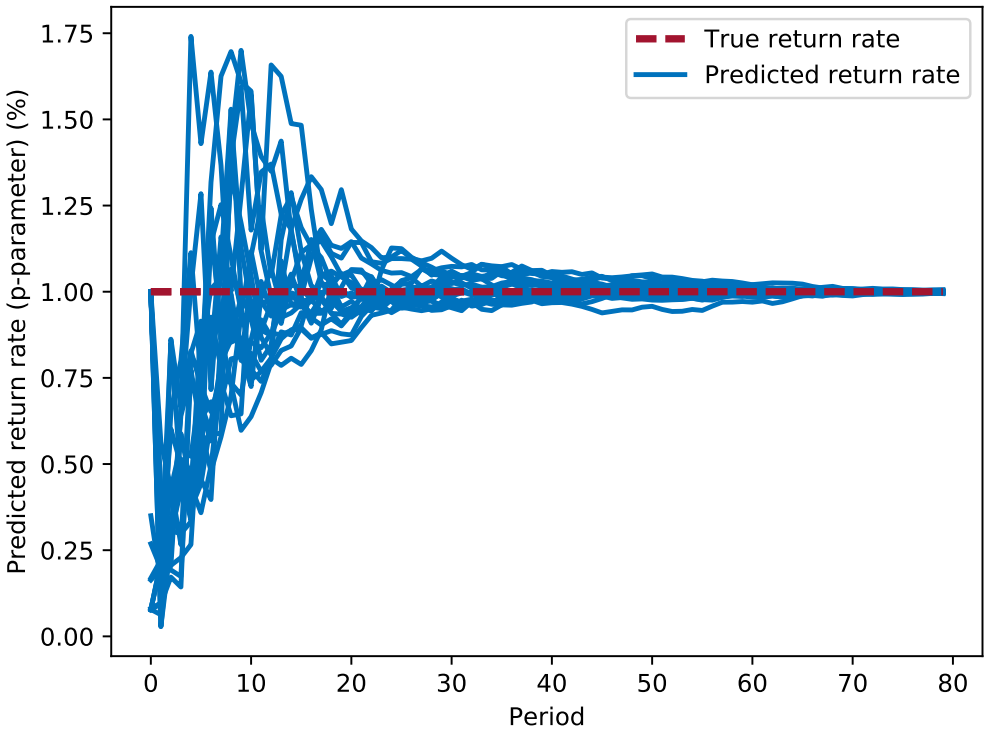

FIGURE 4.8: Predicted return rate using ECME with a negative binomial distribution with non-constant r with prior on p on generated data with a true return rate of 1%.

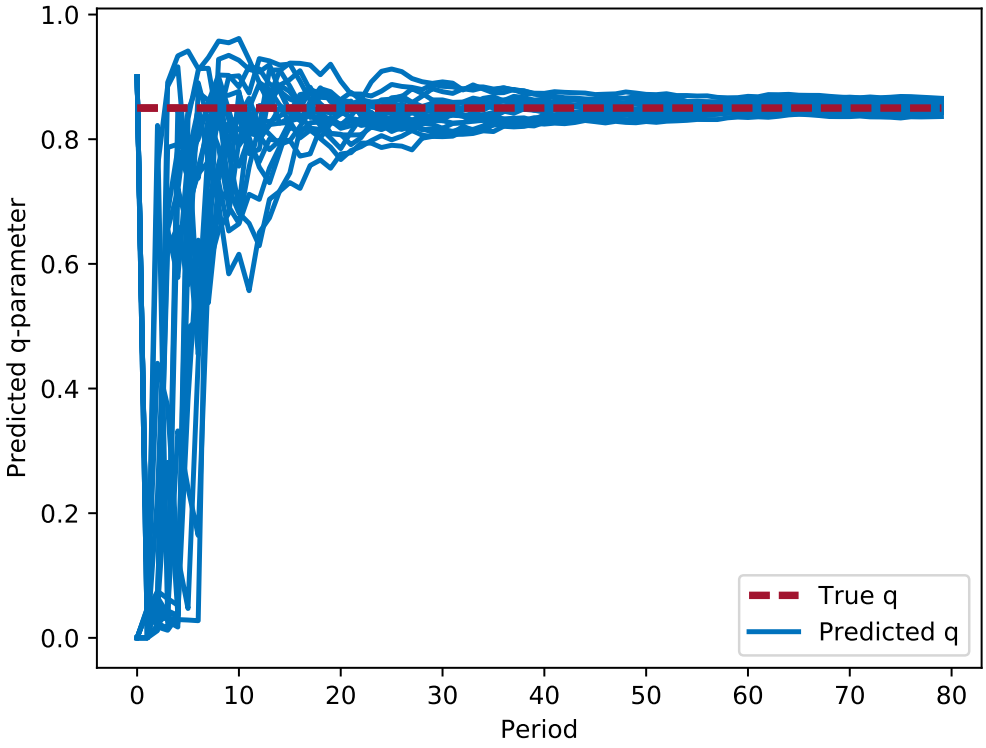

FIGURE 4.9: Predicted q-parameter using ECME with a negative binomial distribution with non-constant r with prior on p on generated data with a true q-parameter value of 0.85.

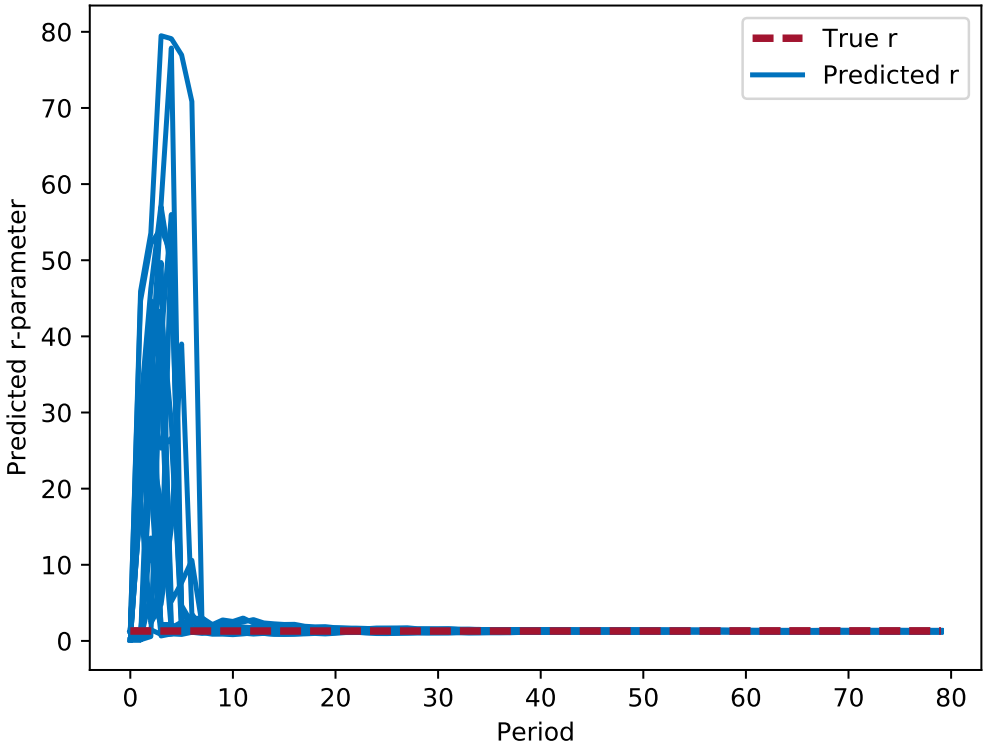

FIGURE 4.10: Predicted r using ECME with a negative binomial distribution with non-constant r with prior on p on generated data with a true r-parameter value of 1.3.

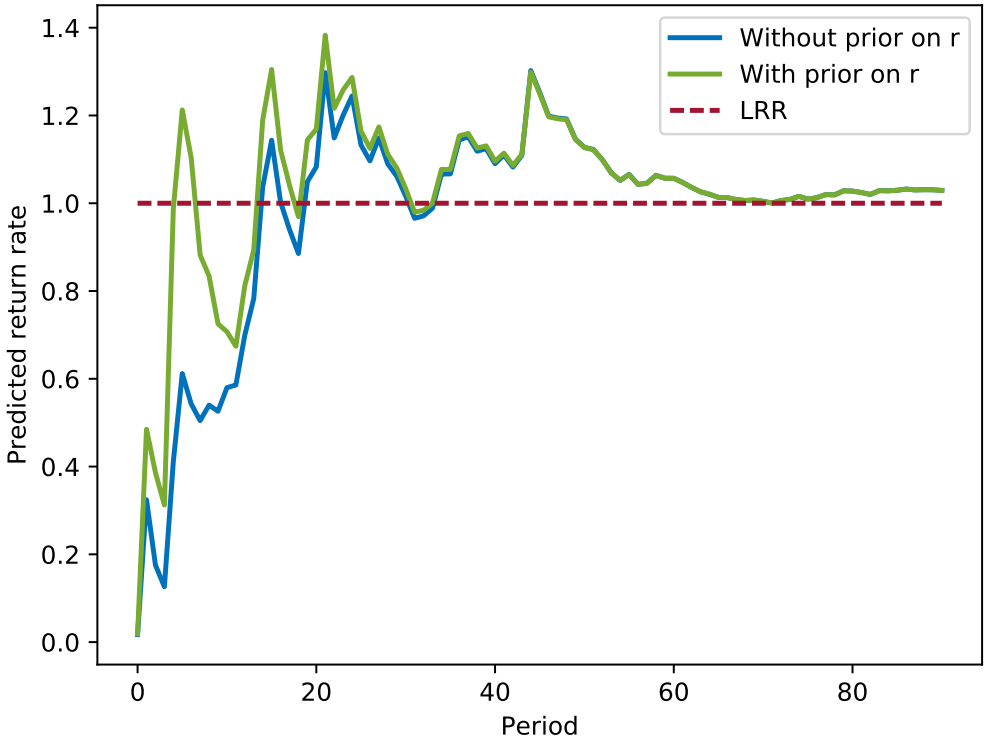

FIGURE 4.11: Predicted return rate for product A using the ECME-algorithm with negative binomial distribution with non-constant r with and without prior on r.

The r-parameter is highly overestimated when there is a small amount of data. This leads to an underestimation of the return rate in the early periods. A possible fix to this problem is to use a prior distribution for the r-parameter, see section 3.11. Axis' end of life products were used to select parameter values for the prior distribution. The resulting predicted return rates are also shown in figure 4.11. We see that with a prior on r the predicted return rate in the first few periods are higher than without a prior and, in addition, they are closer to the LRR.

4.6 Weibull distribution with constant k-parameter

As for the case with the negative binomial distribution, when using the normal EM-algorithm with a Weibull distribution we had to let the k-parameter be constant. First, the optimal value of the k-parameter was found and then the result when using it on product A is shown.

Using the same approach as for r with a negative binomial distribution, see section 4.3.1, the optimal value was set to k=1.35. This gave the results shown in figure 4.12.

4.7 Weibull distribution with non-constant k-parameter

To let the k-parameter be non-constant and be able to vary between periods we used the ECME algorithm, see Section 3.9.6. The results from using this algorithm with a

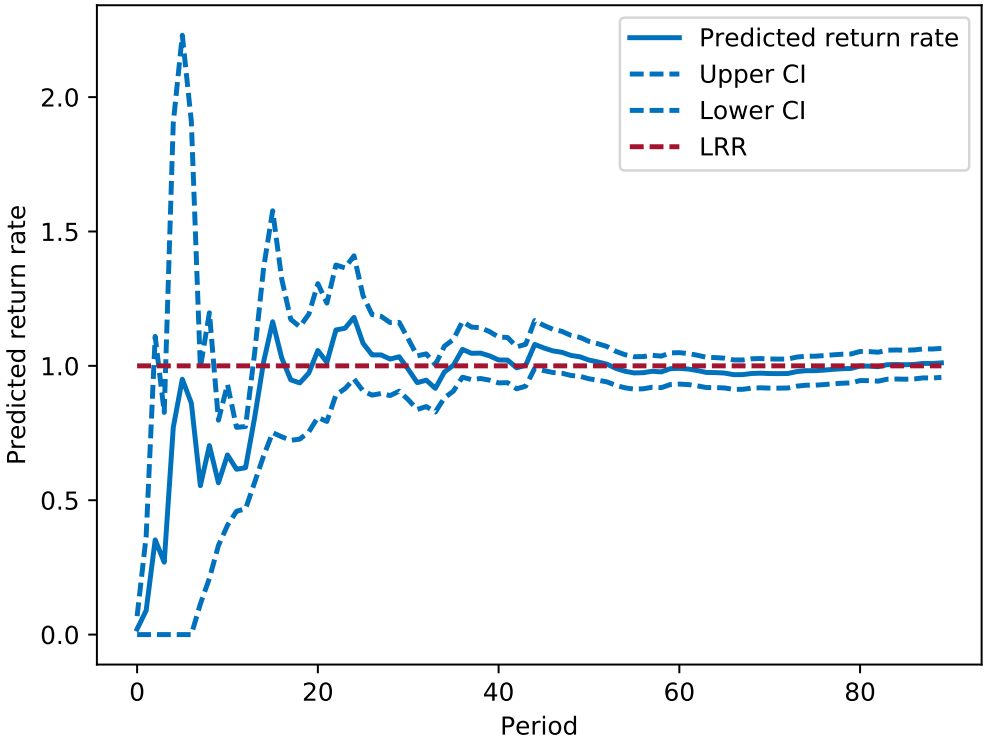

FIGURE 4.12: Predicted return rate for product A using the EM-algorithm with Weibull distribution with constant k together with approximate 95% confidence intervals.

Weibull distribution on product A is shown in figure 4.13.

For the first few periods the algorithm very heavily overestimates the value of the k-parameter. To avoid this we use a prior on k with parameters based on the Axis' products which have reached end of life. The results obtained with a prior on k are shown in the same figure. This keeps the value of the k-parameter closer to what we may expect which makes the predicted return rates in the beginning of the product's life span a bit higher and for many early periods it is closer to the LRR.

4.8 Method comparison

This section contains plots with comparisons between the different methods used. Figure 4.14 show the results yielded for product A when using the EM-algorithm with a negative binomial distribution with constant r-parameter compared to when using the same algorithm with a Weibull distribution with constant k-parameter. The methods give similar results which indicates that the two methods give similar approximations for the time to return distribution. Hence the difference between using a negative binomial distribution and using a Weibull distribution is small.

In figure 4.15 a comparison between the results obtained for product A when using the ECME algorithm for the distributions is shown. One is with a negative binomial distribution with a non-constant r-parameter with a prior on r and the other is with a Weibull distribution with a non-constant k-parameter with a prior on k. The two

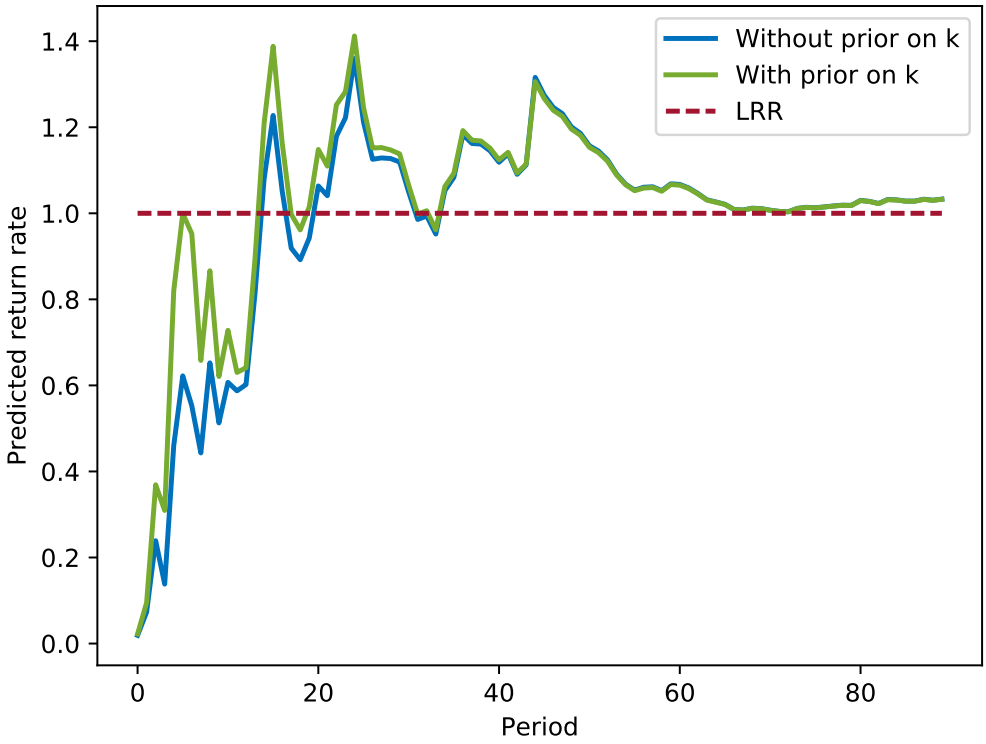

FIGURE 4.13: Predicted return rate for product A using the ECME-algorithm with Weibull distribution with non-constant k with and without prior on k.

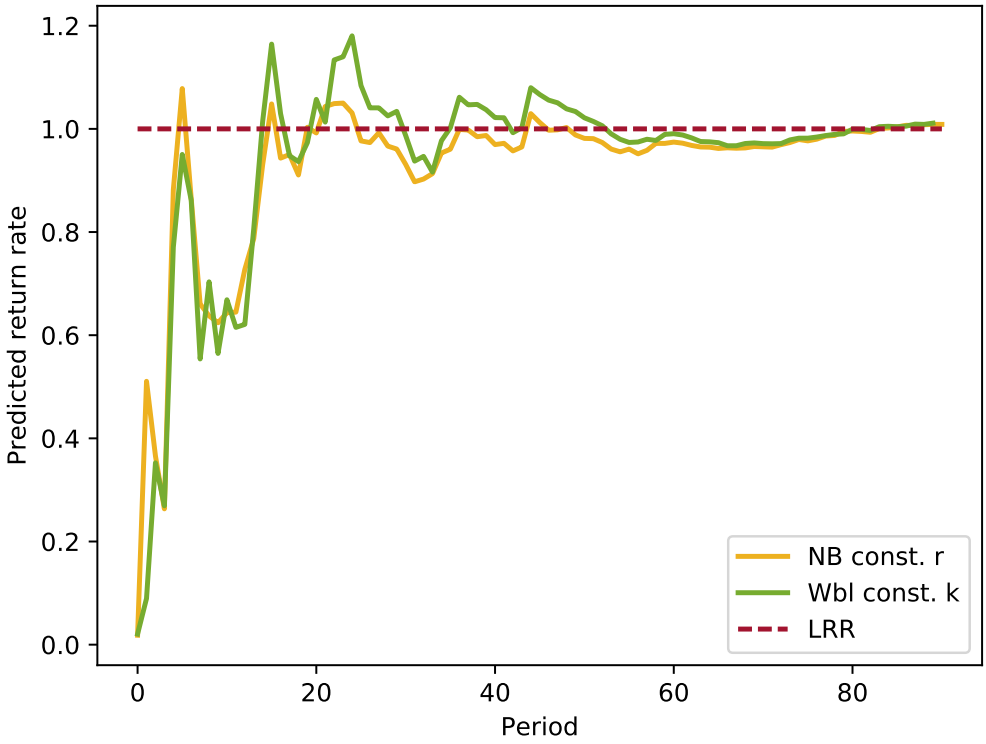

FIGURE 4.14: Comparing the predicted return rate for product A using the negative binomial distribution with constant r (NB const. r) and the Weibull distribution with constant k (Wbl const. k).

plots obtained are similar again indicating that there is no major difference between using the negative binomial distribution and the Weibull distribution.

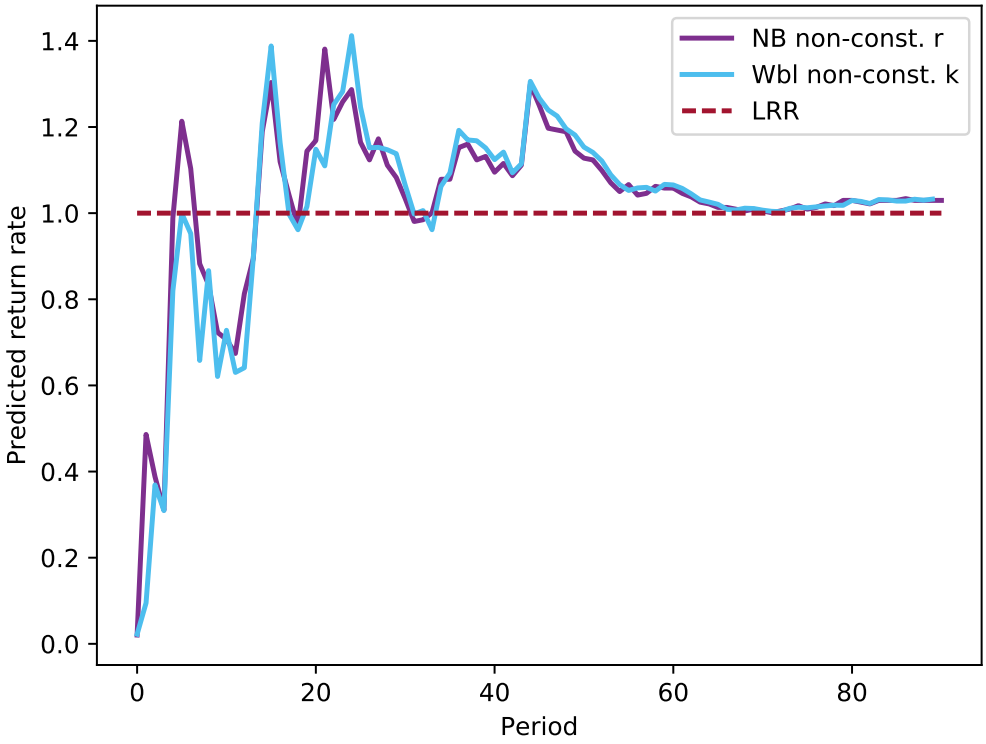

FIGURE 4.15: Comparing the predicted return rate for product A using the negative binomial distribution with non-constant r with prior on r (NB non-const. r) and the Weibull distribution with non-constant k with prior (Wbl non-const. k).

Finally, we show in figure 4.16 the difference between using the ARR method, the Kaplan-Meier method, the negative binomial distribution with constant r (using the EM algorithm) and negative binomial distribution with non-constant r (using the ECME algorithm). This figure shows that the slowest to reach the LRR is the ARR method, the second slowest is the Kaplan-Meier method and the methods with constant and non-constant r are approximately equally fast.

4.9 Method comparisons using error values

Error values that were computed as described in Section 3.12 were used to compare the accuracy of the different methods we used to predict the return rate. The error value calculated is defined in such a way that a lower error means the method is more accurate.

In table 4.1 the error values for products A-H are shown for each of the different methods. Below is a list with what method each number in the table represent. In the table we see that the method with the highest error value on average is the ARR method and the one with the lowest error value on average is the EM-algorithm with a negative binomial distribution with constant r-parameter. Methods with a

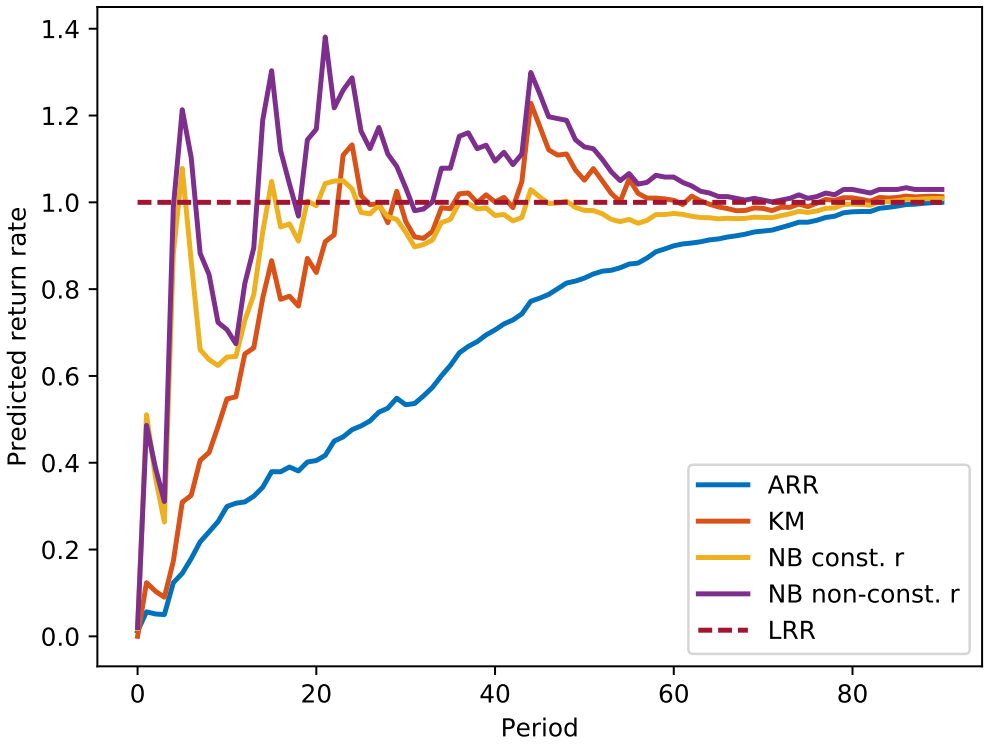

FIGURE 4.16: Comparing the predicted return rate for product A using the aggregated return rate (ARR), the Kaplan-Meier method (KM) and the negative binomial distribution with constant r (NB const. r) and non-constant r with prior (NB non-const. r).

negative binomial distribution for the time to return generally performed better than methods which used a Weibull distribution.

I : ARR

II : KM

III : Negative binonimal with constant r

IV : Negative binomial with non-constant r

V : Negative binomial with non-constant r with prior on r

VI : Weibull with constant k

VII : Weibull with non-constant k

VIII : Weibull with non-constant k with prior on k

TABLE 4.1: The table shows the measured errors for every products using different methods. The smallest error for each product is written in bold style.

Method\Product	A	B	C	D	E	F	G	H	Average
I	0.324	0.368	0.358	0.202	0.361	0.341	0.370	0.326	0.331
II	0.141	0.158	0.219	0.102	0.193	0.190	0.182	0.158	0.168
III	**0.085**	**0.083**	0.126	0.087	0.143	0.147	0.157	0.107	**0.117**
IV	0.147	0.163	0.163	**0.084**	0.152	0.135	**0.135**	0.120	0.137
V	0.124	0.144	**0.098**	0.092	**0.122**	0.155	0.206	0.132	0.134
VI	0.096	0.120	0.124	0.085	0.154	**0.121**	0.169	**0.097**	0.121
VII	0.157	0.181	0.167	0.091	0.154	0.140	0.156	0.130	0.147
VIII	0.139	0.171	0.116	0.093	0.138	0.128	0.199	0.124	0.138

4.10 Results for other products

Figures 4.17 - 4.23 show the results of using four different methods to predict the return rate for products B-H. The methods are the aggregated return rate, the Kaplan-Meier method, the EM-algorithm with negative binomial distribution with constant r and the ECME-algorithm with negative binomial distribution with non-constant r. The results for the on average most successful method, see table 4.1, when applied to products B-H together with 95% confidence intervals is shown in appendix A.8.

4.11 Confidence intervals accuracy

Since the confidence intervals are only approximately 95% and they are valid only for the data we have seen up to the current period, it is interesting to see how often the intervals cover the actual lifetime return rate. We did this test for the optimal method found, i.e. the EM-algorithm with a negative binomial distribution with constant $r = 2.05$. The results are shown in table 4.2. As can be seen the confidence interval only covers the lifetime return rate in $\approx 78.8\%$ of all periods for all products which is significantly lower than 95%.

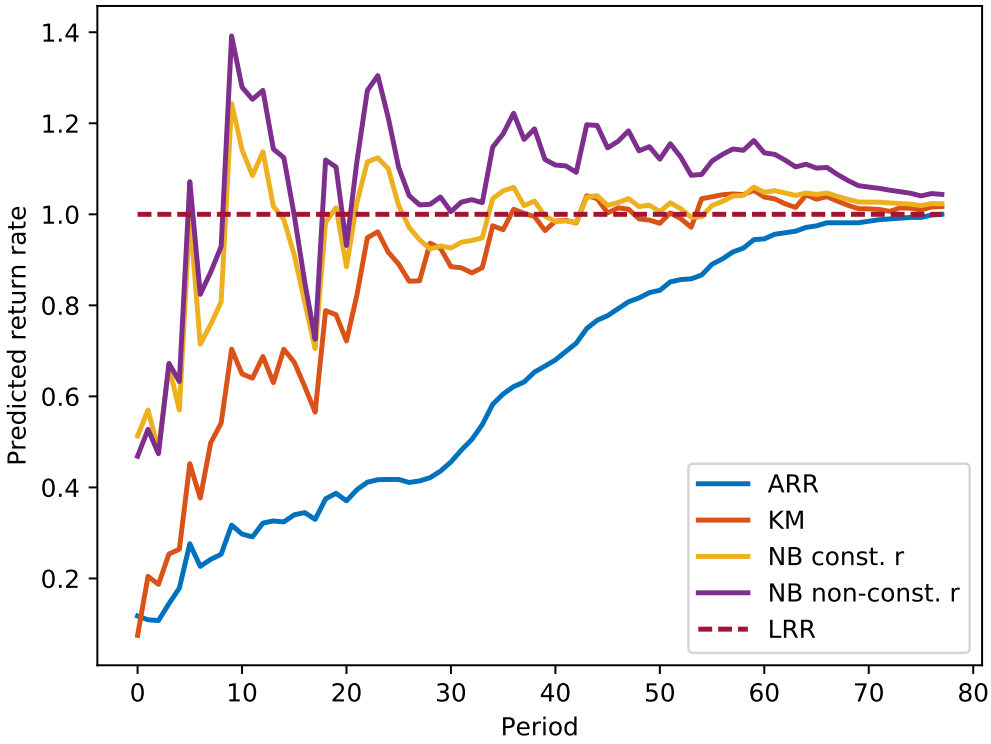

FIGURE 4.17: Predicted return rates for product B using different methods.

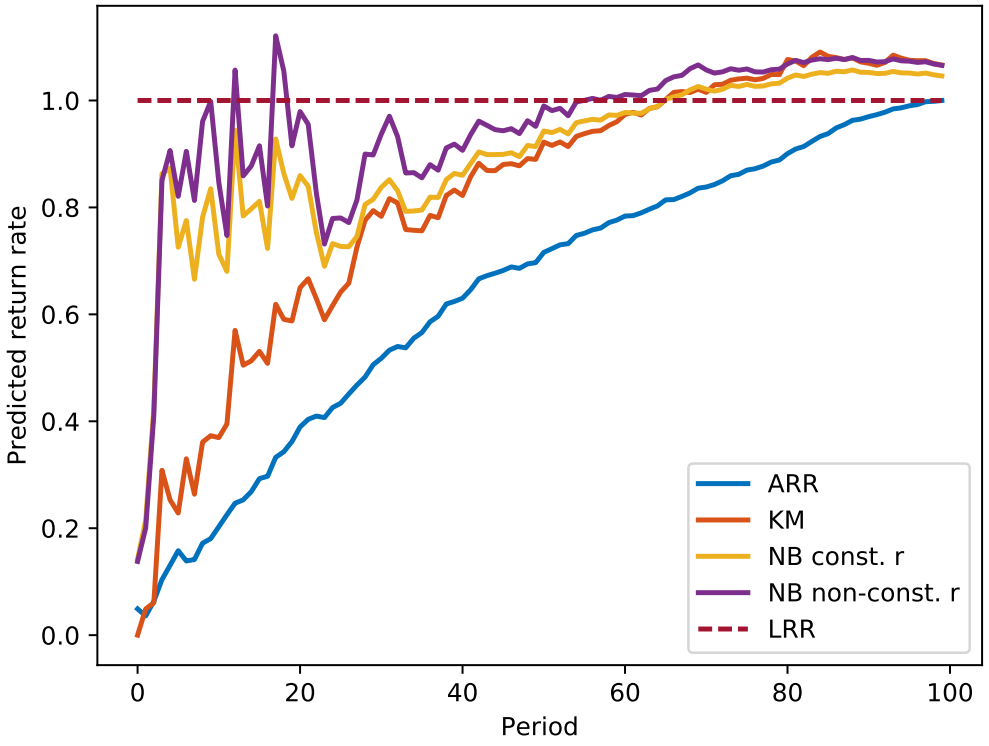

FIGURE 4.18: Predicted return rates for product C using different methods.

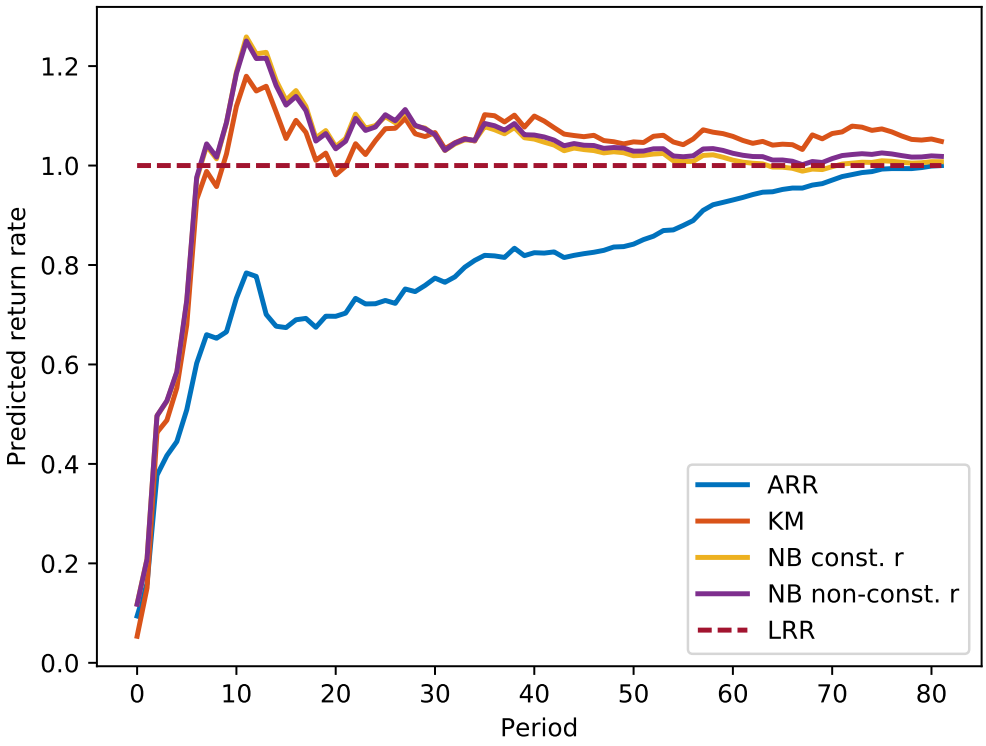

FIGURE 4.19: Predicted return rates for product D using different methods.

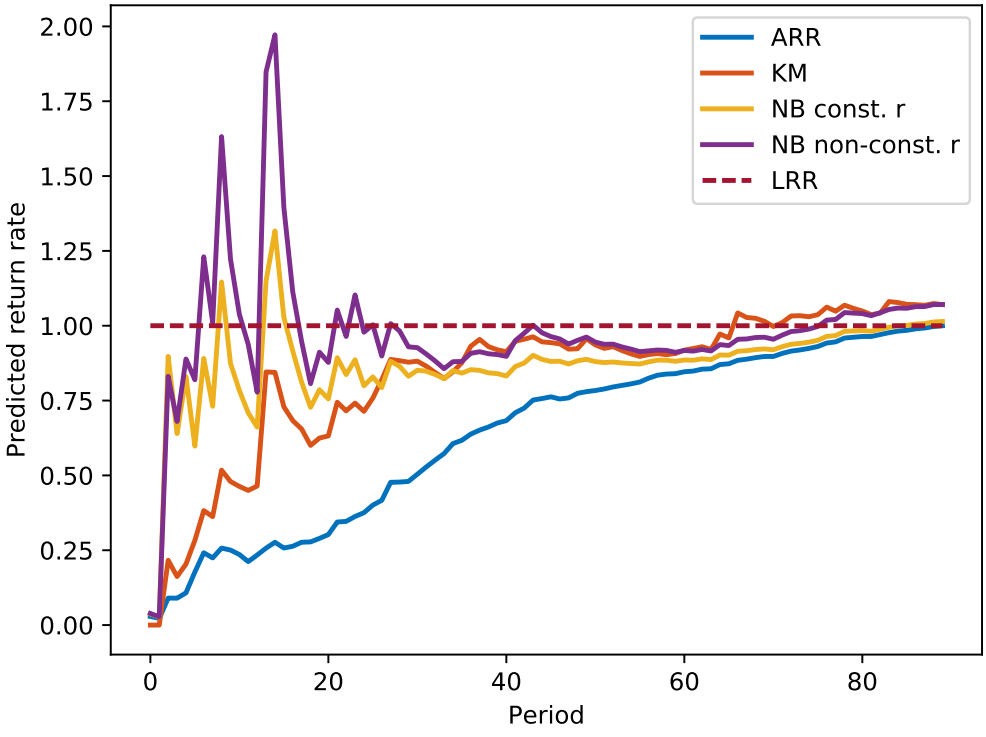

FIGURE 4.20: Predicted return rates for product E using different methods.

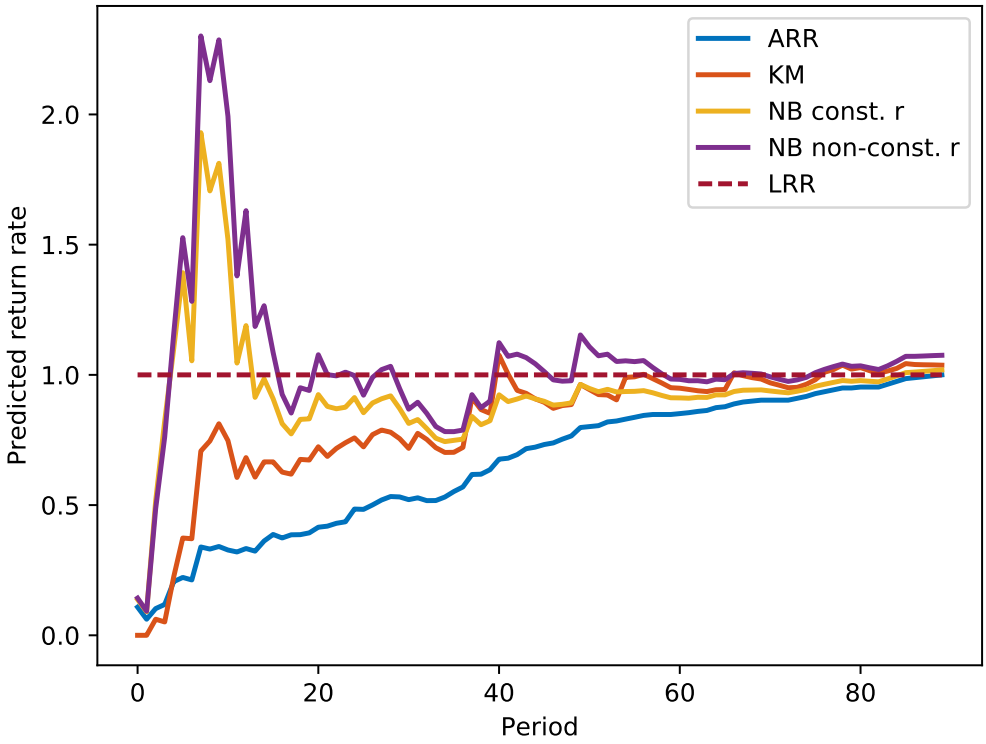

FIGURE 4.21: Predicted return rates for product F using different methods.

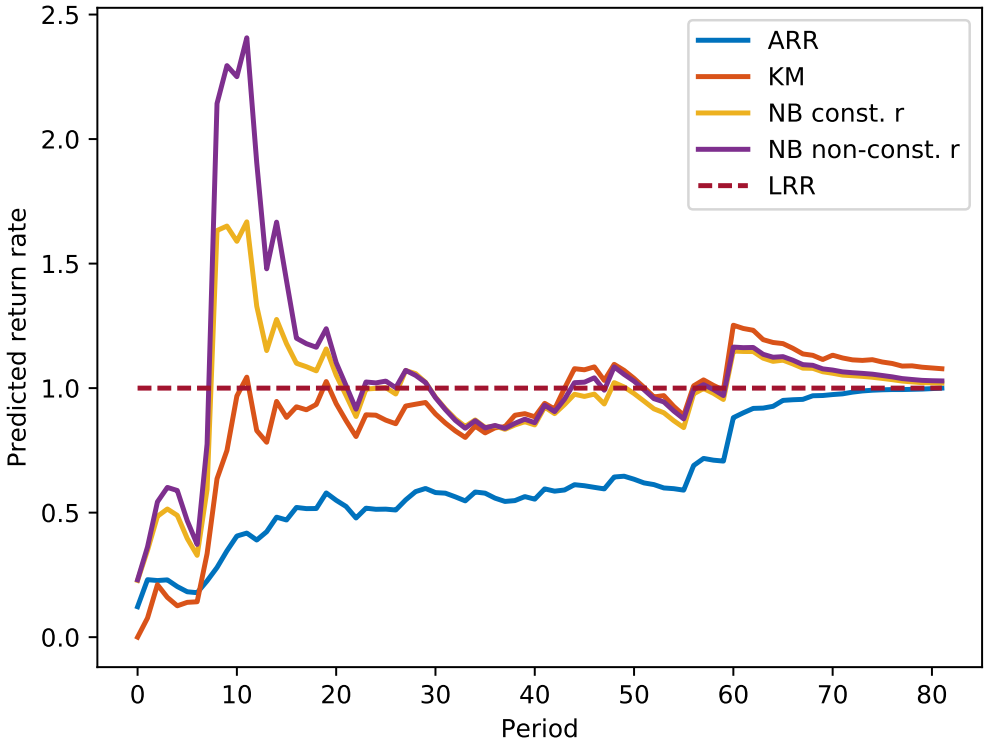

FIGURE 4.22: Predicted return rates for product G using different methods.

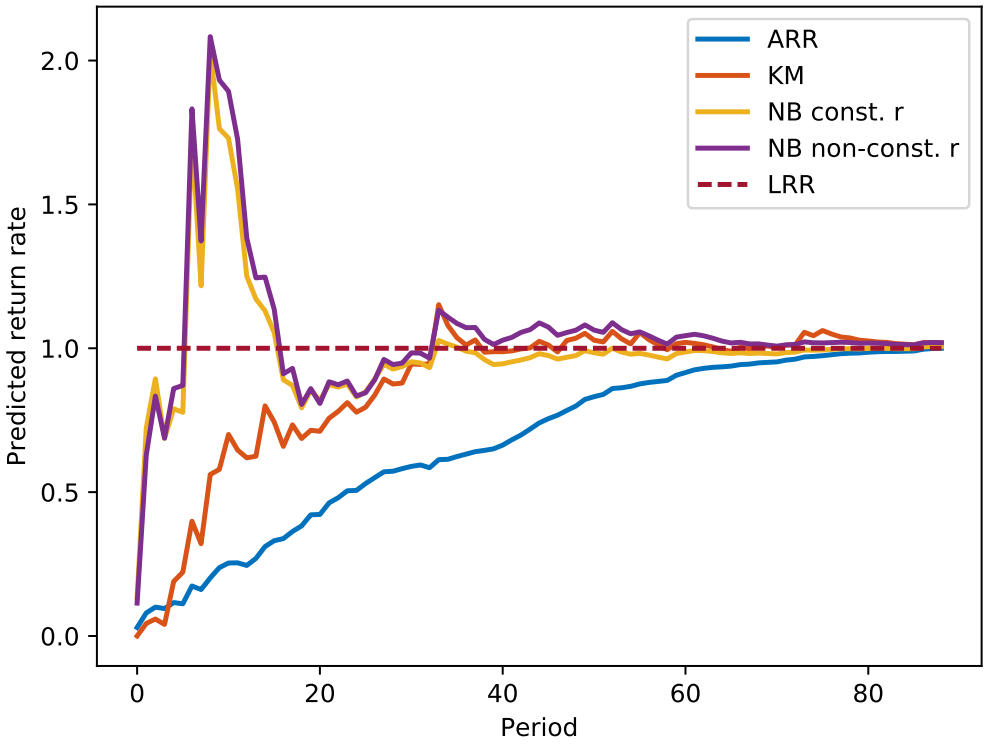

FIGURE 4.23: Predicted return rates for product H using different methods.

TABLE 4.2: The table shows the observed accuracy of the confidence intervals obtained when using the EM-algorithm with a negative binomial distribution with constant r for each product and for all products combined.

Product	CI Accuracy (%)
A	87.9
B	98.7
C	68.0
D	90.2
E	42.2
F	87.8
G	68.3
H	91.0
All	**78.8**

4.12 Performance analysis

In order to compare the performance of the different methods we measured the time and the number of iterations it took to predict the return rate for products A-H in every period. The execution times include the processing time of the data relevant to each method. The results of the performance analysis are shown in table 4.3 and as can be seen the fastest method was ARR. We also see that the Kaplan-Meier method was faster than the EM-algorithm without any acceleration, but slightly slower than the accelerated version. By far the slowest method was ECME which took 17 times longer than the normal EM-algorithm.

TABLE 4.3: Table showing the execution times and the number of iterations required for different methods.

Method	Execution time (s)	Number of EM-iterations
ARR	7.60	-
KM	12.86	-
EM	18.22	48572
EM with acceleration	11.78	5905
ECME	312.75	164959

5 Discussion

The purpose of this thesis was to test different methods for predicting the return rate of a product. As can be seen in the results the methods we tested all outperformed the aggregated return rate. The EM-algorithm with negative binomial distribution with a constant r-parameter proved to be the most successful method. Surprisingly it proved to be better with a constant r-parameter rather than estimating it. This is most likely because the algorithm can not accurately estimate all parameters until there is a sufficient amount of data. Hence, by forcing the r-parameter to be a reasonable value constantly, the estimated return rates are accurate even for small amounts of data.

Another advantage with using a constant r-parameter is that execution time is significantly faster compared to using a non-constant r-parameter, as seen in table 4.3. This is due to both the difficulty in maximizing the likelihood with non-constant r, since this introduces an extra dimension and hence requires more EM-iterations to converge, and the fact that each EM-iteration also consists of a one-dimensional line-search.

A disadvantage with using a constant r is that the model is slightly less general i.e. makes more assumptions about the data. We conducted some sensitivity tests using generated data with a true r-parameter that differed to the constant r-parameter assumed by the model. These tests showed that the model is robust to differences in the r-parameter and the model was still able to make good return rate predictions.

With a negative binomial distribution the optimal constant value of the r-parameter was found to be $r = 2.05$ which is close to 2. A negative binomial distributed stochastic variable with parameters r and q is a sum of r geometrically distributed stochastic variables with parameter q [9]. This means that a possible explanation for the optimal value of r being 2 is that the unit spends an approximately geometrically distributed amount of time with the distributor and then another approximately geometrically distributed amount of time with the customer.

In order to predict the return rate for all of Axis' products periodically each month we also developed a Python script which automatically goes through Axis databases and calculates predictions for all products. These predictions are stored in a text file and can be imported to a data visualization program to get an overview of which of Axis' products are currently performing well and which should be investigated further.

A problem with the methods tested here is the assumption of a constant p. In general this is a simplification since products may be updated or unexpected events happen that increase or decrease the underlying return rate. As can be seen in table 4.2 the accuracy of the confidence intervals for product E is significantly lower than 95%. Looking at figures A.4 and 4.20 it can be seen that the predictions are fairly constant up until period ≈ 60 and then increases, indicating that something happened which altered the underlying return rate. Similar reasons may explain why the overall

accuracy of the confidence intervals for all products is significantly lower than 95%. Allowing for a time-dependent return rate p could possibly improve the accuracy of the confidence intervals. Another reason this could be beneficial is that it could allow the effects of updating or changing a product to be seen more quickly.

A simple way of taking the time-dependency of p into account is by using a window function. With this the predictions will only be based on recently sold units and corresponding returns. E.g. using a window length of one year the predictions are based on units sold at most one year ago and their corresponding returns. This, however, decreases the amount of data available for the predictions, thus making the predictions more noisy.

5.1 Covariates

It may be possible to improve the predictions by including additional information about the units such as: where the unit was sold, where it was produced, price and other unit specifications. One possible way of doing this would be to include the covariates by modeling p using logistic regression as done by Wu et al. [26] and Hess and Mayhew [11].

If the column-vector $\boldsymbol{x}_i = [1\ x_1^{(i)}\ \dots\ x_c^{(i)}]^{\mathrm{T}}$ contains the covariates for unit i and $\boldsymbol{\beta} = [\beta_0\ \beta_1\ \dots \beta_c]^{\mathrm{T}}$ is a column-vector containing the corresponding coefficients. Then the return rate p for unit i is modeled as

$$p(\boldsymbol{x}_i, \boldsymbol{\beta}) = \frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{x}_i}}. \tag{5.1}$$

The resulting complete likelihood and a formula for the return rate of a group of units is shown in appendix A.6.

It is also possible to include covariates in the modeling of the time to return distribution. This could be done by using a Cox-proportional model which models the influence of the covariates on the hazard function. The hazard function corresponding to the distribution ρ is

$$h(t) = \frac{\rho(t)}{1 - R(t)} = \frac{\rho(t)}{S(t)} \tag{5.2}$$

The Cox-proportional model uses an arbitrary baseline hazard h_0 and is [13]

$$h(t|\boldsymbol{\alpha}, \boldsymbol{z}_i) = h_o(t)e^{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{z}_i} \tag{5.3}$$

where $\boldsymbol{z}_i = [z_1^{(i)}\ z_2^{(i)}\ \dots\ z_d^{(i)}]^{\mathrm{T}}$ is a vector containing the covariates for unit i and $\boldsymbol{\alpha} = [\alpha_1\, \alpha_2\, \dots\, \alpha_d]^{\mathrm{T}}$ is a vector containing the corresponding coefficients. To calculate the distribution ρ corresponding to the hazard function given by Cox-proportional model the survival function is needed. It can be calculated using

$$S(t) = \exp\left(-\int_0^t h(\tau)d\tau\right) \tag{5.4}$$

and ρ can then be calculated using (5.2). The distribution acquired includes the covariates and can be used in the complete likelihood which can then be solved using the EM-algorithm to estimate $\boldsymbol{\alpha}$.

5.2 Truncated distribution

In some cases it is known that the time to return cannot exceed a certain value, denoted T_r. This is the case when e.g. there is a limited warranty period and when this has been exceeded the unit can no longer be returned. In these cases it might be appropriate to use a right truncated distribution such that the probability of return after T_r is zero. If the non-truncated time to return distribution is denoted $\rho(t|\boldsymbol{\theta})$ with CDF $R(t|\boldsymbol{\theta})$ then the right truncated distribution with truncation time T_r becomes

$$\begin{aligned} P(T=t) &= \begin{cases} \dfrac{\rho(t|\boldsymbol{\theta})}{R(T_r|\boldsymbol{\theta})} & \text{for } 0 \leq t \leq T_r \\ 0 & \text{otherwise} \end{cases} \\ P(T \leq t) &= \begin{cases} \dfrac{R(t|\boldsymbol{\theta})}{R(T_r|\boldsymbol{\theta})} & \text{for } 0 \leq t \leq T_r \\ 1 & \text{for } t > T_r \end{cases} \end{aligned} \tag{5.5}$$

where we divide by $R(T_r|\boldsymbol{\theta})$ to normalize the distribution. It might seem that this simple adjustment would not cause any problems when estimating the parameters $\boldsymbol{\theta}$ and p, this is however not the case. The complete likelihood used in the EM-algorithm becomes

$$\begin{aligned} \mathcal{L}(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{t}^{(u)}, \mathbf{z}) &= \prod_{i=1}^{n} p^{z_i} \cdot \left(\frac{\rho(t_i|\boldsymbol{\theta})}{R(T_r|\boldsymbol{\theta})} \right)^{z_i} (1-p)^{(1-z_i)} = \\ &= p^m \prod_{i=1}^{m} \frac{\rho(t_i^{(o)}|\boldsymbol{\theta})}{R(T_r|\boldsymbol{\theta})} \prod_{i=m+1}^{n} p^{z_i} \cdot \left(\frac{\rho(t_i|\boldsymbol{\theta})}{R(T_r|\boldsymbol{\theta})} \right)^{z_i} (1-p)^{(1-z_i)}. \end{aligned} \tag{5.6}$$

for a general truncated time to return distribution, compare with (3.18).

This can in general no longer be solved for all parameter $\boldsymbol{\theta}$ since the normalizing constant $R(T_r|\boldsymbol{\theta})$ is a function of $\boldsymbol{\theta}$. The normalizing constant would e.g. be $1 - I_q(T_r + 1, r)$ in the case of a negative binomial distribution. Hence the truncation would make it intractable to find an analytic updating formula for q.

5.3 Handling missing data

The entries in the RMA-database for some units lack a sales date or the sales date is past the return date giving a negative time to return, as described in section 2.1.2. We chose not to include these units since they were rare. If a significant number of units have a missing sales date it may be appropriate to handle these units. In this section we show how to include such units in our model.

The units with missing sales dates can be seen as left or interval-censored since we know that the true time to return is less than or equal to r_i and larger or equal to 0,

where r_i is the return period of unit i. Hence the units with missing data contribute to the observed likelihood in the form of the factor

$$\prod_{i=o+1}^{m} pR(r_i|\boldsymbol{\theta}) \tag{5.7}$$

if we index the units such that units $i = 1, \ldots, m$ have been returned but units $i = o+1, \ldots, m$ have missing or invalid sales date. The complete likelihood becomes

$$\begin{aligned} \mathcal{L}(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{t}^{(u)}, \mathbf{z}) &= \prod_{i=1}^{n} p^{z_i} \cdot \rho(t_i|\boldsymbol{\theta})^{z_i}(1-p)^{(1-z_i)} = \\ &= p^m \prod_{i=1}^{o} \rho(t_i^{(o)}|\boldsymbol{\theta}) \prod_{i=o+1}^{m} \rho(t_i^{(o)}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} p^{z_i} \cdot \rho(t_i^{(u)}|\boldsymbol{\theta})^{z_i}(1-p)^{(1-z_i)}. \end{aligned} \tag{5.8}$$

It can easily be seen that when including these units in the EM-algorithm the updating formula for p is unchanged since we still have m observed returns. However, $t_i^{(o)}$ is unknown for units $i = o+1, \ldots, m$, and must be estimated.

Appendix A.7 shows how units with missing or invalid sales date can be incorporated in the model when using the negative binomial or Weibull distribution.

Bibliography

[1] O. O. Aalen, Ø. Borgan, and H. K. Gjessing. *Survival and Event History Analysis: A Process Point of View*. Springer Science+Business Media, LLC, 2008, pp. 1–2, 5, 90–94.

[2] N. Balakrishnan and S. Pal. "EM Algorithm-Based Likelihood Estimation for Some Cure Rate Models." In: *Journal of Statistical Theory and Practice* 6.4 (2012), p. 698.

[3] B. H. Baltagi. *Econometrics*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011, pp. 131–.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, 2006, pp. 440–441, 537–542.

[5] T. Clottey, W. C. Benton, and R. Srivastava. "Forecasting Product Returns for Remanufacturing Operations". In: *Decision Sciences* 43.4 (Aug. 2012), pp. 589–614.

[6] Morris H. DeGroot and Mark J. Schervish. *Probability and statistics.* 3rd ed. Boston, Mass. ; London : Addison-Wesley, cop, 2002, pp. 172, 435–444.

[7] A. Dempster, N. Laird, and D. Rubin. "Maximum Likelihood from Incomplete Data via the EM Algorithm". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 39.1 (Jan. 1977), pp. 1–38.

[8] B. Efron and D. V. Hinkley. "Assessing the Accuracy of the Maximum Likelihood Estimator: Observed Versus Expected Fisher Information." In: *Biometrika* 3 (1978), pp. 457–482.

[9] C. Forbes and M. Evans. *Statistical Distributions*. 4th ed. Oxford : Wiley-Blackwell, 2010, pp. 55–56, 139.

[10] G. H. Givens and J. A. Hoeting. *Computational Statistics*. 2nd ed. John Wiley & Sons, Inc., 2013, pp. 97–98, 201–202.

[11] James D. Hess and Glenn E. Mayhew. "Modeling Merchandise Returns in Direct Marketing." In: *Journal of Direct Marketing* 11.2 (1997), pp. 20 –35.

[12] E. L. Kaplan and P. Meier. "Nonparametric estimation from incomplete observations". In: *Journal Of The American Statistical Association* 53.282 (June 1958), pp. 457–481.

[13] J. P. Klein and M. L. Moeschberger. *Survival Analysis: Techniques for Censored and Truncated Data*. 2nd ed. Springer-Verlag New York, Inc., 2003, pp. 63–70.

[14] J. P. Klein et al. *Handbook of Survival Analysis*. Boca Raton : CRC Pres, 2014, pp. 113–114.

[15] M. Krapp, J. Nebel, and R. Srivastava. "Forecasting product returns in closed-loop supply chains". In: *International Journal Of Physical Distribution & Logistics Management* 43.8 (2013), pp. 614–637.

[16] C. Lai. *Generalized Weibull Distributions*. 1st ed. Springer Berlin Heidelberg, 2014, pp. 1–5.

[17] K. Lange. "A Quasi-Newton acceleration of the EM-algorithm". In: *Statistica Sinica* 5.1 (Jan. 1995), pp. 1–18.

[18] C Liu and D. B. Rubin. "The ECME Algorithm: A Simple Extension of EM and ECM with Faster Monotone Convergence." In: *Biometrika* 81.4 (1994), pp. 633–648.

[19] T. A. Louis. "Finding the Observed Information Matrix when Using the EM Algorithm". In: *Royal Statistical Society* 44.2 (Jan. 1982), pp. 226–233.

[20] R. A Maller and S. Zhou. "Estimating the Proportion of Immunes in a Censored Sample". In: *Biometrika* 79.4 (Dec. 1992), pp. 731–739.

[21] G. J. McLachlan and T. Krishan. *The EM Algorithm and Extensions*. 2nd ed. John Wiley & Sons, Inc., 2008, pp. 1–2, 5–6, 77–78, 137–138, 159–176.

[22] X.-L. Meng and D. B. Rubin. "Maximum Likelihood Estimation via the ECM Algorithm: A General Framework." In: *Biometrika* 80.2 (1993), pp. 267–278.

[23] P. R. Rider. "The Negative Binomial Distribution and the Incomplete Beta Function". In: *The American Mathematical Monthly* 69.4 (Apr. 1962), pp. 302–304.

[24] G. R. Terrell. *Mathematical Statistics: A Unified Introduction*. Springer-Verlag New York, Inc, 1999, pp. 245–248.

[25] B. L. Toktay, L. M. Wein, and S. A. Zenios. "Inventory Management of Remanufacturable Products". In: *Management Science* 46.11 (Nov. 2000), pp. 1412–1426.

[26] Yu Wu et al. "Extension of a Cox proportional hazards cure model when cure information is partially known." In: *Biostatistics* 15.3 (2014), pp. 540 –554.

A Appendix

A.1 Finding an expression for the CDF of the negative binomial distribution

Following [23] we make the variable change $u = q/(1-q)$. This gives the negative binomial pmf

$$P(T = k) = \rho(k, u, r) = \frac{\Gamma(k+r)}{k!\,\Gamma(r)} \frac{u^k}{(1+u)^{r+k}} \tag{A.1}$$

If we denote the cumulative distribution function of ρ with R we get

$$R(k, u, r) = P(T \leq k) = \sum_{i=0}^{k} \rho(i, u, r) \quad \text{and} \quad 1 - R(k, u, r) = \sum_{i=k+1}^{\infty} \rho(i, u, r) \tag{A.2}$$

Differentiating the second expression with respect to u gives

$$-\frac{\partial R(k, u, r)}{\partial u} = \sum_{i=k+1}^{\infty} \frac{\partial \rho(i, u, r)}{\partial u} \tag{A.3}$$

$$\frac{\partial \rho(i, u, r)}{\partial u} = \frac{\Gamma(i+r)}{i!\,\Gamma(r)} \left(\frac{u^{i-1} i}{(1+u)^{r+i}} - \frac{u^i (r+i)}{(1+u)^{r+i+1}} \right) \tag{A.4}$$

The last part of the j:th term in the sum and the first part of the $j+1$:th term in the sum cancel out, as shown below

$$\begin{aligned} &\frac{\Gamma(j+1+r)}{(j+1)!\,\Gamma(r)} \frac{u^j (j+1)}{(1+u)^{r+j+1}} - \frac{\Gamma(j+r)}{j!\,\Gamma(r)} \frac{u^j (r+j)}{(1+u)^{r+j+1}} = \\ =&\frac{\Gamma(j+r)(j+r)}{j!\,\Gamma(r)} \frac{u^i}{(1+u)^{r+i+1}} - \frac{\Gamma(i+r)}{i!\,\Gamma(r)} \frac{u^i (r+i)}{(1+u)^{r+i+1}} = \\ =&\frac{\Gamma(i+r)(r+i)u^i}{i!\,\Gamma(r)(1+u)^{r+i+1}} (1-1) = 0 \end{aligned} \tag{A.5}$$

where we have used that $\Gamma(z+1) = z\Gamma(z)$.

This leaves only the first part of the first term in the sum which gives

$$\begin{aligned} -\frac{\partial R(k, u, r)}{\partial u} &= \sum_{i=k+1}^{\infty} \frac{\Gamma(i+r)}{i!\,\Gamma(r)} \left(\frac{u^{i-1} i}{(1+u)^{r+i}} - \frac{u^i (r+i)}{(1+u)^{r+i+1}} \right) = \\ &= \frac{\Gamma(k+1+r) u^k (k+1)}{(k+1)!\,\Gamma(r)(1+u)^{r+k+1}} = \frac{\Gamma(k+1+r) u^k}{k!\,\Gamma(r)(1+u)^{r+k+1}} \end{aligned} \tag{A.6}$$

The beta function can be expressed using the gamma function as

$$B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}. \tag{A.7}$$

Using this and the fact that $\Gamma(n) = (n-1)!$ for integers n, (A.6) can be rewritten as

$$-\frac{\partial R(k,u,r)}{\partial u} = \frac{u^k}{B(k+1,r)(1+u)^{r+k+1}}. \tag{A.8}$$

Integrating with respect to u gives

$$-R(k,u,r) = \frac{1}{B(k+1,r)}\int_0^u \frac{a^k}{(1+a)^{r+k+1}}da + C. \tag{A.9}$$

Changing variables to

$$t = \frac{a}{1+a}, \quad a = \frac{t}{1-t}, \quad \frac{da}{dt} = \frac{1}{1-t} + \frac{t}{(1-t)^2} = \frac{1}{(1-t)^2} \tag{A.10}$$

yields

$$-R(k,u,r) = \frac{1}{B(k+1,r)}\int_0^{u/(1+u)} t^k(1-t)^{r+1}\frac{1}{(1-t)^2}dt + C. \tag{A.11}$$

Changing back to our original variable q from u gives

$$\begin{aligned} -R(k,q,r) &= \frac{1}{B(k+1,r)}\int_0^q t^k(1-t)^{r-1}dt + C = \\ &= \frac{B_q(k+1,r)}{B(k+1,r)} + C \end{aligned} \tag{A.12}$$

where

$$B_x(\alpha,\beta) = \int_0^x t^{\alpha-1}(1-t)^{\beta-1}dt \tag{A.13}$$

The constant C can be found by

$$\lim_{q\to 0^+} -R(k,q,r) = \lim_{q\to 0^+}\frac{B_q(k+1,r)}{B(k+1,r)} + C = C \tag{A.14}$$

$$\lim_{q\to 0^+} R(k,q,r) = \lim_{q\to 0^+}\sum_{i=0}^{k}\frac{\Gamma(i+r)}{i!\,\Gamma(r)}(1-q)^r q^i = 1 \tag{A.15}$$

since

$$\lim_{q\to 0^+} q^0 = \lim_{q\to 0^+} 1 = 1 \quad \text{and} \quad \lim_{q\to 0^+}\frac{\Gamma(i+r)}{i!\,\Gamma(r)}(1-q)^r q^i = 0 \quad \forall i > 0. \tag{A.16}$$

Hence $C = -1$ and so the CDF of the negative binomial distribution can be written as

$$R(k,q,r) = 1 - \frac{B_q(k+1,r)}{B(k+1,r)} = 1 - I_q(k+1,r) \tag{A.17}$$

where I_q is the regularized incomplete beta function as defined in (3.29).

A.2 Rewriting series

The series in (3.34) can be rewritten using the regularized incomplete beta function for $r+1$.

$$\sum_{l=t-s_i+1}^{\infty} l \cdot \frac{\Gamma(l+r^{(j)})(1-q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})} = \\ = \sum_{l=t-s_i+1}^{\infty} \frac{\Gamma(l+r^{(j)})(1-q^{(j)})^{r^{(j)}}(q^{(j)})^l}{(l-1)!\,\Gamma(r^{(j)})} \quad \text{(A.18)}$$

changing variables to $k = l-1$ gives

$$\sum_{k=t-s_i}^{\infty} \frac{\Gamma(k+r^{(j)}+1)(1-q^{(j)})^{r^{(j)}}(q^{(j)})^{k+1}}{k!\,\Gamma(r^{(j)})} = \\ = \frac{r^{(j)}q^{(j)}}{1-q^{(j)}} \sum_{k=t-s_i}^{\infty} \frac{\Gamma(k+r^{(j)}+1)(1-q^{(j)})^{r^{(j)}+1}(q^{(j)})^k}{k!\,\Gamma(r^{(j)}+1)}. \quad \text{(A.19)}$$

In the case $t-s_i > 0$ the sum above can be rewritten using the regularized incomplete beta function i.e. the CDF of the negative binomial function, this gives

$$\sum_{k=t-s_i}^{\infty} \frac{\Gamma(k+r^{(j)}+1)(1-q^{(j)})^{r^{(j)}+1}(q^{(j)})^k}{k!\,\Gamma(r^{(j)}+1)} = I_{q^{(j)}}(t-s_i, r^{(j)}+1). \quad \text{(A.20)}$$

The regularized incomplete beta function is only defined for arguments > 0. The case when $t-s_i = 0$ is the sum of a probability mass function for all possible values which simply equals 1. We end up with

$$\sum_{l=t-s_i+1}^{\infty} l \cdot \frac{\Gamma(l+r^{(j)})(1-q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})} = \\ = \frac{r^{(j)}q^{(j)}}{1-q^{(j)}} \begin{cases} I_{q^{(j)}}(t-s_i, r^{(j)}+1), & \text{if } t-s_i > 0 \\ 1, & \text{if } t-s_i = 0. \end{cases} \quad \text{(A.21)}$$

A.3 Rewriting integral

This section shows how to evaluate the integral in (3.39) for $k = k^{(j)}$

$$\int_{t-s_i}^{\infty} \tau^{k^{(j)}} \cdot \frac{k^{(j)}\tau^{k^{(j)}-1}e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}}e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}} d\tau = \\ = \frac{k^{(j)}}{e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}} \int_{t-s_i}^{\infty} \tau^{k^{(j)}} \cdot \frac{\tau^{k^{(j)}-1}e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}}} d\tau \quad \text{(A.22)}$$

With the variable change $u = (\tau/\lambda^{(j)})^{k^{(j)}}$ we get

$$\begin{aligned} \tau = t - s_i \implies u &= \left(\frac{t-s_i}{\lambda^{(j)}}\right)^{k^{(j)}} \\ \frac{d\tau}{du} = \frac{1}{k^{(j)}} u^{1/k^{(j)}-1}\lambda^{(j)} \implies d\tau &= \frac{1}{k^{(j)}} u^{1/k^{(j)}-1}\lambda^{(j)} du \end{aligned} \tag{A.23}$$

and the integral above becomes

$$\begin{aligned} &\int_{t-s_i}^{\infty} \tau^{k^{(j)}} \cdot \frac{\tau^{k^{(j)}-1} e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}}} d\tau = \\ &= \int_{((t-s_i)/\lambda^{(j)})^{k^{(j)}}}^{\infty} u \cdot \frac{(\lambda^{(j)})^{k^{(j)}} u e^{-u}}{u^{1/k^{(j)}}\lambda^{(j)}} \frac{1}{k^{(j)}} u^{1/k^{(j)}-1}\lambda^{(j)} du = \\ &= \frac{(\lambda^{(j)})^{k^{(j)}}}{k^{(j)}} \int_{((t-s_i)/\lambda^{(j)})^{k^{(j)}}}^{\infty} u e^{-u} du. \end{aligned} \tag{A.24}$$

This integral can easily be evaluated by hand using integration by parts. The following is obtained

$$\begin{aligned} \int_{((t-s_i)/\lambda^{(j)})^{k^{(j)}}}^{\infty} u e^{-u} du &= \left[-u e^{-u}\right]_{((t-s_i)/\lambda^{(j)})^{k^{(j)}}}^{\infty} + \int_{((t-s_i)/\lambda^{(j)})^{k^{(j)}}}^{\infty} e^{-u} du = \\ &= \left(\frac{t-s_i}{\lambda^{(j)}}\right)^{k^{(j)}} e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}} + e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}} = \\ &\left(\left(\frac{t-s_i}{\lambda^{(j)}}\right)^{k^{(j)}} + 1\right) e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}. \end{aligned} \tag{A.25}$$

This gives the final result as

$$\begin{aligned} &\int_{t-s_i}^{\infty} \tau^{k^{(j)}} \cdot \frac{k^{(j)} \tau^{k^{(j)}-1} e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}} e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}} d\tau = \\ &= \frac{k^{(j)}}{e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}}} \frac{(\lambda^{(j)})^{k^{(j)}}}{k^{(j)}} \left(\left(\frac{t-s_i}{\lambda^{(j)}}\right)^{k^{(j)}} + 1\right) e^{-((t-s_i)/\lambda^{(j)})^{k^{(j)}}} = \\ &= (t-s_i)^{k^{(j)}} + (\lambda^{(j)})^{k^{(j)}}. \end{aligned} \tag{A.26}$$

A.4 Variance

A.4.1 General time to return distribution

Since the observed Fisher information matrix is the negative Hessian of the observed log-likelihood, an expression for this is needed. The observed likelihood function for a general time to return distribution ρ with CDF R is shown in (3.15) and below.

$$\mathcal{L}(p, \boldsymbol{\theta} | \mathbf{t}^{(o)}, \mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}) = p^m \prod_{i=1}^{m} \rho(t_i^{(o)} | \boldsymbol{\theta}) \prod_{i=m+1}^{n} (1 - pR(t - s_i | \boldsymbol{\theta})). \tag{A.27}$$

This yields the following log-likelihood function for the observed data.

$$\ell(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}) = m \log(p) + \sum_{i=1}^{m} \log(\rho(t_i^{(o)}|\boldsymbol{\theta})) + \sum_{i=m+1}^{n} \log(1 - pR(t - s_i|\boldsymbol{\theta}))) \tag{A.28}$$

The elements of the observed Fisher information matrix become

$$\begin{aligned}
-\frac{\partial^2 \ell}{\partial p^2} &= \frac{m}{p^2} + \sum_{i=m+1}^{n} \left(\frac{R(t - s_i|\boldsymbol{\theta})}{1 - pR(t - s_i|\boldsymbol{\theta})} \right)^2 \\
-\frac{\partial^2 \ell}{\partial p \partial \boldsymbol{\theta}} &= \sum_{i=m+1}^{n} \frac{\frac{\partial}{\partial \boldsymbol{\theta}} R(t - s_i|\boldsymbol{\theta})}{(1 - pR(t - s_i|\boldsymbol{\theta}))^2} \\
-\frac{\partial^2 \ell}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathrm{T}}} &= -\sum_{i=1}^{m} \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathrm{T}}} \log(\rho(t_i^{(o)}|\boldsymbol{\theta})) + \\
&\quad + \sum_{i=m+1}^{n} \left(\frac{p \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathrm{T}}} R(t - s_i|\boldsymbol{\theta})}{1 - pR(t - s_i|\boldsymbol{\theta})} + \frac{p^2 \frac{\partial}{\partial \boldsymbol{\theta}} R(t - s_i|\boldsymbol{\theta}) \frac{\partial}{\partial \boldsymbol{\theta}^{\mathrm{T}}} R(t - s_i|\boldsymbol{\theta})}{(1 - pR(t - s_i|\boldsymbol{\theta}))^2} \right).
\end{aligned} \tag{A.29}$$

where we assume that $\boldsymbol{\theta}$ is a column vector containing the distribution parameters.

A.4.2 Negative binomial distribution

With a negative binomial distribution for the time to return we get the following ρ and R.

$$\rho(k|q, r) = \frac{\Gamma(k + r)}{k!\,\Gamma(r)} (1 - q)^r q^k \quad \text{and} \quad R(k|q, r) = 1 - I_q(k + 1, r) \tag{A.30}$$

When calculating the elements of the Hessian of the log-likelihood, it is intractable to let r vary. We therefore set r to be constant which gives us the following elements of the Hessian of the log-likelihood.

$$\begin{aligned}
\frac{\partial^2 \ell}{\partial p^2} &= -\frac{m}{p^2} - \sum_{i=m+1}^{n} \left(\frac{1 - I_q(t - s_i + 1, r)}{1 - p + pI_q(t - s_i + 1, r)} \right)^2 \\
\frac{\partial^2 \ell}{\partial p \partial q} &= \sum_{i=m+1}^{n} \frac{\frac{\partial}{\partial q} I_q(t - s_i + 1, r)}{(1 - p + pI_q(t - s_i + 1, r))^2} \\
\frac{\partial^2 \ell}{\partial q^2} &= -\frac{rm}{(1 - q)^2} - \frac{1}{q^2} \sum_{i=1}^{m} t_i^{(o)} + \\
&\quad + \sum_{i=m+1}^{n} \left(\frac{p \frac{\partial^2}{\partial q^2} I_q(t - s_i + 1, r)}{1 - p + pI_q(t - s_i + 1, r)} - \left(\frac{p \frac{\partial}{\partial q} I_q(t - s_i + 1, r)}{1 - p + pI_q(t - s_i + 1, r)} \right)^2 \right)
\end{aligned} \tag{A.31}$$

To evaluate $\frac{\partial}{\partial q} I_q$ we use the fundamental theorem of calculus

$$\frac{\partial}{\partial q} I_q(t - s_i + 1, r) = \frac{1}{B(1; t - s_i + 1, r)} \frac{\partial}{\partial q} \int_0^q x^{t-s_i} (1-x)^{r-1} dx = \frac{q^{t-s_i}(1-q)^{r-1}}{B(1; t - s_i + 1, r)} \tag{A.32}$$

which in turn can be used to evaluate $\frac{\partial^2}{\partial q^2} I_q$. The result is

$$\begin{aligned} \frac{\partial^2}{\partial q^2} I_q(t - s_i + 1, r) &= \frac{(t-s_i)q^{t-s_i-1}(1-q)^{r-1} - (r-1)q^{t-s_i}(1-q)^{r-2}}{B(1; t - s_i + 1, r)} = \\ &= \frac{q^{t-s_i-1}(1-q)^{r-2}((t-s_i)(1-q) - (r-1)q)}{B(1; t - s_i + 1, r)} \end{aligned} \tag{A.33}$$

With the elements of the Hessian of the log-likelihood function for the observed data the estimated covariance matrix can simply be calculated from $\Sigma = -(\ell'')^{-1}$ using the estimated parameters $\hat{p}$ and $\hat{q}$.

A.4.3 Weibull distribution

Letting a Weibull distribution describe the time to return, the following ρ and R will be used.

$$\rho(t|\lambda, k) = \frac{k}{\lambda} \left(\frac{t}{\lambda} \right)^{k-1} e^{-(t/\lambda)^k} \quad \text{and} \quad R(t|\lambda, k) = 1 - e^{-(t/\lambda)^k} \tag{A.34}$$

As was the case for the negative binomial distribution, it is not tractable letting k vary and we therefore let it be constant. Doing so yields the elements of the Hessian of the log-likelihood

$$\begin{aligned} \frac{\partial^2 \ell}{\partial p^2} &= -\frac{m}{p^2} - \sum_{i=m+1}^{n} \left(\frac{1 - e^{-((t-s_i)/\lambda)^k}}{1 - p + pe^{-((t-s_i)/\lambda)^k}} \right)^2 \\ \frac{\partial^2 \ell}{\partial p \partial \lambda} &= \sum_{i=m+1}^{n} \frac{\frac{\partial}{\partial \lambda} e^{-((t-s_i)/\lambda)^k}}{(1 - p + pe^{-((t-s_i)/\lambda)^k})^2} \\ \frac{\partial^2 \ell}{\partial \lambda^2} &= \frac{km}{\lambda^2} - \frac{k(k+1)}{\lambda^{k+2}} \sum_{i=1}^{m} (t_i^{(o)})^k + \\ &\quad + \sum_{i=m+1}^{n} \left(\frac{p \frac{\partial^2}{\partial \lambda^2} e^{-((t-s_i)/\lambda)^k}}{1 - p + pe^{-((t-s_i)/\lambda)^k}} - \frac{(p \frac{\partial}{\partial \lambda} e^{-((t-s_i)/\lambda)^k})^2}{(1 - p + pe^{-((t-s_i)/\lambda)^k})^2} \right) \end{aligned} \tag{A.35}$$

where

$$\frac{\partial}{\partial \lambda} e^{-((t-s_i)/\lambda)^k} = \frac{k}{\lambda} \left(\frac{t - s_i}{\lambda} \right)^k e^{-((t-s_i/\lambda))^k} \tag{A.36}$$

and

$$\frac{\partial^2}{\partial \lambda^2} e^{-((t-s_i)/\lambda)^k} = \frac{k}{\lambda^2} \left(\frac{t - s_i}{\lambda} \right)^k \left(\left(\frac{t - s_i}{\lambda} \right)^k k - k - 1 \right) e^{-((t-s_i/\lambda))^k}. \tag{A.37}$$

The covariance matrix can now be calculated from $\Sigma = -(\ell'')^{-1}$ with the estimated parameters $\hat{p}$ and $\hat{\lambda}$ as described in section 3.10.2.

A.5 Variance with prior

The elements of the observed Fisher information matrix are altered when using a prior. Since the prior in this case is only a function of p, the only entry that is affected is $\dfrac{\partial^2 \ell}{\partial p^2}$ which in the case of a beta distribution prior becomes

$$\frac{\partial^2 \ell}{\partial p^2} = -\frac{m}{p^2} - \sum_{i=m+1}^{n} \left(\frac{R(t - s_i|\boldsymbol{\theta})}{1 - pR(t - s_i|\boldsymbol{\theta})} \right)^2 - \frac{\alpha - 1}{p^2} - \frac{\beta - 1}{(1-p)^2}. \tag{A.38}$$

A.6 Covariates

The complete likelihood when modeling p using logistic regression is

$$\begin{aligned} \mathcal{L}(p, \boldsymbol{\theta}|\mathbf{t}^{(o)}, \mathbf{t}^{(u)}, \mathbf{z}, \boldsymbol{X}) &= \prod_{i=1}^{n} p(\boldsymbol{x}_i, \boldsymbol{\beta})^{z_i} \cdot \rho(t_i|\boldsymbol{\theta})^{z_i} (1 - p(\boldsymbol{x}_i, \boldsymbol{\beta}))^{(1-z_i)} = \\ &= \prod_{i=1}^{m} \frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{x}_i}} \rho(t_i^{(o)}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} \left(\frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{x}_i}}{1 + e^{\boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{x}_i}} \right)^{z_i} \cdot \rho(t_i^{(u)}|\boldsymbol{\theta})^{z_i} \left(\frac{1}{1 + e^{\boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{x}_i}} \right)^{(1-z_i)} \end{aligned} \tag{A.39}$$

where $\boldsymbol{X}$ is a matrix containing the covariates $\boldsymbol{x}_i$ for all units. When using the complete likelihood in (A.39) in the EM-algorithm it is not possible to maximize the Q-function with respect to the coefficient vector $\boldsymbol{\beta}$ analytically. Hence this has to be done numerically in each iteration.

After running the EM-algorithm and finding estimates of $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ it is possible to calculate the return rate for the units as a group. When calculating the return rate for the whole set of units $i = 1, \ldots, n$ with covariates $\boldsymbol{x}_i$ where $i = 1, \ldots, m$ has been returned it holds that

$$p = \frac{1}{n} \left(m + \sum_{i=m+1}^{n} Z_i \right) \tag{A.40}$$

hence p can be estimated as

$$\hat{p} = \mathbb{E}[p|\bullet, \boldsymbol{X}, \boldsymbol{\beta}] = \frac{1}{n} \left(m + \sum_{i=m+1}^{n} \mathbb{E}[Z_i|\bullet, \boldsymbol{X}, \boldsymbol{\beta}] \right). \tag{A.41}$$

where $\bullet$ denotes the conditional set $\{\mathbf{t}^{(o)}, \mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}, \boldsymbol{\theta}\}$.

The expectation of Z_i with covariates becomes, compare with (3.22),

$$\begin{aligned}\mathbb{E}[Z_i|\bullet, \boldsymbol{X}, \boldsymbol{\beta}] &= P(Z_i = 1|\mathbf{t}^{(o)}, \mathbf{T}^{(u)} > t \cdot \mathbf{1} - \mathbf{s}, \boldsymbol{\theta}, \boldsymbol{X}, \boldsymbol{\beta}) = \\ &= \frac{P(T_i^{(u)} > t - s_i|Z_i = 1, \boldsymbol{\theta}, \boldsymbol{X}, \boldsymbol{\beta})P(Z_i = 1|\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{X})}{\sum_{k=0}^{1} P(T_i^{(u)} > t - s_i|Z_i = k, \boldsymbol{\theta}, \boldsymbol{X}, \boldsymbol{\beta})P(Z_i = k|\boldsymbol{\theta}, \boldsymbol{X}, \boldsymbol{\beta})} = \\ &= \frac{(1 - R(t - s_i|\boldsymbol{\theta}))p(\boldsymbol{x}_i, \boldsymbol{\beta})}{1 - p(\boldsymbol{x}_i, \boldsymbol{\beta}) + p(\boldsymbol{x}_i, \boldsymbol{\beta})(1 - R(t - s_i|\boldsymbol{\theta}))}\end{aligned} \tag{A.42}$$

inserting this into (A.41) gives

$$\hat{p} = \frac{1}{n}\left(m + \sum_{i=m+1}^{n} \frac{(1 - R(t - s_i|\boldsymbol{\theta}))p(\boldsymbol{x}_i, \boldsymbol{\beta})}{1 - p(\boldsymbol{x}_i, \boldsymbol{\beta}) + p(\boldsymbol{x}_i, \boldsymbol{\beta})(1 - R(t - s_i|\boldsymbol{\theta}))}\right). \tag{A.43}$$

A.7 Missing data

A.7.1 Negative binomial distribution

For the negative binomial distribution the expected value that must be calculated is, similar to (3.33),

$$\begin{aligned}\gamma_i^{(j)} &= \mathbb{E}\left[T_i|Z_i = 1, T_i \leq r_i, p^{(j)}, q^{(j)}, r^{(j)}\right] = \\ &= \sum_{l=0}^{\infty} l \cdot P(T_i^{(u)} = l|Z_i = 1, T_i \leq r_i, p^{(j)}, q^{(j)}, r^{(j)}) = \\ &= \sum_{l=0}^{\infty} l \cdot \frac{P(T_i = l, T_i \leq r_i|Z_i = 1, p^{(j)}, q^{(j)}, r^{(j)})}{P(T_i \leq r_i|Z_i = 1, p^{(j)}, q^{(j)}, r^{(j)})}\end{aligned} \tag{A.44}$$

Inserting the negative binomial distribution (3.28) gives

$$\gamma_i^{(j)} = \frac{1}{1 - I_q(r_i + 1, r^{(j)})} \sum_{l=0}^{r_i} l \cdot \frac{\Gamma(l + r^{(j)})(1 - q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})}. \tag{A.45}$$

We can use the fact that the sum in (A.45) is simply a partial sum of the expected value

$$\begin{aligned}&\sum_{l=0}^{r_i} l \cdot \frac{\Gamma(l + r^{(j)})(1 - q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})} = \\ &= \sum_{l=0}^{\infty} l \cdot \frac{\Gamma(l + r^{(j)})(1 - q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})} - \sum_{l=r_i+1}^{\infty} l \cdot \frac{\Gamma(l + r^{(j)})(1 - q^{(j)})^{r^{(j)}}(q^{(j)})^l}{l!\,\Gamma(r^{(j)})}.\end{aligned} \tag{A.46}$$

From the calculations in appendix A.2 we know how to calculate the sum from $l = r_i + 1$ to $l = \infty$ and the first sum is simply the expected value which is $\frac{q^{(j)}r^{(j)}}{1-q^{(j)}}$, which gives

$$\gamma_i^{(j)} = \frac{q^{(j)}r^{(j)}}{1 - q^{(j)}} \frac{1 - I_q(r_i, r^{(j)} + 1)}{1 - I_q(r_i + 1, r^{(j)})} \tag{A.47}$$

assuming $r_i > 0$, if $r_i = 0$ the expected value is trivially equal to 0.

This gives the following updating formula for q, compare with (3.35),

$$q^{(j+1)} = \frac{\sum_{i=1}^{o} t_i^{(o)} + \sum_{i=o+1}^{m} \gamma_i^{(j)} + \sum_{i=m+1}^{n} \mu_i^{(j)} \cdot \pi_i^{(j)}}{r^{(j)} m + \sum_{i=1}^{o} t_i^{(o)} + \sum_{i=o+1}^{m} \gamma_i^{(j)} + \sum_{i=m+1}^{n} (\mu_i^{(j)} + r^{(j)}) \cdot \pi_i^{(j)}}. \tag{A.48}$$

A.7.2 Weibull distribution

For the Weibull distribution the expected value that must be calculated is, similar to (3.39),

$$\begin{aligned} \gamma_i^{(j)} &= \mathbb{E}\left[(T_i)^{k^{(j)}} \,\middle|\, Z_i = 1, T_i \leq r_i, p^{(j)}, k^{(j)}, \lambda^{(j)} \right] = \\ &= \int_0^\infty \tau^{k^{(j)}} \cdot P(T_i = \tau | Z_i = 1, T_i \leq r_i, p^{(j)}, k^{(j)}, \lambda^{(j)}) d\tau = \\ &= \int_0^\infty \tau^{k^{(j)}} \cdot \frac{P(T_i = \tau, T_i \leq r_i | Z_i = 1, p^{(j)}, k^{(j)}, \lambda^{(j)})}{P(T_i \leq r_i | Z_i = 1, p^{(j)}, k^{(j)}, \lambda^{(j)})} d\tau. \end{aligned} \tag{A.49}$$

Inserting the Weibull distribution (3.36) gives

$$\gamma_i^{(j)} = \frac{1}{1 - e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}} \int_0^{r_i} \tau^{k^{(j)}} \cdot \frac{k^{(j)} \tau^{k^{(j)}-1} e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}}} d\tau \tag{A.50}$$

Similar as for the negative binomial distribution we can use the result in A.3

$$\begin{aligned} &\int_0^{r_i} \tau^{k^{(j)}} \cdot \frac{k^{(j)} \tau^{k^{(j)}-1} e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}} e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}} d\tau = \\ &= \int_0^\infty \tau^{k^{(j)}} \cdot \frac{k^{(j)} \tau^{k^{(j)}-1} e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}} e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}} d\tau - \int_{r_i}^\infty \tau^{k^{(j)}} \cdot \frac{k^{(j)} \tau^{k^{(j)}-1} e^{-(\tau/\lambda^{(j)})^{k^{(j)}}}}{(\lambda^{(j)})^{k^{(j)}} e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}} d\tau = \\ &= \frac{(\lambda^{(j)})^{k^{(j)}}}{e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}} - (r_i^{k^{(j)}} + (\lambda^{(j)})^{k^{(j)}}) = (\lambda^{(j)})^{k^{(j)}} (e^{(r_i/\lambda^{(j)})^{k^{(j)}}} - 1) - r_i^{k^{(j)}} \end{aligned} \tag{A.51}$$

which gives

$$\gamma_i^{(j)} = \left((\lambda^{(j)})^{k^{(j)}} (e^{(r_i/\lambda^{(j)})^{k^{(j)}}} - 1) - r_i^{k^{(j)}} \right) \frac{e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}}{1 - e^{-(r_i/\lambda^{(j)})^{k^{(j)}}}}. \tag{A.52}$$

This gives the following updating formula, compare with (3.41)

$$\lambda^{(j+1)} = \left(\frac{\sum_{i=1}^{o} (t_i^{(o)})^{k^{(j)}} + \sum_{i=o+1}^{m} \gamma_i^{(j)} + \sum_{i=m+1}^{n} \mu_i^{(j)} \cdot \pi_i^{(j)}}{m + \sum_{i=m+1}^{n} \pi_i^{(j)}} \right)^{1/k^{(j)}}. \tag{A.53}$$

A.8 Results for other products

The results of running the EM-algorithm with a negative binomial distribution with a constant r, i.e. the method which on average gave the lowest error, for products B-H is shown below in figures A.1 - A.7.

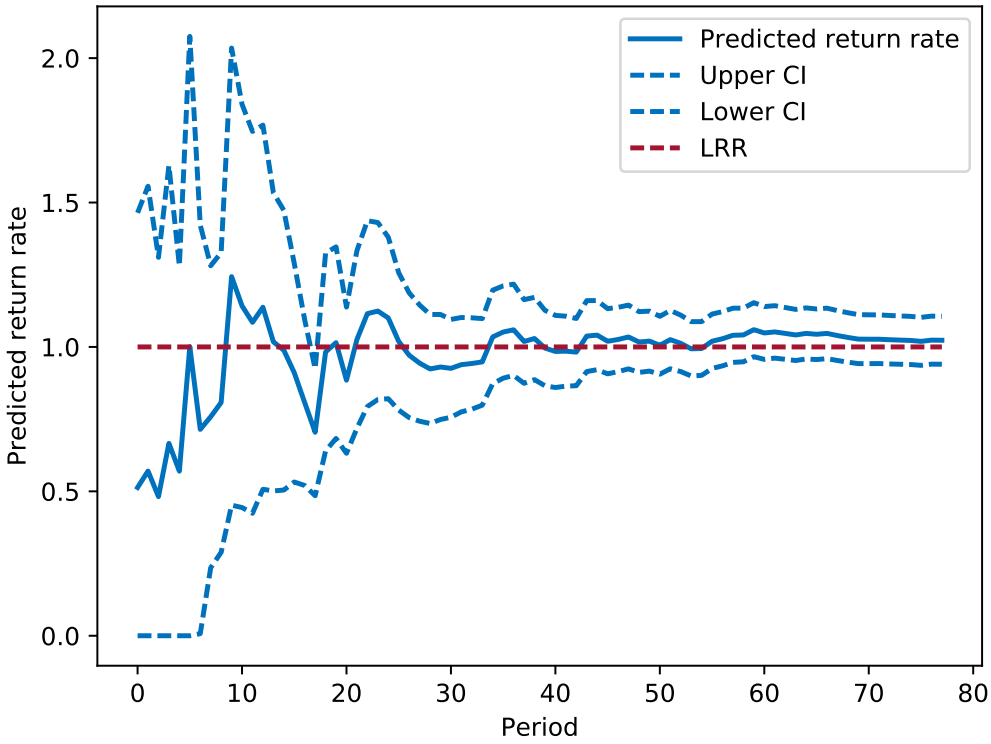

FIGURE A.1: Predicted return rate for product B using the optimal distribution together with approximate 95% confidence intervals.

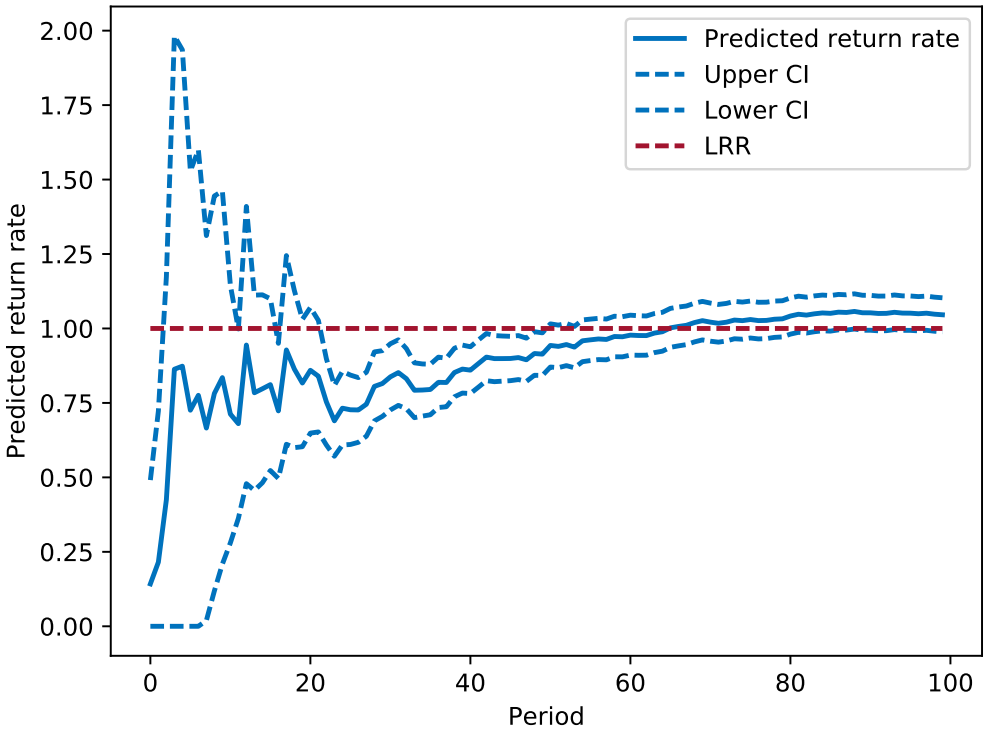

FIGURE A.2: Predicted return rate for product C using the optimal distribution together with approximate 95% confidence intervals.

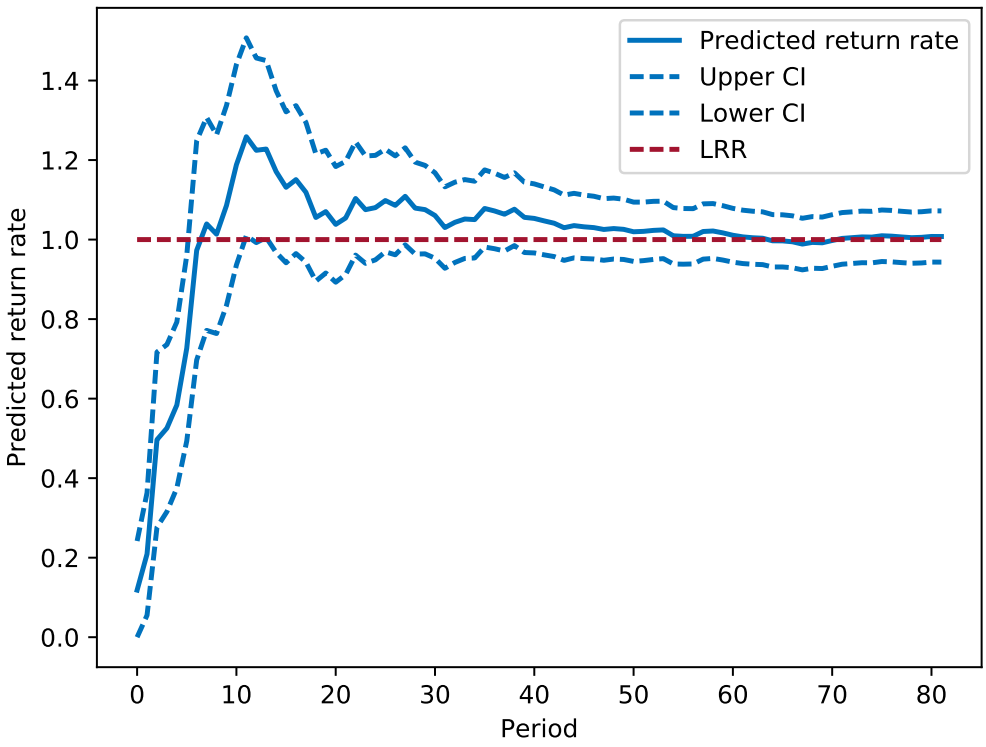

FIGURE A.3: Predicted return rate for product D using the optimal distribution together with approximate 95% confidence intervals.

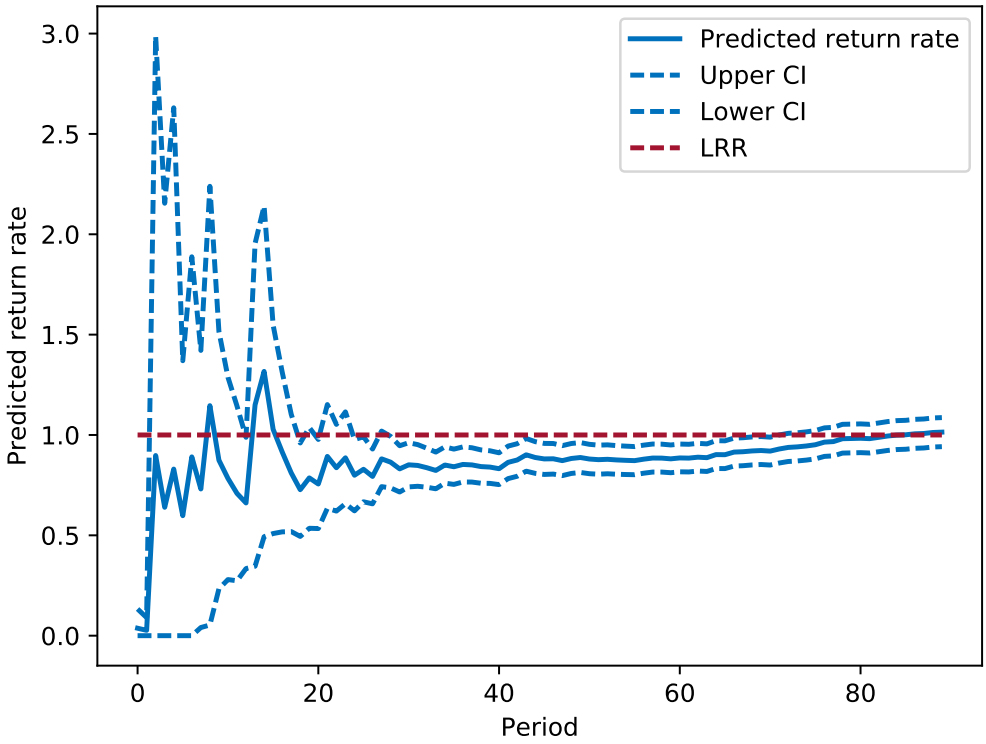

FIGURE A.4: Predicted return rate for product E using the optimal distribution together with approximate 95% confidence intervals.

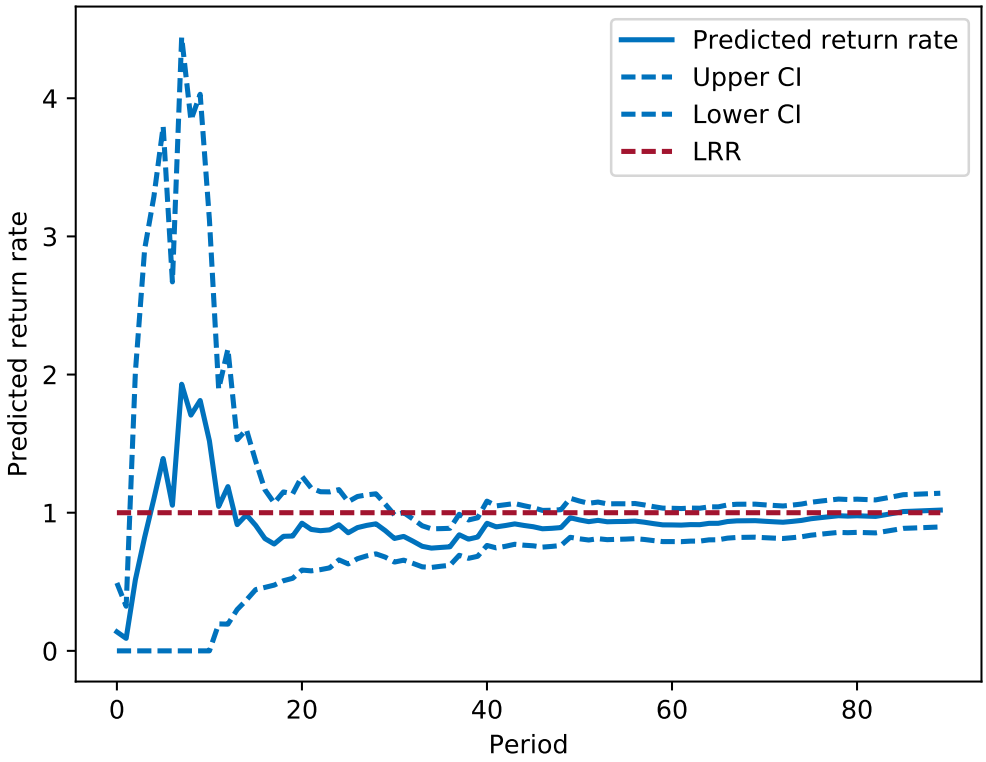

FIGURE A.5: Predicted return rate for product F using the optimal distribution together with approximate 95% confidence intervals.

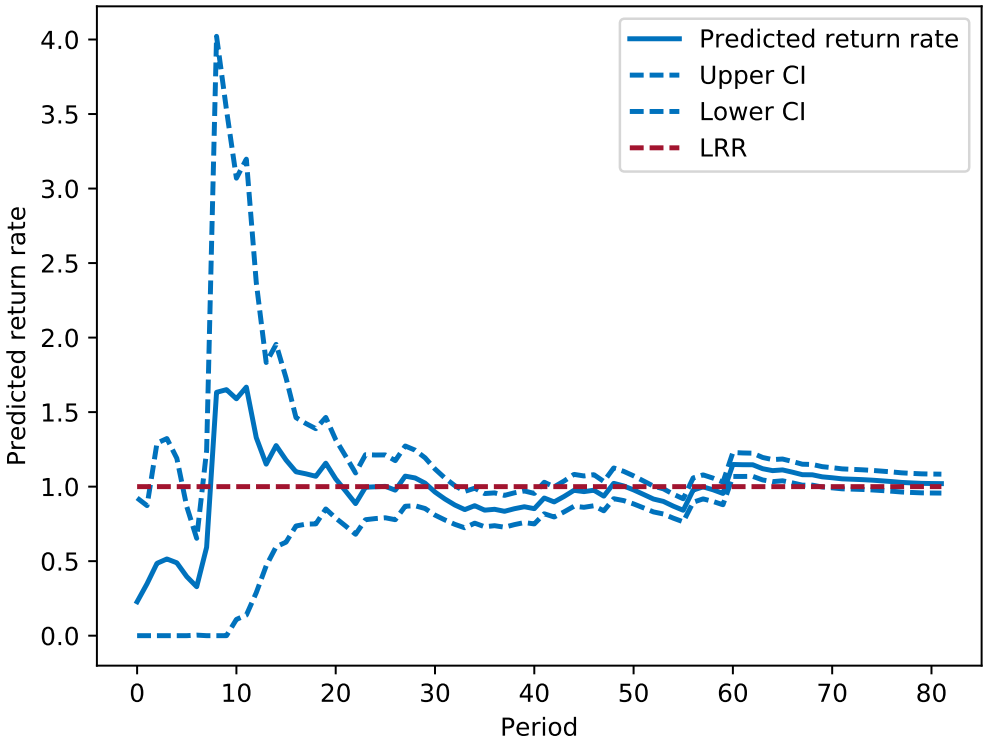

FIGURE A.6: Predicted return rate for product G using the optimal distribution together with approximate 95% confidence intervals.

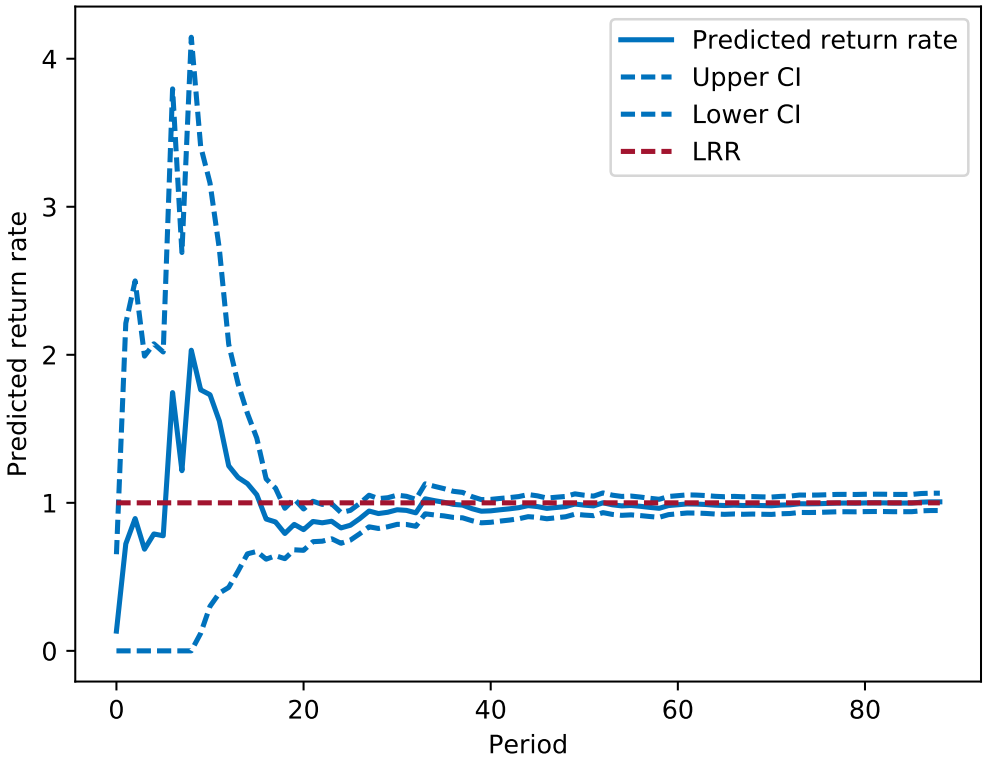

FIGURE A.7: Predicted return rate for product H using the optimal distribution together with approximate 95% confidence intervals.

Return Rate Prediction

Erik Karlén and Caspar Welin

A major concern for every company is the quality of their products. Predicting the lifetime return rate allows for identification of products with atypically high return rates. Furthermore, predicting a product's return rate allows for estimation of the company's expected cost of handling returns. We explored multiple methods for predicting a product's lifetime return rate and show the results of applying them to real-life data.

Faulty units is a problem that all companies selling physical products have to deal with. Selling such units may lead to dissatisfied customers, which in turn leads to less revenue. To avoid this companies often have a return policy to make up for it and normally the defective unit will be either repaired or replaced. Handling returns can cost a lot of money for a company, especially if the return rate is high. It is therefore in the company's best interest to minimize the number of faulty units sold.

To reduce return rates it is important to as early as possible in a product's life span identify if there are any issues with the product. This, however, is not a simple task. One way this can be accomplished is by predicting a product's lifetime return rate, i.e. the probability of a unit being returned, early in the product's life span. Comparing this prediction with the usual return rates of the company's products may give an indication of whether there is a problem with the product.

This project was carried out in cooperation with Axis Communications AB, a company that mainly designs and produces network cameras. They contributed by letting us analyze the data they have collected regarding sold and returned units.

Results

A simple way of estimating the lifetime return rate of a product is by calculating the fraction of returns observed so far. This is the blue line in figure 1 and is called the aggregated return rate. As can be seen in the figure it takes many periods before it is a decent estimation of the lifetime return rate. The other curves in the figure show some of the more successful methods we used to predict the return rate. They are all superior to the aggregated return rate. Our best performing method

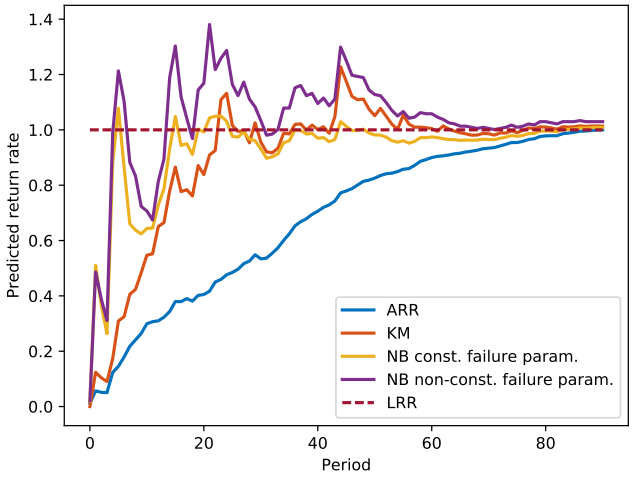

Fig. 1. The figure shows the predicted return rates calculated using the aggregated return rate (ARR), the Kaplan-Meier method (KM), the EM-algorithm (NB const. r) constant r (NB const. r) and with a non-constant r with prior (NB non-const. r). Each period is 30 days long.

(NB const. failure param.) was on average three times better than the aggregated return rate when using our error measure which is based on the deviation from the lifetime return rate in each period.

Model

The most successful method utilizes concepts from survival analysis which is commonly used to analyze time to event data, e.g. time to death in medical applications. Since only a portion of the units experience the event, which in our case means being returned, we used a so called cure model. With this we could estimate the time to return distribution for a product and the probability that a unit will be returned.

Implementation

In order to predict the return rate for all of Axis' products periodically we developed a Python script. This script automatically goes through Axis' databases and calculates predictions for all products. The predictions can then be imported to a data visualization program to get an overview of Axis' products. This gives an indication of which products are currently performing well and which may need to be investigated further.

Master's Theses in Mathematical Sciences 2017:E46
ISSN 1404-6342

LUTFMS-3329-2017

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/