ERASMUS UNIVERSITY ROTTERDAM
Erasmus School of Economics
Bachelor Thesis Econometrics & Operational Research

Comparing spare parts demand forecasting models

Name student: Aron van Woerkom
Student ID number: 480749

Abstract

Controlling inventory properly is essential to many large industrial corporations. Good inventory policy can save huge costs to these companies, while stockouts at the same time can be destructive to the all-day operations of the company. This research compares the performances of several methods that can predict the spare parts demand while making use of data from a large oil refinery. All of these models need to be capable of handling very intermittent demand realizations, which is a key feature in spare parts demand data. In this paper I propose a new forecasting method that does outperform formerly introduced methods for this dataset.

Supervisor: dr. Oktay Karabag
Second Assessor: prof.dr.ir. Rommert Dekker
Date final version: July 3, 2020

Contents

1 Introduction

Predicting spare parts demand successfully is of great importance for several industry and transport sectors of the economy. Companies operating in these sectors usually dispose of huge inventories with lots of different kinds of spare parts in their storage houses. Maintaining all these storage houses with high inventory levels generally results in very high costs for these companies. This is a consequence of the usually very expensive spare parts needed for a company's operations, which results in an unnecessary high amount of capital stored in the warehouses. The demand of spare parts at these companies is often intermittent, meaning that there exist usually big periods with zero demands between demand realizations. In addition, there is often a great variation in the values of the nonzero demand realizations. When controlling a company's inventory, providing accurate forecasts of the spare parts demand is very important [Silver et al., 1998]. However, the intermittent nature of the realized demands can make it very hard to provide the forecasts [Watson, 1987]. Additionally, another problem that needs to be addressed is the fact that not every item has the same importance for a company's operations. For some parts a lack of availability in the company would be destructive to the all-day operations, while a lack of availability for other parts would not have a very big effect, which results in different policies for the spare parts. Therefore, also making a distinction between spare parts on this aspect, might leave space open for savings.

For these aforementioned business problems, this paper tries to provide accurate models for forecasting spare parts demand. In total, the performances of seven different models are evaluated. All these methods are defined through three research questions. Four out of the seven models have been applied by Porras and Dekker [2008]. The optimization results of these four models are going to be replicated while using the same dataset of an oil refinery in the Netherlands. This replication part is the first part of this paper and the second part consists of new implemented models. In the end a conclusion on the best model will be drawn by comparing the performances of all the seven models. Similar to the paper of Porras and Dekker [2008], I make use of an ex-post approach, which means that the same dataset is used for fitting and testing. This is done due to the limited number of observations (see Section 3).

The first research question is about replicating the four models that have been implemented by Porras and Dekker [2008]. These models are a normal based model, a bootstrap model, an empirical model and a Poisson model. Next, also the performance of Croston's method will be evaluated when answering the first research question. The reason for picking Croston's method is that it is based on the same distribution of the best model of Porras and Dekker [2008]. Their best model, the normal based model, was based on the normal distribution. The first research question is as follows:

What are the forecasting performances of the normal based model, the bootstrap model, the empirical model, the Poisson model and Croston's method?

Secondly, it will be investigated whether the normal based model for modeling lead time demand can be improved. Porras and Dekker [2008] use in their normal based model a normal distribution for modeling the lead time demand (LTD) distribution. Since this distribution is not truncated on the left and the right, this distribution allows in the tails for lead time demands that are not feasible. I will try to improve this by imposing a truncation for modeling the LTD distribution. The reason why specifically this model is chosen, is because this model turned out to be the most successful one, out of the four models tested. The question regarding improving the normal based model is as follows:

Can there be improvements made on a normal based model for modeling lead time spare parts demand in an oil refinery by making use of a truncated normal distribution?

The last research question that I try to give an answer to in this paper is about improving the Willemain's bootstrap model that has been applied by Willemain et al. [2004]. In their paper they compared the performances of this method to Croston's method and exponential smoothing and concluded that it provided the best results. However, Porras and Dekker [2008] concluded in their research that the bootstrap model was outperformed by the normal based model, which was not suspected beforehand. The bootstrap model even resulted in negative savings for items with high nonzero demand realizations. Therefore, this method is going to be applied to items which had a total demand higher than 60 (see Section 3 for the division of items into classes). Another reason for trying to improve the forecasts of these items, is that high demands can possibly sooner result in stockouts. For an exact description of the revised Willemain's bootstrap model I refer to Section 4.7. The third research question is the following:

Can there be improvements made on Willemain's bootstrap model for modeling lead time spare parts demand in an oil refinery for spare parts with high demands?

In Section 2 after the introduction a discussion of the literature regarding this topic is provided. Section 3 is the data section in which the data used for this research is described and visualized. Furthermore, the necessary extractions and modifications to the data are reported. Next, in Section 4 all used methods and models are described. Also, I describe in this section how exactly I will come to the results by using an optimization process. These results are reported in Section 5. Afterwards, there is also a conclusion in Section 6, in which also the limitations and implications of the research are discussed.

2 Literature

In this literature section I discuss several models that have been implemented regarding forecasting intermittent demand. Therefore, only papers making use of these models and intermittent demand data are selected. When reviewing literature about forecasting spare parts demand, it needs to be found out which models have been implemented with what degree of success. In the next paragraphs models appearing in the literature for forecasting spare parts demand will be discussed.

Croston's method has been used successfully in forecasting intermittent demands in multiple researches. Croston's method makes separate exponential smoothing forecasts of nonzero demand sizes and the intervals between demands, then it divides these to forecast average demand per period [Willemain et al., 1994]. For using Croston's method there are made assumptions about normality [Croston, 1972]. Willemain et al. [1994] applied Croston's method to real data that violated these assumptions. It was concluded that Croston's method does still give satisfying results, in both intermittent simulated data and in intermittent real industrial data. Furthermore, it was found out that in both cases Croston's method outperformed the exponential smoothing method. The data used in my research does clearly not satisfy any normality assumption. However, as has been shown by Willemain et al. [1994] this does not have a negative impact on the forecasting performance, relative to other methods.

In spite of the superiority of Croston's method to exponential smoothing, exponential smoothing is probably still the most used of statistical approaches to forecasting intermittent demand [Willemain et al., 2004].

Furthermore, exponential smoothing seems to be the most robust forecasting method of all statistical approaches. Willemain et al. [2004] compare exponential smoothing, Croston's method and bootstrapping while applying these techniques to nine large industrial datasets. It was concluded that the bootstrapping method was the most successful method for forecasting the intermittent demands in these datasets. Although the bootstrapping method has outperformed Croston's method, I still expect Croston's method to work well for this dataset because it is relying on the normal distribution. Moreover, the best model by Porras and Dekker [2008] also relies on the normal distribution, while I use the same dataset as they use. The bootstrapping method by Willemain et al. [2004] was used to predict lead time demand for spare parts, by first predicting zero and nonzero observations for data points within the predicted lead time. Afterwards, the values of the nonzero observations are predicted by sampling with replacement from the nonzero demands and by using a jittering procedure. This jittering procedure allows for making point forecasts that deviate a bit from historical demands, which improves forecasting performance.

The same bootstrapping method was implemented in the research of Porras and Dekker [2008]. They compare four different models to estimate the lead time demand distribution of spare parts with data from an oil refinery. The other three models were based on three distributions: on an empirical distribution (derived from the dataset), a normal distribution and a Poisson distribution. It was concluded that the demand modeling based on the normal distribution provides the best results, although it was expected that the bootstrap method would provide the best results. This was expected because Willemain et al. [2004] concluded that bootstrap outperformed Croston's method and exponential smoothing. Furthermore, the lead time demand was clearly not normally distributed. However, still this normal distribution produced the most satisfying results out of the four models tested. Thus, similarly to estimating with Croston's method shown by Willemain et al. [2004], the items do not need to have an underlying normal distribution for the demand data to use this method. Another model that can be used is the model based on the Poisson distribution [Feeney and Sherbrooke, 1966]. This model assumes that the interarrival times between demands follow an exponential distribution. The only parameter needed for the demand distribution is the average lead time demand. Porras and Dekker [2008] only focus on demands with all nonzero demands equal to one, when implementing this model. I will also follow this approach when this model is replicated.

Additionally, there is a wide range of other forecasting methods available for forecasting spare parts demand. For instance, Ghobbar and Friend [2003] compare 13 forecasting methods for forecasting spare parts demand in the aviation industry. They conclude that the weighted moving average method and Croston's method outperform seasonal regression models and exponential smoothing. Next, Zhu et al. [2017] propose applying extreme value theory to model the tail of the lead time demand distribution. It is shown that the method improves the inventory performance compared to the empirical method and is competitive with the WSS method, Croston's method and SBA for a range of demand distributions. However, the method has problems with forecasting items with limited demand history. In contrast, the method's performance is better when very few realizations are observed, but over a large time horizon. In the case of limited demand history, the method based on extreme value theory is outperformed by Croston's method, as concluded by Zhu et al. [2017]. Since all items in my dataset only have 55 observations, extreme value theory is not used in this research.

To conclude, it is concluded in many papers that exponential smoothing is outperformed by some other method. Therefore, exponential smoothing is not going to be evaluated in this research. Next, I observe that Croston's method was outperformed by the bootstrap method, but there is not much literature available about this bootstrap method. In addition, it can be seen in the literature that before the introduction of the bootstrap method, Croston's method is recommended a lot. For instance, Silver et al. [1998] also recommends to use Croston's method for forecasting intermittent demand. Additionnally, Zhu et al. [2017] show that Croston's method is satisfactory for items with limited demand history compared to extreme value theory, which is the case in this research with only 55 observations for each item. This is the first reason for me to extend the research of Porras and Dekker [2008] with implementing Croston's method. The second reason for implementing Croston's method is that it is based on the normal distribution, as mentioned before. The model based on the normal distribution did outperform the bootstrap method in the paper of Porras and Dekker [2008]. Hence, I will try to improve this normal based model even further. Thus, improving this normal based model by making use of a truncated normal distribution is the second extension. The third extension is implementing another approach of Willemain's bootstrap method, as has been defined in the introduction. Thus, the extra contribution provided by this paper is implementing Croston's method on this dataset, using a truncated normal distribution to estimate demand and implementing a new bootstrap method.

3 Data

The dataset available for this research contains several kinds of information for each spare part at the oil refinery. For each of the 14,523 observations there are monthly demand realizations visible from January 1997 until August 2001. Next, for each item its delivery time (in days), its price and its criticality class is stated in the dataset. There are three types of criticality an item can have, which is indicating how important the availability of the item is to the company's operations. The following types exist:

- Low (L): Unavailability of such an item does not cause any significant problems for the operations or for the safety of the people and environment.
- Medium (M): When an item is unavailable from this class, it would result in a loss of production. This would, however, not result in a dangerous situation for the people or the environment.
- High (H): Unavailability of an item in this class would result in high costs or could cause danger to the safety of the people and the environment.

There are items which are related to several pieces of equipment, meaning that they have a joint criticality node.

3.1 Classification of spare parts

Porras and Dekker [2008] state that important differences among spare parts data exist also in demand and price, next to criticality nodes. They found this out in a more refined analysis of the spare parts data. Therefore, I also group the items into different classes depending on criticality, price and demand. This is done in the same way as Porras and Dekker [2008] implement it.

3.1.1 Criticality class

The first feature that determines the class of an item is criticality. It can happen that an item is related to several sorts of equipment. This means that it has several criticality nodes. This will be taken into account when defining the six criticality classes. The first criticality class consists of items which have the node H.

The second class contains items which have all three nodes H, M and L or only H and L. The third class only contains items which have the node M. The fourth class consists of items which have both an M and L node. The fifth class contains items which have only an L node. The last and sixth class contains items that do not have any criticality node at all. In Table 1 some insights about each criticality class are given.

Table 1: Criticality classes of spare parts

Criticality class	1	2	3	4	5	6
Specification	H	H/M/L or H/L	M	M/L	L	None
Spare parts (total = 14523)	8.2 %	7.8 %	38.9 %	13.2	5.9 %	26.0 %
Average demand	0.65	0.52	0.22	0.27	-0.02	0.71

3.1.2 Demand classes

Items can also be grouped according the size of their demands observed in the data. Porras and Dekker [2008] motivate their decision to also take into account this feature by showing that there exist huge differences in the sizes of the demand realizations. For instance, some items only have 0/1 demands while others observe demands of more than 300 in one single month. This means that the actual forecasting performances vary significantly between these classes. When dividing the items into classes with respect to demand, the first demand class consists of items only having 0/1 demand realizations. The second class consists of items with a total demand between 0 and 60 and not only 0/1 demands. The third class contains items with a total demand higher than 60. The fourth and last demand class contains items which had -1/0/1 demands. In Table 2 some insights about each demand class are given.

Table 2: Demand classes of spare parts

Demand class	1	2	3	4
Spare parts (total = 14523)	35.4 %	48.8 %	4.8 %	9.0 %
Average demand	0.03	0.20	5.74	0.01

3.1.3 Price classes

Also for the price there is a distinction made to group the items in the classes. See Table 3 for the divisions made.

Table 3: Price classes of spare parts

Price class	1	2	3	4	5
Price (p) in euros	$p = 0$	$0 \leq 13.6$	$13.6 < p \leq 169$	$169 < p \leq 2112$	$p > 2112$
Spare parts (total = 14523)	11 %	22.6 %	33.0 %	28.4 %	5.0 %
Average demand	-0.09	1.17	0.21	0.08	0.04

3.1.4 Combined classes

After I identified for each item to which class it belonged for each feature, these criticality, demand and price classes are combined to create a combined class '*abc*.' This can be written as a number consisting of three integers. So a refers to a demand class (class 1 till 4), b refers to a criticality class (class 1 till 6) and c refers to a price class (class 1 till 5). Grouping all the items into several classes allows me to optimize the whole system on a class level. Porras and Dekker [2008] have combined the classes in the same way. Therefore, using this way of combining them makes it easy to compare my results with theirs.

3.2 Data cleaning process, cost structure & lead times

In this research the exact same implementation for cleaning the data is used as has been done by Porras and Dekker [2008]. This is described in Appendix A. In the original paper the dataset contains 14,383 items. However, my dataset contains 14,523 items. After cleaning the data, I still have 8,610 items, whereas Porras and Dekker [2008] have 8,494 items. This deviation of approximately 1 % does, however, not result in different conclusions. Nevertheless, there could be a bias in the sense that I have some extra items that all belong to a small number of classes. This could make the results differ for some certain classes.
In this research there is made use of lead times and costs to estimate the lead time demand distribution. The lead times available in the dataset are provided in days. Porras and Dekker [2008] convert these lead times into months since the demands are also specified on a monthly basis. However, they do not provide a specific rounding procedure. From their results it appears to be the case that they ceil the lead times, so a lead time of 31 days results in a lead time of 2 months. This rounding procedure is also applied in my paper.
When controlling inventory, usually three types of costs appear, these are: holding costs, ordering costs and stockout costs. Since there is no information available about stockout costs, these are left out in this research. To calculate the holding costs, I use a 25 % annual fixed rate. This means that holding an item of price p costs $\frac{1}{4} * \frac{1}{12} * p = 0.02p$ per unit on a monthly basis. This is used when calculating costs in the system. The ordering costs are also considered and these are set equal to 36 euros per order, which is independent of the number of items ordered.

4 Methodology

In this section all the used methods of this research are reported. The core of the methodology consists of the forecasting methods that are used for estimating the lead time demand distribution (LTD). In contrast to Porras and Dekker [2008], I only use an ex-post approach, which means that the same dataset is used for fitting and testing purposes. I made this decision because there are only 55 observations available for each item included in the research. Two types of performances for each method are considered in this research, which are the fill rate and the cycle service level (CSL). The fill rate is the number of items that can be delivered out of own inventory and the CSL is the number inventory cycles without stockout. An inventory cycle is defined as the period between the placement of an order and the arrival of an order.
The structure of the rest of the methodology is as follows. First, the four replicated forecasting methods from the paper of Porras and Dekker [2008] are described. These methods are Willemain's bootstrap method, the normal distribution, the empirical distribution and the Poisson distribution. Second, the newly implemented forecasting methods are described. These are Croston's method, the truncated normal distribution and the revised Willemain's bootstrap method. After all the methods have been described, these methods will

be implemented. The goal of each method is to provide a re-order point for each item, meaning that if the inventory level is on that point, a new order is placed. The methods do so by estimating a certain distribution and then the re-order point is chosen in such a way that a specified fill rate is met theoretically according to its corresponding distribution.

4.1 Willemain's bootstrap method

The Willemain's bootstrap method that is used in this research was originally developed by Willemain et al. [2004]. This method estimates the lead time demand distribution by making point forecasts for each item during its lead time. To do so, it uses a two state, first order Markov process. It first forecasts for each demand during the lead time whether it is going to be zero or nonzero. These forecasts are depending on whether the last demand was zero or nonzero. After the forecasts have been provided, the method implements a jittering procedure for forecasting the values of the nonzero demands. This jittering procedure can cause a deviation in the point forecasts, which improves accuracy according to Willemain et al. [2004]. A more thorough description of this method is given in Appendix B.1.

In the end, this method calculates a CDF $F(x)$, with $F(x)$ equalling the probability that the lead time demand is smaller than x. To determine a re-order point s for this research, from the calculated CDF $F(x)$, the expected units short $ES(s)$ needs to be calculated iteratively for different values of s. The formula for $ES(s)$ is as follows:

$$ES(s) = \sum_{x|x>s} (x - s)f(x) \tag{1}$$

Then this calculated $ES(s)$ must be satisfying the following condition, to obtain a minimum theoretical fill rate (according to the estimated distribution) of $0 \leq \beta \leq 1$:

$$\beta \leq (1 - \frac{ES(s)}{Q}) \tag{2}$$

Q is the lot size and the calculation of this is provided in Section 4.8. It holds that $1 - \frac{ES(s)}{Q}$ is equal to the calculated fill rate for a given s. Furthermore, it can be observed that $ES(s)$ is decreasing in s and it will eventually reach 0. Therefore, it holds that at some point equation (2) is going to be met.

4.2 Empirical distribution

The next replicated method is the method based on the empirical distribution. Porras and Dekker [2008] construct for each item an empirical CDF for the demand during the lead time. These probabilities in the CDF are calculated from demands during the lead time without sampling. A histogram is created that does contain all lead time demands that have occurred during the period described in the dataset. From this histogram the CDF $F(x)$ is created for the probability that the demand during the lead time is smaller than x. Porras and Dekker [2008] say that since demands are directly taken from the dataset over fixed periods of time equal to the lead time, this method also captures autocorrelations and fixed demand intervals due to preventive maintenance. Moreover, this method is easier to implement than the Willemain's bootstrap method. When determining a re-order point s for this method with a confidence level of β, the same procedure as with Willemain's bootstrap method is applied, making use of equations (1) and (2) in Section 4.1.

4.3 Normal distribution

The next approach of estimating the lead time demand distribution uses the normal distribution. According to Porras and Dekker [2008] the parameters $\mu_{LTD} = \bar{D} * L$ and $\sigma_{LTD} = SD * \sqrt{L}$ from the data are calculated for the normal distribution. These are calculated simply from the demand values. Here L is the lead time in months, $\bar{D}$ is the average demand per month and SD is the standard deviation of the demand. $\bar{D}$ and SD are evaluated using all values for each item.

For seeking an appropriate fill rate s, the smallest unit normal variate z must be found that meets the following condition:

$$\beta \leq 1 - \frac{\sigma_{LTD} * UNLI(z)}{Q} \tag{3}$$

In this equation β is the obtained fill rate by the current system for the class the specific item belongs to and Q is the lot size. $UNLI(z)$ is the unit normal loss integral associated with the unit normal variate z. This is: $\int_z^\infty (t-z)\phi(t)dt$. The following derivation for $UNLI(z)$ is made use of when computing:

$$\int_z^\infty (t-z)\phi(t)dt = \int_z^\infty t\phi(t)dt - z(1-\Phi(z)) = \phi(z) - z(1-\Phi(z)) \tag{4}$$

In this derivation $\phi(z)$ and $\Phi(z)$ are the PDF and CDF of the normal distribution functions respectively. The re-order point s follows from calculating $s = \mu_{LTD} + z * \sigma_{LTD}$ and this rounded to the nearest integer.

4.4 Poisson distribution

The last replicated model from Porras and Dekker [2008] is based on the Poisson distribution. The only parameter to estimate for this distribution for each item is its average rate of demand over the lead time. This method is only used for demand class 1, which contains items with only 0 and 1 demands. To determine a re-order point s for this method, the same procedure as for Willemain's method in Section 4.1 can be applied. This means that equation (2) must hold, while making use of the formula for $ES(s)$ in equation (1). The probabilities $f(x)$ are then substituted for the Poisson probabilities (with parameter $\lambda_{LTD} = \bar{D} * L$).

4.5 Croston's method

The next method discussed is Croston's method. This method assumes that the demand during the lead time is normally distributed, similar to the most successful method tested by Porras and Dekker [2008]. In this research Croston's method is applied as it is by Willemain et al. [1994]. This method runs a number of iterations that is equal to the number of observed periods in the dataset, for each item. In each iteration it estimates the mean size of the next nonzero demand and the mean interval between nonzero demands. It does only update these estimates when it has seen a nonzero demand occurring. After the method has been run till the last period T, it estimates the mean and standard deviation of the lead time demand distribution. The mean is only depending on the forecasts of time T. With the calculated mean and standard deviation, a normal distribution is estimated. The same analysis as for the normal distribution can be applied for determining the re-order point s and it has to meet the same requirement as equation (3) in Section 4.3. The thorough description of the algorithm of Croston's method is provided in Appendix B.2.

4.6 Truncated normal distribution

The next method discussed has an LTD distribution based on the truncated normal distribution. In the dataset 7080 out of 8610 items do not have any negative demand realization at all, whereas 1530 do have at least one negative demand realization. This means that for most items a distribution could be used that is truncated on the left side on zero. For most items with negative demands this truncation is lower than zero, depending on the size of the most negative lead time demand realization. However, the regular normal distribution allows in many cases for negative values in the left tail, but it is practically impossible to get these values out of the data as lead time demand realizations. This makes it unrealistic to use this non-truncated distribution, especially when this tail is very thick. Additionally, lead time demand realizations higher than the maximum observed lead time demand realization are not present in the dataset as well. This means that the non-truncated normal distribution does also allow for lead time demand realizations in the right tail that are not observed. Therefore, it makes sense to improve the fit of the LTD distribution to the dataset for the normal distribution by also truncating on the right side. Hence, I propose to make use of a normal distribution truncated on both sides. Similar to the normal distribution by Porras and Dekker [2008], a continuous distribution is used to model integer demand values. However, this has not affected the forecasting performance negatively. Therefore, I do not expect this to be a problem for this distribution as well. The new truncated normal distribution will be bounded between its minimum and its maximum total observed lead time demand, resulting in a two-sided truncated normal distribution for modeling the LTD. I choose to use these bounds because nowhere in the dataset there will be lead time demand realizations outside these bounds for each specific item. For many items this will mean that its distribution is truncated between 0 and its maximum observed single demand realization, since 82.2 % of the items do have only positive demands and for 81.5 % of the items it holds that its maximum observed demand realization also equals the maximum total observed lead time demand.

In Figure 1 the PDFs of the lead time demand of item #1669 are visible. In this figure it can be observed for this item that in the new truncated PDF there is no left and right tail that is practically impossible to reach. The figure is truncated on 0 (minimum total observed lead time demand) and on the right side it is truncated on 1 (maximum total observed lead time demand). Additionally, in Figure 2 it can be seen that there is an intersection of both lines somewhere around (0.5; 0.55). This means that for required service levels higher than 0.55, the truncated normal distribution is able to provide a lower re-order point s compared to the non-truncated normal distribution. For required service levels lower than 0.55 the non-truncated normal distribution is able to provide a lower re-order point. When using this example, I assume that the re-order points can be continuous numbers. Nevertheless, when determining a re-order point in the real implementation, I make sure these numbers are integers by rounding them off.

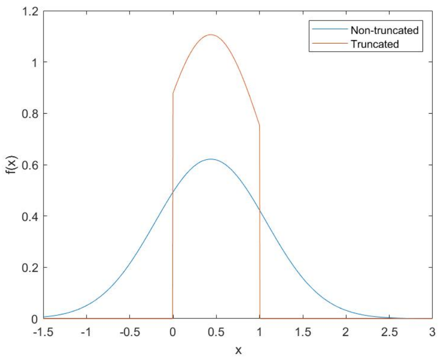

Figure 1: Item #1669: Truncated vs. non-truncated PDF

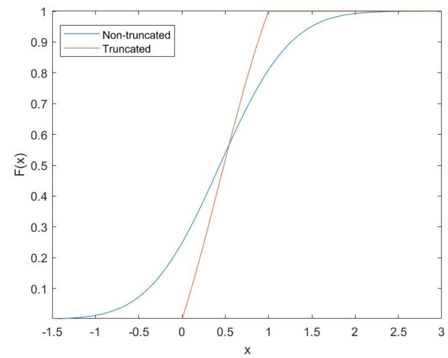

Figure 2: Item #1669: Truncated vs. non-truncated CDF

When it comes to calculating a re-order point s, which is the crucial part for each method, the same procedure is used as when using Willemain's method. This means that I make use of equations (1) and (2) in Section 4.1, while making use of PDF $f(x)$ of the truncated normal distribution. For calculating the mean and the standard deviation, I use the same formulas as for the normal distribution. Hence, the parameters are $\mu_{LTD} = \bar{D} * L$ and $\sigma_{LTD} = SD * \sqrt{L}$. Here L is the lead time in months, $\bar{D}$ is the average demand per month and SD is the standard deviation of the demand.

4.7 Revised Willemain's bootstrap method

The last method that is introduced in this paper deals with improving the bootstrap method that was proposed by Willemain et al. [2004] and it was implemented by Porras and Dekker [2008] as well. This method is going to be revised for demand classes 3. For Porras and Dekker [2008] the bootstrap method resulted in negative savings and in poor service levels (see their research on page 127) for this demand class. Although Porras and Dekker [2008] expected that bootstrap method would not be easily outperformed by all methods, this was not the case. Moreover, demand class 3 contributed for a large part to this outperformance. This demand class contains demands ranging from negative values to over 10,000 per realization. This high volatility is possibly a reason for the bad performance, since the original bootstrap only makes a distinction in nonzero and zero values in the two stage first-order Markov process. This means that there is no distinction made between a demand realization of 1 and a realization of 10,000. Therefore, I propose a 3 state first-order Markov process for the probabilities. I expect this method to work better because it is better able to capture the difference between demands not seriously causing stockout (in the second state) and demands having the serious chance of causing stockout (in the third state). These states will be defined in an exact way later in this section.

The first state is as it was a zero demand, the second state is a nonzero demand (including negative demands), but lower than a certain threshold and the third state is for demands bigger than that threshold. The following matrix will be used to describe the Markov process. 1 means that the process is in the first state, 2 that it is in the second state and 3 means that the process is in the third state.

$$\begin{bmatrix} P(1|1) & P(2|1) & P(3|1) \\ P(1|2) & P(2|2) & P(3|2) \\ P(1|3) & P(2|3) & P(3|3) \end{bmatrix}$$

This new method thus allows for transition probabilities between the state with demands seriously causing stockout and the state with demands not seriously causing stockouts. This new method will largely resemble the ordinary Willemain's bootstrap, because the probability from going from a zero to a nonzero demand will be equal to the probability of going from state 1 (zero demand) to state 2 plus the probability of going from state 1 to state 3. Equivalently, the probability of going from the nonzero demand to a zero demand will be equal to the probability of going from state 3 to state 1 and the probability of going from state 2 to state 1. However, the difference exists in the presence of the transition probabilities in which the process stays in state 2, stays in state 3, moves from state 2 to state 3 or moves from state 3 to state 2. Thus, the original bootstrap could make no distinction on transitions between high and low demands. This new method can do this and therefore I expect it to give a better estimation of the CDF. For instance, there can be items that do not see two subsequent large demands (state 3) but there can be a small (even negative) demand (state 2) after the large demand. In that case the ordinary Willemain's bootstrap does still make it possible to get two subsequent large demands. However, the revised Willemain's bootstrap makes sure that it is impossible to get these two subsequent large demands because it has calculated the transition probability of staying in state 3 equal to 0. In that way this results in a better CDF that shall provide more realistic re-order points for given confidence levels.

It is now to be determined how to define the value that separates the second state from the third state, which is the threshold. When calculating a threshold that divides the nonzero demands, it needs to be made sure that there is at least one nonzero demand above and one nonzero demand below this threshold. There are some items that have nonzero demands being all equal to each other. To be able to get the performance of these items with this method, the performance of the ordinary Willemain's bootstrap method is used for these items. In total there are going to be four different ways of calculating the threshold that separates the second from the third state. When the threshold is calculated, I would like to make sure that demands that are then put in the second category should not affect the inventory system too much. With this I mean that these demands do not stand a great chance to cause a stockout. However, demands in the third category should be the items having a bigger chance to cause a stockout and affect the system negatively. With the created transition probabilities I can then better estimate the probability of two large subsequent demands that cause definitely a stockout or the probability of two subsequent small demands not causing a stockout. With this distinction made possible in this way, I hope to improve the performance of this method compared to the ordinary Willemain's bootstrap method.

The first way of calculating the threshold is making the threshold equal to the mean of the nonzero demand observations. I choose this threshold criterion as a baseline method, to be able to compare it with the other performances. Furthermore, setting the threshold equal to the mean results automatically in the presence of demand observations belonging to both the second and third state.

The second way of calculating the threshold is setting it equal to the average lead time demand. I expect this to work well since items lower than this threshold cannot form a serious risk to the system performance, because the system should already be capable of anticipating on these demands, without causing stockout. However, demands higher than this threshold may cause a stockout since these single demand realizations are already higher than what is expected in one single month.

The third way of calculating the threshold is setting it equal to re-order point s, which is the re-order point of the current policy. I expect this threshold to work well because demands lower than this threshold can in many cases not cause a stockout since the standard level of items is higher than this threshold. Moreover, demands higher than this threshold can cause a serious stockout because the standard level of items available

in the warehouse can be lower than these demands. This is because there is only placed a re-order after the level reaches s.
The fourth way of calculating the threshold is setting it equal to re-order point s plus lot size Q. I expect this to work well because demands higher than this threshold will cause a stockout in any case, since the inventory level is never higher than this threshold. Demands lower than this threshold can cause stockouts in some cases since the inventory level is generally varying between s and $s + Q$. This range can be pretty wide, especially for items in this demand class because Q is positively correlated with demand D, which is visible in equation (5) in Section 4.8. Therefore, there might be too many demands below the threshold. However, still I want to implement this threshold to find out the effects of such a high threshold.
It can happen for an item that one of the categories does not contain any demand. In the case the second category does not contain a demand, the smallest demand is moved to that category and the threshold is set equal to that demand. In the case the third category does not contain a demand, the highest demand is moved to that category and the threshold is set equal to that demand.

4.8 Economic order quantity

In the latter sections models to determine the re-order point s have been described. This means that there can be now calculated for each item a level s and if the inventory falls below this value, then an extra quantity of items equal to Q is ordered. This lot size Q is calculated independent of each method directly from the data. The lot size Q is the economic order quantity (EOQ), that is rounded off afterwards. A traditional definition of EOQ from Axsäter [2000] is used, which is the following formula:

$$EOQ = \sqrt{2 * \frac{DS}{H}} \tag{5}$$

In this formula D is the demand on annual basis (as derived from the dataset), S is the order costs per order (independent from quantity ordered) and H is the holding costs per unit on annual basis. The costs have been described in Section 3.2. After the EOQ has been calculated it is rounded off, also according to Axsäter [2000]. First I calculate $m = \lfloor EOQ \rfloor$, which is equivalent with rounding off below to the closest integer. Then this is rounded off as follows:

$$Q = \begin{cases} 1 \text{ if } m = 0 \\ m \text{ if } m \neq 0 \text{ and } \frac{EOQ}{m} \leq \frac{m+1}{EOQ} \\ m + 1 \text{ otherwise} \end{cases}$$

Imagine the case when there is an order that makes the inventory level decrease far below s, such that ordering Q does not result in an inventory level higher than s. In that case an amount nQ is ordered such that the inventory level exceeds s again, where n is the smallest integer for which this holds. Q is set to 1 by default for demand classes 1.

4.9 Optimization process

Now all necessary topics have been defined, the optimization process is explained in this section. The goal of the optimization process is minimizing the costs of the system on class level. When implementing the process, there are two conditions that need to be met for each class. The first one is that the fill rate must be equal to the fill rate of the current system or at much 1 % below this original fill rate. The second one is that

the CSL must be higher than the current CSL minus 5 % for each class. This CSL condition does not need to be met when the savings are negative. Porras and Dekker [2008] also describe this CSL requirement, but they also say that in some cases this specification was not met, without a further specification on when this could happen. In this research this requirement is implemented and is treated the same as the requirement for the fill rate. The fill rate condition, however, has to be met in all cases, except when a required fill rate of 100 % is not resulting in satisfying this condition. This also what Porras and Dekker [2008] do.
Since I use the same notation and same optimization process as Porras and Dekker [2008] do, the description of this can be found in Appendix B.3. The optimization process including all methods is implemented in Matlab R2019b on a Windows laptop with an Intel(R) Core(TM) i7-1065G7 with also 8GB of RAM.

5 Results

Now all necessary methods have been defined, all these methods are implemented and the results are provided in this section. In Section 5.1 the output of the replication part is provided. In Section 5.2 the output of all newly implemented methods are provided. For each of the seven methods I provide insights in the results on item level and on class level combined with some charts representing trade-offs between costs, target fill rate and realized fill rate. Additionally, for each of the seven methods the results of the overall optimization are visible for each class.

5.1 Issues emerged in the replication process

First, I have replicated the results of Porras and Dekker [2008] for classes 135 and 233, which will be discussed in this first subsection. Second, I will discuss the results of item #1835, item #7559 and some items from class 212. Third, the overall optimization results will be discussed.
Generally, the results of Porras and Dekker [2008] have been replicated closely. However, some issues with the dataset and some uncertainties about the exact implementation of the used methods in the paper caused some deviations from the original results. These problems will be further explained in the remaining part of this section.
First of all, I had an issue with the grouping for the price classes and the number of observations in the dataset. I started with a difference of observations of 140 and after cleaning this difference was 116. More importantly, as can be seen in Section 3.1.3, 5 % of my observations were part of price class 5 and 22.6 % of my observations were part of price class 2. Porras and Dekker [2008], however, calculate these percentages on 8 % and 19.6 % respectively. An important consequence of this is the wrong calculation of the EOQ, which is inversely correlated to the price. This happened to items with a very low price (price class 2) in my dataset, of which I had significantly more than Porras and Dekker [2008] had. A very high value for the EOQ means a very high starting stock, which caused some items even not reach the re-order point s during the optimization period. This happened for all the items that are present in Appendix D.1. Furthermore, the fact that I had more items with very low prices resulted in significantly lower total costs in my research.
Another issue with the dataset, of which I cannot say to what extent it forms a problem, is the difference in demand I observed for one item in my dataset compared to the dataset of Porras and Dekker [2008]. This was the case for item #1835, which I will more extensively explain in Section 5.1.2. This difference was only observed for this item, but I cannot say to how many items happened the same.
The second issue I want to discuss is that Porras and Dekker [2008] report in their paper that, when optimizing the system, they make sure that the fill rate is higher (or within 1 % precision) of the current fill rate.

Furthermore, they say that the CSL must be within a 5 % range of the current CSL, but this requirement had not longer to be met when there were negative savings. Additionally, they say, this was hard to meet in some cases but there is left out a further numerical specification on when this had not to be the case. I have made sure that my fill rate values were meeting these requirements, unless imposing a 100 % requirement would still not meet these requirements. The condition on CSL had in my research to be met when there were positive savings. Although my fill rates do match for most classes pretty closely to the original ones (especially for classes not in price class 2 and 5), my values for the CSL do not match as well as the fill rates do (see Appendix E). This mismatch is largely caused by the volatile values for demand class 3, for which both my results and the ones from Porras and Dekker [2008], have very low and highly varying CSL values. Nevertheless, generally my values CSL for this class are significantly higher. Furthermore, for the normal distribution it is impossible to impose a 100 % required fill rate, since the formula for UNIL(z) will never reach 0 (see equation (3) in Section 4.3). Therefore, this observation is left out. When a 100 % required fill rate was needed, the corresponding value for a 99 % fill rate was reported.
The third issue discussed deals with the costs, which were not always within a reasonable range of the costs calculated by Porras and Dekker [2008]. This is for the biggest part caused by the aforementioned points, which caused small deviations from the calculated fill rates and CSL values. Next, it can be very hard, in general, to calculate the exact same values for these service levels. Small deviations of these values, however, can cause larger deviations in calculated costs, since the costs often shift with these values. This feature can for example be observed for the figures that are referred to for class 135 in Section 5.1.1. The reason for the big mismatch in costs will also be further explained in Section 5.1.3.

5.1.1 Replication of specific classes

In this section the output of classes 135 and 233 are visualized for the methods used by Porras and Dekker [2008]. The detailed results are given in Appendix C. Looking at the output of class 135 in Figures 5 and 6, it can be seen that the realized fill rate is logically increasing with the target fill rate and there is a stair-shaped function visible for this class. The costs are logically increasing in the realized fill rate as well. Willemain's method and the normal based model seem to provide the best results for this class when looking at the output. However, this is a class in the price class 5, with which there was an issue in the dataset. I only had 31 items in this class, whereas the original paper had 71 items, which is the reason for the different shape of the functions compared to the original ones. This is clearly visible in the stair-shaped first function, as a result of less observations in this class. This different shape does also result in a different conclusion in terms of comparison to the current policy. In my research all methods do outperform the current policy slightly. However, Porras and Dekker [2008] can argue, based on their figures, that all methods clearly outperform the current policy in terms of savings. This conclusion cannot be drawn from my figures. Therefore, this class is a perfect example of the differences between my results and the results of Porras and Dekker [2008] with different conclusions for this class and a clearly different shape of the charts.
The second class I want to discuss is class 233 whose output is visible in Figures 7 and 8 in Appendix C.2. The results of this class almost match perfectly with theirs. Just as in the original paper, the empirical model seems to result in the smallest costs and the normal based model has the best fill rate. Moreover, all three methods are very close for this class.

5.1.2 Replication of single items

In this section the replication of some single items is discussed. I start with some items in class 212, visible in Table 7 and Table 8 in Appendix D.1. The items shown are in the paper of Porras and Dekker [2008] part of class 215, but in my paper they are part of class 212, which is caused by their price of 0.01, resulting in a very high lot size Q. The first table shows the performance of the current system, which is exactly matching with the results of Porras and Dekker [2008] (page 119). However, for all the three replicated methods for this class there were no inventory cycles and demand was fully out of own inventory. This is caused by the very high lot size Q. Since the start level is equal to s plus Q, there were no inventory cycles necessary and all items were served out of own inventory. Therefore, my items do not match with the originals at all. Hence, the output of this class is a perfect example of what happens due to the dissimilarities in the dataset. This problem does, however, not occur for the current system because then the lot size Q is set equal to the max value minus the min value in the dataset, which is independent of the price.
In Appendix D.2.1 and D.2.2 the output of items #1835 and #7559 is visible. The results do pretty match the results of Porras and Dekker [2008] (page 123). However, here my functions for item #1835 are smoother. This is caused by a dissimilarity in the dataset. For item #1835 I observe in the dataset an extra demand equal to 10 in period 10, which is shown in Table 9. Porras and Dekker [2008] do not report this demand in their table on page 123. It is hard to say to what extent the dataset of Porras and Dekker [2008] differs from my dataset and thus what the consequences are for the comparability of the results. Furthermore, I assumed a lead time of 2 months since in the dataset a lead time of 31 days was reported, which is in contrast to Porras and Dekker [2008] that assumed a lead time of 1 month for this item. As has been mentioned before, all lead times were rounded up(wards) to the closest integer in months in my paper, while Porras and Dekker [2008] do not report a rounding procedure for this. For item #7559 the results are almost exactly the same. However, it strongly seems to me that Porras and Dekker [2008] have Willemain's method given the name of the normal based model and the other way around for this item (see page 123). This is because I get the results for the normal based model as they are for Willemain's method and the other way around for item #7559. Furthermore, on the left side of the page they use the order N-E-W, whereas on the right side W-E-N (N and W changed).

5.1.3 Replication of overall optimization

In Appendix E the optimization is visible per class and joint classes for the replicated methods. When looking at the totals for the CSL and the fill rate below in Tables 10 and 11, these have been replicated quite sharply. However, I obtained higher values for the average CSL. This deviation in average CSL is almost entirely caused by the deviations that are present in demand class 3 (total demands higher than 60). In this joint class Porras and Dekker [2008] had an average CSL of 0.37 for Willemain's method and the empirical method, whereas my values are around 0.75 for these methods. However, for classes I observed low CSL values, Porras and Dekker [2008] did as well. For class 362, a class in misspecified price class 2, Porras and Dekker [2008] have a very low value for its CSL which I do not have. This class is very much impacting the average with its 287/631 items. The same holds for class 332 with 145/631 items.
Additionally, my total costs and savings do not seem to match as well. This is due to the dissimilarities in price class 5. For example, for the current policy class 135 already contributes to a cost of around 1,800,000 in the paper of Porras and Dekker [2008] whereas I have a cost of 138,242 in this class. In the paper of Porras and Dekker [2008] this contributes to a saving of roughly 600,000 for each method for class 135. This problem does also apply to other classes like: 115, 235, 255, 265 and 455. This affects the total costs, although looking

on class level of correctly specified classes, it can be seen that most costs are very close to the ones of Porras and Dekker [2008].

5.2 General results

After implementing all the methodology, I found that the truncated normal based model had the best performance of all models. These optimization results can per class and joint classes be found in Appendix E. The second best model was the normal based model, as it has been implemented by Porras and Dekker [2008]. In spite of the issues with the dataset, still the same conclusion can be drawn that the normal based model is the best out of those four models. Croston's method, although also relying on the normal distribution, did in terms of savings not provide the best results. It performed worse than both the empirical model and the normal based model, although it did outperform the Poisson model for demand class 1. Additionally, it did outperform the best out of four implementations of the revised Willemain's method. The normality assumption for the demand needed for Croston's method did not hold for this dataset, although this assumption does not need to hold for satisfactory results when using this distribution. This was shown by Porras and Dekker [2008]. The results of the four implementations of this revised Willemain's method have not been satisfactory. In this section I will give some more extensive results about the revised Willemain's method in Section 5.2.1 and I will show some optimization results on item level for the newly implemented methods in Section 5.2.2. Additionally, I will compare the best performing models, which are the truncated normal based model and the normal based model, with each other by an optimization on item level in Section 5.2.3.

5.2.1 Results revised Willemain's method

After implementing all different specifications of the revised Willemain's method, the optimization results of this method are to be found in Appendix E.4. From these results it can be concluded that this method does generally not improve the original method of Willemain et al. [2004] by allowing for an extra category in the model, based on these calculated bounds. Allowing for an extra state in the Markov process gives different state-transition probabilities that have not contributed to a better estimation of the LTD distribution. All four implementations resulted in very similar service levels (CSL somewhat below 0.70 and the fill rate around 0.80). Additionally, all methods except for the one having a bound solely determined by its original re-order point, did result in significantly lower savings than the original bootstrap procedure. However, defining the bound by the original re-order point s, makes sense since it is the only method that is providing similar performances to the original Willemain's method. Apparently, defining the bound in this way can provide the most realistic distribution and is able to realize the best performance in terms of forecasting. Since also all original three methods by Porras and Dekker [2008] have similar service levels for the third demand class (see Appendix E.5 Table 13 and 14), the savings for the implementation of Willemain's method are shown in Table 4. In this table it can be clearly seen that the performance of the implementation based on re-order point s is comparable to the original methods and it is clearly outperforming the normal based model for this class. However, the other implementations of the revised Willemain's method are strongly outperformed. From the analysis of this method it can be concluded that determining the bound that separates the second state from the third state is a very determinative factor for the performance.

Table 4: Savings of different models compared to the current system

.

	C	L	M	S	QS	N	E	B
class	Costs	Savings	Savings	Savings	Savings	Savings	Savings	Savings
312	9252	829	850	1038	790	1353	-313	-189
313	81006	4513	3661	6464	3357	-2043	-4503	-7489
322	14976	-2359	-2978	-2736	-2176	4959	4294	4344
323	8385	-66	-34	-517	-195	-370	-505	-290
332	80204	24267	26566	27463	24466	30080	18806	18236
333	94218	-5628	-5971	-2041	-5715	-2123	-11772	-11991
334	44279	-717	-294	-55	-72	-59	-5	-170
342	16113	1379	1536	1601	1624	6773	4347	4053
343	102969	-3852	-1548	-676	-877	-10863	-2329	3258
352	1942	-117	-4	-22	-22	-77	-46	-19
362	131782	12180	13184	26863	12927	-849	52867	47242
363	76429	-10480	-10623	-1796	-9689,9	17514	-1043	-2509
Total	661556	18062	22460	55588	22532	44298	59798	54476

The table contains savings of the revised revised bootstrap methods combined with the methods used by Porras and Dekker [2008]. C is the current system, L is the revised bootstrap with bound determined by average lead time demand, M is the one with bound determined by mean, S is the one with bound determined by re-order point and QS is the one with bound determine by re-order point plus EOQ. N is the normal based model, E is the empirical model and B is Willemain's method.

5.2.2 Optimization on item level

To provide some extra insights in the optimization of the newly implemented methods in this research, I show some figures displaying the fill rate and the costs for each method for item #1835 in class 334. Table 9 in Appendix D.2 contains the demands of this item. The revised Willemain's method in this case has a bound which is depending solely on the original re-order level s. The figures of the three methods are visible in Appendix D.2.3. Looking at the charts, it can be said that the truncated normal based model is resulting in the lowest costs, while Croston's method and the revised Willemain's method are able to achieve the highest fill rates. The fact that these two methods can achieve higher fill rates than the truncated normal based model is due to the fact that the underlying distributions have much thicker right tails than the truncated normal distribution. For instance, Croston's method estimates a mean equal to 4.4 with a standard deviation equal to 26.3 whereas the truncated normal based model estimates a mean of 4.2 with a standard deviation equal to 7.3. This results in a much thicker tail for Croston's method and it is therefore able to provide higher re-order levels for given required fill rates. Also, the revised Willemain's method has a somewhat thicker right tail for which thus the same property holds. This described feature is visualized in Figure 3 that does show the CDFs of each of the three methods. The shapes of the functions are stair-wise because for the revised Willemain's method the point of the right side truncation for item #1835 is at 76 meaning that there is barely cut off any mass on the right side. This feature does also hold for many other items and will be described more extensively in Section 5.2.3.

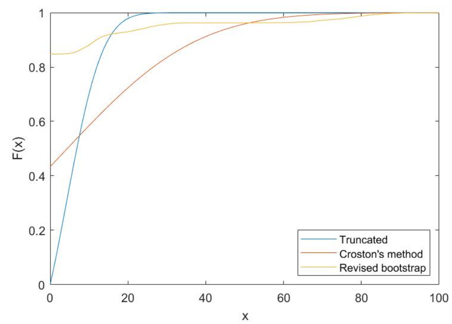

Figure 3: Cumulative distribution functions using the truncated normal based model, Croston's method and revised Willemain's bootstrap

5.2.3 Comparing the normal distributions

To easily compare the best model of Porras and Dekker [2008] (the normal based model) and the best model of my methods (the truncated normal based model), the joint performances of these models are visible in Table 5. Here it is visible that the service levels of both models are very much the same whereas the truncated normal based model is outperforming the normal based model based on costs. The truncated normal based model is actually outperforming the normal based model mainly in demand classes 3 (total demands higher than 60) based on costs.

Table 5: Joint results of the current system (C), the truncated normal based model (N) and the normal based model (TN).

		C			N			TN		
Dem	# items	Costs	tCSL	t fill rate	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
1	2244	1790450	0,960	0,956	252422	0,955	0,949	247920	0,955	0,950
2	5220	4764710	0,852	0,855	516471	0,814	0,901	530880	0,812	0,908
3	631	661556	0,744	0,832	44298	0,602	0,862	131072	0,654	0,817
4	502	1392239	0,961	0,968	-15587	0,963	0,971	-21743	0,962	0,970
Critic										
1	527	924251	0,887	0,876	-43697	0,855	0,901	-32956	0,855	0,902
2	454	298101	0,941	0,949	61206	0,907	0,955	61086	0,903	0,958
3	3220	3383313	0,904	0,894	267185	0,865	0,912	258540	0,873	0,920
4	1410	898211	0,933	0,925	117795	0,898	0,936	131191	0,904	0,942
5	389	556660	0,906	0,895	118616	0,878	0,921	116708	0,876	0,924
6	2610	2548420	0,801	0,844	276498	0,771	0,902	353559	0,765	0,890
Price										
2	2441	551561	0,833	0,891	188806	0,784	0,960	266169	0,789	0,948
3	3299	1576656	0,883	0,877	225454	0,849	0,903	234559	0,853	0,912
4	2503	3575596	0,911	0,893	353428	0,892	0,888	378568	0,888	0,891
5	367	2905142	0,910	0,890	29916	0,872	0,899	8833	0,878	0,900
Total	8610	8608955	0,878	0,886	797604	0,844	0,915	888129	0,846	0,916

I have also tried to give an answer to the different performances of the truncated normal based model in the demand classes by looking at the masses that are cut off from the distribution. The masses that are cut off per demand class are visible in Table 6. For all demand classes it can be seen that there is barely cut off any mass on the right side of the distribution. However, for all demand classes except for class 4 there is cut off more than 30 % of the total mass on the left side. The fact that there is also almost no mass cut away on the left side in demand class 4 is probably due to the fact that the point of truncation on the left is always equal or smaller to -1, combined with a mean close to zero and a low standard deviation. Due to the relatively high average standard deviation of 30 in demand class 3 one could expect that the tails cut off on the right would be thick as well. However, relatively low means compared to the standard deviations make the mass of the cut-off right part very thin as well. In the fourth row of Table 6 it is visible that for demand classes 1, 2 and 3 the large majority of truncation points on the right side is bigger than the mean plus 2 times the standard deviation. This means that in most cases the mass cut away is less than 2 %. Based on the observation that all demand classes have almost no mass cut away on the right side and all (except for class 4) have roughly the same mass cut away on the left side, I cannot tell why there are lower costs in demand class 2 and 3 in the truncated normal based model compared to the non-truncated model. However, from this analysis it can be concluded that the truncation on the right side has been too little and these points should have been shifted more towards the left. Thus, the improvement by the truncation is almost entirely caused by truncation on the left side.

Table 6: Masses under the normal distribution that have been cut off in the truncated normal distribution.

Demand class	1	2	3	4
Average mass cut-off on the left	0.358	0.301	0.340	0.015
Average mass cut-off on the right	0.010	0.006	0.0025	0.028
% Right bounds > $\mu + 2\sigma$	86.1	92.2	98.8	61.4

To visualize the lower costs that are present in the truncated normal based model compared to the original normal based model, Figures 9 and 10 are provided in Appendix C.3. These figures apply to class 343, for which the truncated normal based model had higher savings than the normal based model. The figures show very similar service levels for both methods. However, the truncated normal based model does have lower costs for same realized fill rates than the normal based model. Moreover, the truncated normal based model does also have slightly higher realized fill rates for the same target fill rates for this class.

6 Conclusion

The goal of this research was to compare the performances of several methods for predicting spare parts demand at an oil refinery. I started with a thorough cleaning of the data necessary to make it suitable for the research. Four of the used methods had already been implemented by Porras and Dekker [2008] and these have been replicated. Although there have been dissimilarities in the dataset compared to the research of Porras and Dekker [2008], causing some deviations in the results, the same conclusion can be drawn from the results of these methods. After these four methods had been implemented, the results of the other three methods have been calculated.

The purpose of this paper was to find out what the best model was out of several models to predict spare parts demand at an oil refinery. To be able to draw a conclusion about the best model, I divided the research into three different research questions.

My first research question was what the forecasting performances are of the normal based model, the bootstrap model, the empirical model, the Poisson model and Croston's method. After seeing all the performances of these methods, it can clearly be said that the normal based model is the best one, out of these five models.

Although Croston's does also rely on the normal distribution, it has not provided any improvement compared to the normal based model. The empirical method did also outperform Croston's method slightly. However, Croston's did outperform the Poisson model and Willemain's bootstrap method.

Having concluded this, my second research question was whether there could be improvements made on a normal based model for modeling lead time spare parts demand in an oil refinery by making use of a truncated normal distribution. The very short answer to the question would be 'yes.' There can indeed be made improvements on a normal based model for modeling the lead time spare parts demand in oil refinery by making use of a truncated normal distribution. This improvement is reached when the distribution behind the normal based model is truncated on two sides. I have shown that this results in a better performance than the original normal based model. However, from the results it also becomes clear that a truncation on the right side is not impacting the performance a lot.

The last research question was whether there could be improvements made on Willemain's bootstrap model for modeling lead time spare parts demand in an oil refinery for spare parts with high demands. The very short answer to this would be, in my opinion, 'maybe.' Three of the four implementations of the presented revised Willemain's method, have not given satisfactory results at all and only one of the implementations resulted in very similar results to the original implementation of this method. Therefore, I think that this extensified method has no added value with respect to the original implementation and has no added value with respect to the rest of the models as well. This statement can be motivated by the fact that the revised and the original Willemain's method are clearly outperformed by the other models in this research.

Having answered all these three research questions, it can be said that the best model of this research using an ex-post approach is the truncated normal based model, narrowly followed by the normal based model.

After having drawn the conclusion of this research, I would like to provide the limitations of this research and some suggestions for future research. First of all, due to the small amount of observations available, the methods have only been implemented using an ex-post approach. The four replicated methods of Porras and Dekker [2008] have in their research also been applied with an ex-ante approach, which not led to a different best model. Using an ex-post approach, my conclusion is that the truncated normal based model is outperforming the non-truncated one. However, it would be very valuable to see what happens to this model when it is applied to data that has not been used for fitting purposes. For example, if the minimum value of the lead time demand realizations for an item is lower in the data used for testing, what will then happen to the performance for that item? The model that uses a truncated distribution does in fact assume that values outside the truncated bounds cannot appear. In that sense using a truncated distribution is not very robust. Therefore, applying this method to a dataset that has not been used for fitting may give more clarity about the validity of this distribution for modeling spare parts demand. However, when based on expert judgement it can be said that there will not be any returns (negative demands) or returns will not exceed some number, the aforementioned assumption holds and the truncated distribution works well. This problem is not playing a big role when the distribution is truncated on the right side, since I have concluded that the truncation I used is not erasing much mass away on the right side. Therefore, I also suggest for future research to introduce a way of calculating the right truncation point that does shift it more leftwards.

The second thing I would like mention is that it may be good to apply the methods to a larger dataset. Now the dataset only contains 55 periods, which is meaning there is no room to model a possible non-stationarity. This non-stationarity could be caused by a changing lead time demand over time, which is not modelled in this research. Taking into account this possible feature could contribute to the long-term validity of the used models.

References

S. Axsäter. *Inventory control*. Kluwer Academic Publisher, Boston, 1 edition, 2000.

J. Do Croston. Forecasting and stock control for intermittent demands. *Journal of the Operational Research Society*, 23(3):289–303, 1972.

George J. Feeney and Craig C. Sherbrooke. The (S- 1, S) inventory policy under compound poisson demand. *Management Science*, 12(5):391–411, 1966.

Adel A. Ghobbar and Chris H. Friend. Evaluation of forecasting methods for intermittent parts demand in the field of aviation: a predictive model. *Computers & Operations Research*, 30(14):2097–2114, 2003.

Eric Porras and Rommert Dekker. An inventory control system for spare parts at a refinery: An empirical comparison of different re-order point methods. *European Journal of Operational Research*, 184(1):101–132, 2008.

Edward Allen Silver, David F. Pyke, Rein Peterson, et al. *Inventory management and production planning and scheduling*, volume 3. Wiley New York, 1998.

Richard B. Watson. The effects of demand-forecast fluctuations on customer service and inventory cost when demand is lumpy. *Journal of the Operational Research Society*, 38(1):75–82, 1987.

Thomas R. Willemain, Charles N. Smart, Joseph H. Shockor, and Philip A. DeSautels. Forecasting intermittent demand in manufacturing: a comparative evaluation of croston's method. *International Journal of Forecasting*, 10(4):529–538, 1994.

Thomas R. Willemain, Charles N. Smart, and Henry F. Schwarz. A new approach to forecasting intermittent demand for service parts inventories. *International Journal of Forecasting*, 20(3):375–387, 2004.

Sha Zhu, Rommert Dekker, Willem Van Jaarsveld, Rex Wang Renjie, and Alex J. Koning. An improved method for forecasting spare parts demand using extreme value theory. *European Journal of Operational Research*, 261(1):169–181, 2017.

A Data cleaning process

Certain items and observations were removed from the dataset before starting implementing all the methods in the research. First of all, the month November 1997 was deleted, since this month contained strange observations and the real consumption of this month was not known. Next, items with a total demand equal to a negative number or items with no demand or only one realization during the whole period were filtered out. Additionally, I observed some errors in the criticality specifications of certain items. In the dataset there exists a value that indicated the total number of times an item was related to a piece of equipment with a certain criticality. This number should be equal to the summation of the values indicating the number of times each item is related to each of the criticality levels. When this was not the case, this item was removed. Porras and Dekker [2008] report that there exists some error in the criticality class, but do not specify what this error actually was. Therefore, I assumed it was this error. Also, items that were part of a group that contained less than 6 observation were deleted as well. Another deletion of observations deals with the removal of large demands for demand classes 2 and 3. Months that contained observations bigger than the mean of the positive demands plus two times the standard deviations were removed. After implementing all the aforementioned cleaning steps, I am left over 8,610 spare parts.

B Algorithms

B.1 Algorithm Willemain's bootstrap

In this section a more thorough description of the implementation of Willemain's bootstrap algorithm is provided. This method estimates the lead time demand distribution by making point forecasts for each item during its lead time. To do so, it uses a two state, first order Markov process. It first determines for each demand forecast during the lead time whether it is going to be zero or nonzero. These forecasts are depending on whether the last demand was zero or nonzero. After the forecasts have been provided, the method implements a jittering procedure for forecasting the values of the nonzero demands. This jittering procedure can cause a deviation in the point forecasts, which improves accuracy according to Willemain et al. [2004].

The following stochastic matrix shows the transition probabilities in the two state, first order Markov process. The probabilities of this matrix are just calculated from the sample of the observations. N refers to an observation being nonzero in this matrix.

$$\begin{bmatrix} P(0|0) & P(N|0) \\ P(0|N) & P(N|N) \end{bmatrix}$$

Thus, with these calculated probabilities I determine for each month during the lead time whether this demand is equal to zero or nonzero. After this has been done, for each nonzero demand there is randomly selected a demand from the set of nonzero demands available in the dataset. These demands are independently selected and replacement is possible, meaning that the same nonzero demand value can be selected more than once. Furthermore, if a certain nonzero demand value occurs multiple times in the dataset, it stands a greater chance to get selected, proportional to the number of times it occurred.

After the values of nonzero demands have been determined, these are jittered. This jittering procedure allows for some extra variation in the point forecasts. Willemain et al. [2004] argue that their form of jittering generally improves accuracy, especially for items with short lead times. Let X^* be one of the historical nonzero demands selected and let Z be a standard normal random deviate, randomly sampled from the standard normal distribution in each iteration. The jittering process is the following:

$$\text{Jittered} = 1 + \text{INT}(X^* + Z\sqrt{X^*})$$
$$\text{If Jittered} < 0, \text{ then Jittered} = X^*$$

After all nonzero point forecasts have been jittered, these are summed up to get an estimation of the demand during the lead time. Porras and Dekker [2008] repeat this described process 1000 times for each item to estimate the lead time demand distribution. This number of steps is also used in my research. In the end there is calculated a CDF $F(x)$, with $F(x)$ equalling the probability that the lead time demand is smaller than x.

To determine a re-order point s for this research, from the calculated CDF $F(x)$, the expected units short $ES(s)$ need to be calculated iteratively for different values of s. The formula for $ES(s)$ is as follows:

$$ES(s) = \sum_{x|x>s} (x-s)f(x) \tag{6}$$

Then this calculated $ES(s)$ must be satisfying the following condition, to obtain a required fill rate of β:

$$\beta \leq (1 - \frac{ES(s)}{Q}) \tag{7}$$

It holds that $1 - \frac{ES(s)}{Q}$ is equal to the calculated fill rate for a given s. Furthermore, it can be observed that $ES(s)$ is decreasing in s. Therefore, it holds that at some point equation (7) is going to be met. In Figure 4 the CDF of item #741 can be seen. This CDF is obtained by applying the procedure of Willemain's bootstrap. Small deviations compared to Porras and Dekker [2008] can be present for this item in this graph, because there is made use of a random generation of values and the starting state of the Markov process can be different. I have made sure that the starting state was always a zero demand and have thus, not used a random variable to determine the starting state. Porras and Dekker [2008] do not specify in which state they let the process start.

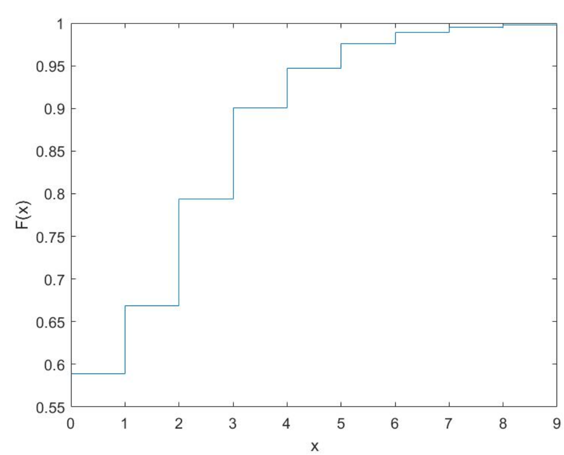

Figure 4: Cumulative distribution function using Willemain's bootstrap

B.2 Algorithm Croston's method

In this section a more thorough description of Croston's method is provided. The method is implemented as it has been by Willemain et al. [1994]. In the algorithm presented next, $X(t)$ is the demand at time t, q is the time interval since last demand, $S(t)$ is the forecast for the mean size of the next nonzero demand after time $t-1$, $I(t)$ is the forecast of the mean time between nonzero demands after time $t-1$, q is the time interval since last demand and α is a smoothing constant. T is the number of months in the dataset. The algorithm for Croston's method is as follows:

Algorithm 1: Croston's method

Result: Forecasts for the mean size of the nonzero demands and the mean time between demand realizations.

Initialization: Initialize by calculating the average time between two demand realizations ($= I(0)$) and the average demand when demand is nonzero ($= S(0)$), initialize t by setting $t = 1$, initialize q by rounding $I(0)$ off to closest integer;

```
while t < T do
    if X(t − 1) = 0 then
        S(t) = S(t − 1)
        I(t) = I(t − 1)
        q = q + 1
    else
        S(t) = αX(t − 1) + (1 − α)S(t − 1)
        I(t) = αq + (1 − α)I(t − 1);
        q = 1
    end
    t = t + 1;
end
```

In this algorithm every time there is a forecast provided for the size of a nonzero demand $S(t)$ and the mean period between 2 nonzero demands $I(t)$, both forecasted at time $t-1$ for the period from t on. Hence, the mean demand per period $M(t)$ is equal to $\frac{S(t)}{I(t)}$. In every iteration of the algorithm, these are updated depending on whether there was a demand realization last period. If there was no demand realization in last period, these forecasts are not updated and q is incremented by one. If there was a realization, the forecasts are updated and q is set to 1. The smoothing constant α determines for forecast $S(t)$ to what extent it depends on $X(t-1)$ and to what extent on $S(t-1)$. For $I(t)$ it determines to what extent it depends on q and to what extent on $I(t-1)$. The mean can in the end be calculated as $L * M(T)$ and the variance as $L * V$. Here L is the lead time, and V is as follows:

$$V = \frac{1}{T}\sum_{t=1}^{T}(X(t) - M(t-1))^2 \tag{8}$$

The smoothing constant $0 < \alpha < 1$ is yet still to be determined. In the original paper by Croston [1972] about Croston's method, it is mentioned that values between 0.10 and 0.20 have empirically shown to result in satisfactory results. This was the case in intermittent demand datasets. Therefore, in this paper I use a value of 0.15 for α. A low value for α results in the forecasts being more dependent on the last forecast compared to the last demand. This lower value seems more appropriate since higher values of α could also

cause a shift of forecasts by one strongly deviating demand.
After the algorithm has been implemented, the same analysis as for the normal distribution can be applied for determining the re-order point s and it has to meet the requirement as equation (3) in Section 4.3.

B.3 Optimization process

In this section a thorough description of the system optimization process is provided. I start with a description of all notation, which is then followed by the steps taken in the optimization process.

- j is the index for an item j that is part of class p
- X is used for the identification of the model in the research, X = current policy (C), Poisson (P), Normal (N), Empirical (E), Willemain (W), Croston (Cr), Truncated Normal (TN) or Revised Willemain (RW).
- $C_j^{(X)}$ is the number of inventory cycles that has been completed by item j using model X. An inventory cycle is defined as an ordering of items.
- $soc_j^{(X)}$ is the the number of stockout inventory cycles of item j, while using model X.
- $D_j^{(t)}$ is the total demand of item j at time t.
- $S_j^{(X)}$ is the total number of items of the item j that had been delivered from stock available in the warehouse with model X.
- $ES_j(s_j)$ is the expected units short for item j for a given re-order point s_j, according to the LTD distribution.

The last step before the optimization process can be started is calculating the service levels of the current system for each class p ($X = C$). These service levels are notated by $CSL_p^{(X)}$ and $\beta_p^{(X)}$, and the following formulas are used for these:

- $CSL_p^{(X)} = 1 - \frac{\sum_{j=1}^{N_p} soc_j^{(X)}}{\sum_{j=1}^{N_p} c_j^{(X)}}$
- $\beta_p^{(X)} = \frac{\sum_{j=1}^{N_p} S_j^{(X)}}{\sum_{j=1}^{N_p} Dj^{(t)}}$

After this has been done, lot size Q_j for each item needs to be calculated. The lot size is calculated according to the procedure described in Section 4.8. After this has been done the optimization process can start. The following steps need to be taken for each item j and each model X.

1. The first step is calculating a re-order point s_j. The LTD distribution combined with $\beta_p = \beta_p^{(C)}$ is used for calculating s_j. The smallest s_j satisfying the following equation is used:

$$\beta_p \leq 1 - \frac{ES_j(s_j)}{Q_j} \tag{9}$$

   The starting stocks when starting the optimization process are as follows: for each item j the starting stock is equal to its re-order point s_j plus its lot size Q_j. This does not hold when we calculate the performances of the current system, then the starting stock is equal to the point to which the item is refilled again when there is an order placed. This is the max value in the dataset.

2. In this step for each item the optimization process is run with the parameters that have been selected in the first step. For each class p, the realized fill rate $r\beta_p$, the realized cycle service level $rCSL_p$ and its total costs TC_p are calculated. For these, the following formulas are used:
   - $r\beta_p^{(X)} = \frac{\sum_{j=1}^{N_p} S_j^{(X)}}{\sum_{j=1}^{N_p} D_j^{(t)}}$
   - $TC_{p,target}^{(X)} = \sum_{j=1}^{N_p} TC_{j,target}$
   - $rCSL_p == 1 - \frac{\sum_{j=1}^{N_p} soc_j^{(X)}}{\sum_{j=1}^{N_p} c_j^{(X)}}$

   The costs are holding costs and ordering costs, as described in Section 3.2. The target in the formula of the costs refers to the target of the fill rate which has been defined β_p for class p. With the calculated costs the savings are calculated for each class p. This is done as follows:

$$Totalsavings_p^{(X)} = TC_{p,target}^{(C)} - TC_{p,target}^{(X)} \tag{10}$$

3. For each class it is checked whether the requirements are met regarding the fill rate and the CSL. If the savings of such an item are positive as well, the calculations regarding that class are finished. If this is not the case, the class enters a new procedure as described in the last step.

4. In the last step an optimization approach is implemented for classes that do not meet the requirements or have negative savings. It holds that for these classes the steps are performed again from step 1 on. This means that for each of these classes a fixed target value for the fill rate β_p is set, beginning at 0.01 and ending at 1, in steps of 0.01. Then for each of these values the fill rate, CSL and savings are calculated. Then, the fill rate and the CSL meeting the requirements with the highest savings are picked.

C Output for several classes

C.1 Class 135

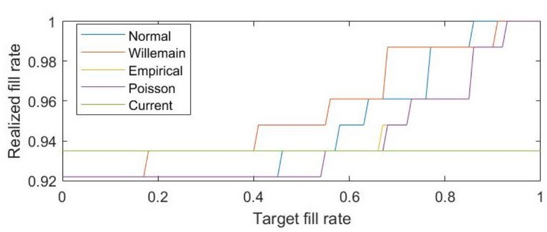

Figure 5: Class 135: Target fill rate vs. realized fill rate

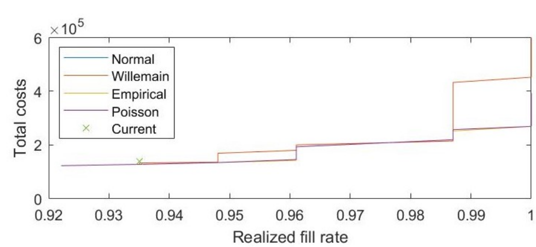

Figure 6: Class 135: Realized fill rate vs. costs

C.2 Class 233

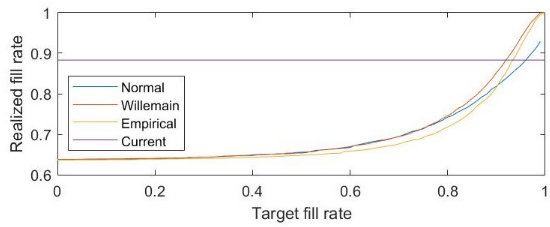

Figure 7: Class 233: Target fill rate vs. realized fill rate

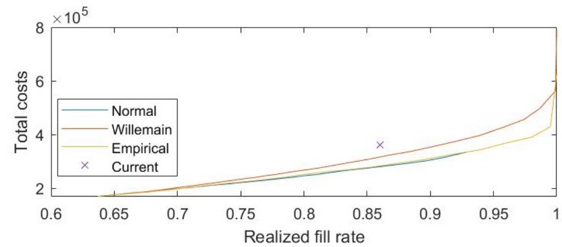

Figure 8: Class 233: Realized fill rate vs. costs

C.3 Class 343

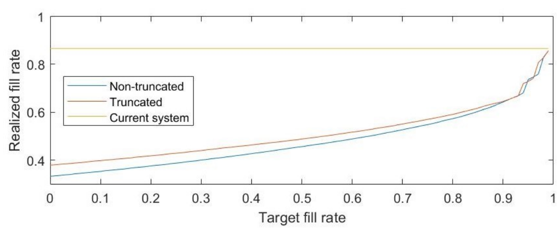

Figure 9: Class 343: target fill rate vs. realized fill rate

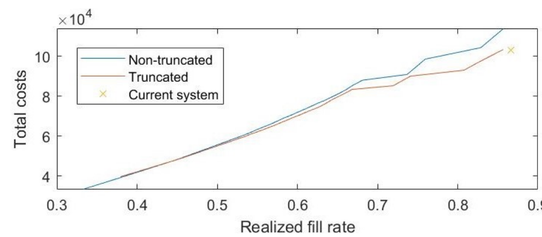

Figure 10: Class 343: realized fill rate vs. total costs

D Specific items

D.1 Output for items in class 212

Table 7: Current performance of some items in class 212.

Item #	(s,S)	#c	#soc	#D	#S	Total costs	CSL	fill rate	Price	Q
870	(1,2)	2	0	3	3	72	1.00	1.00	0.01	137
11216	(2,2)	2	0	4	4	72	1.00	1.00	0.01	112
11186	(2,2)	2	0	4	4	72	1.00	1.00	0.01	112
5982	(2,2)	2	1	4	2	72	0.50	0.50	0.01	112

Table 8: Performances of some items in class 212 of Willemain's method, the normal model and the empirical model. The results of this method are all the same.

Item #	#c	#soc	#D	#S	Total costs	CSL	fill rate	price	Q	s
870	0	0	3	3	1,55	1.00	1.00	0.01	137	0
11216	0	0	4	4	1,26	1.00	1.00	0.01	112	0
11186	0	0	4	4	1,26	1.00	1.00	0.01	112	0
5982	0	0	4	4	1,26	1.00	1.00	0.01	112	0

D.2 Item #1835 and item # 7559

Table 9: Details about items #1835 and #7559

Item #1835, Class 334 L = 31 Days (2 months) Price = 236 euros											
Demand	10	10	24	76							
Period	10	12	49	52							
Item #7559, Class 124 L = 44 Days (2 months) Price = 485 euros											
Demand	1	1	1	1	1	1	1	1	1	1	1
Period	6	9	12	13	18	26	29	39	47	49	55

D.2.1 Item #1835 replicated methods

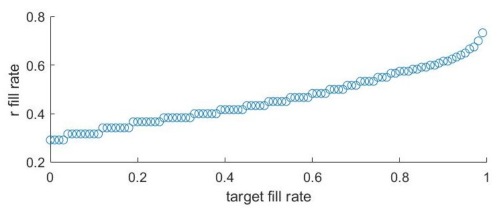

Figure 11: Normal model

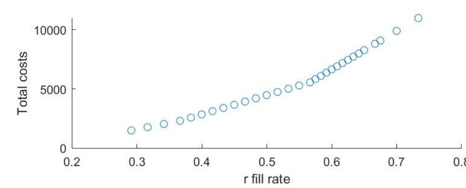

Figure 12: Normal model

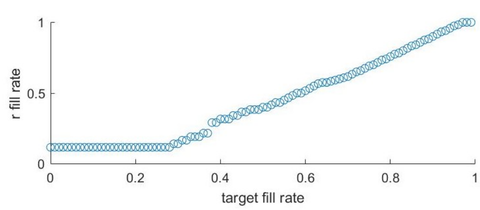

Figure 13: Empirical model

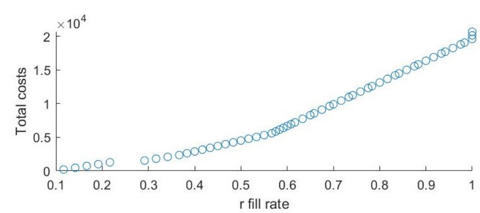

Figure 14: Empirical model

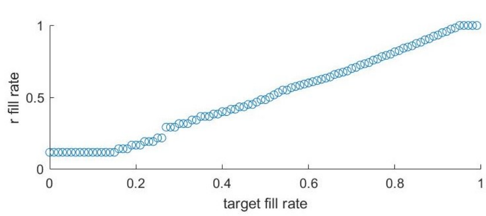

Figure 15: Willemain's method

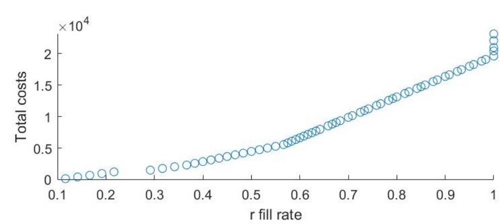

Figure 16: Willemain's method

D.2.2 Item #7559 replicated methods

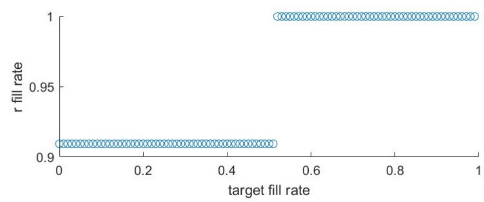

Figure 17: Normal model

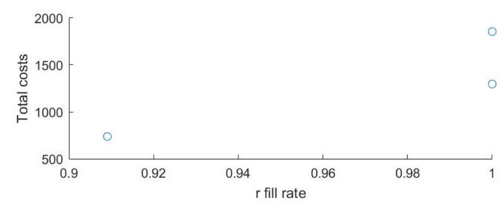

Figure 18: Normal model

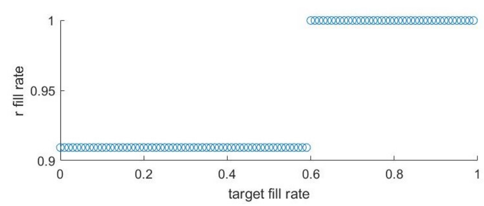

Figure 19: Empirical model

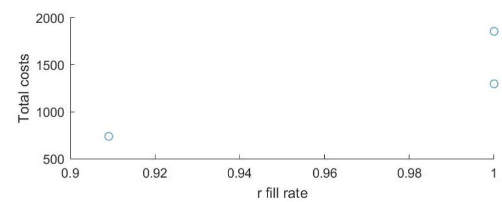

Figure 20: Empirical model

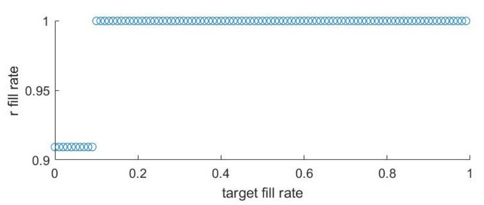

Figure 21: Willemain's method

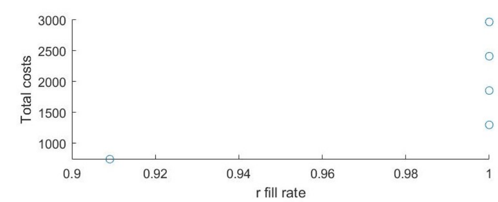

Figure 22: Willemain's method

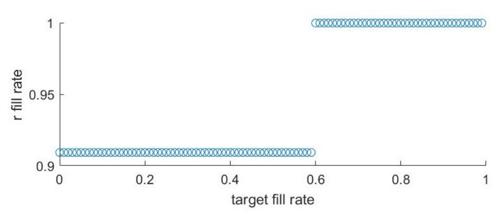

Figure 23: Poisson model

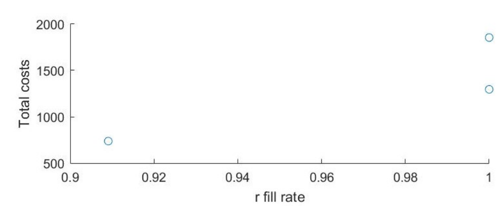

Figure 24: Poisson model

D.2.3 Item #1835 extension methods

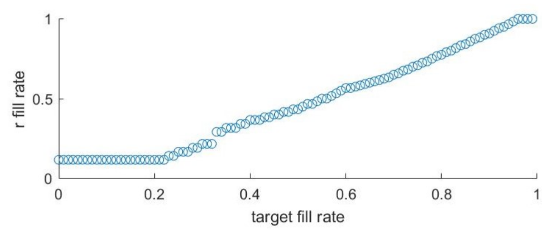

Figure 25: Revised Willemain's method

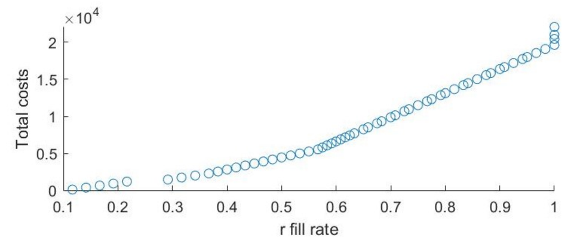

Figure 26: Revised Willemain's method

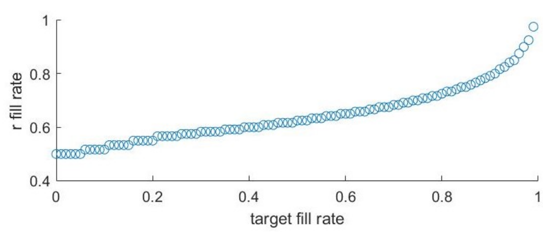

Figure 27: Croston's method

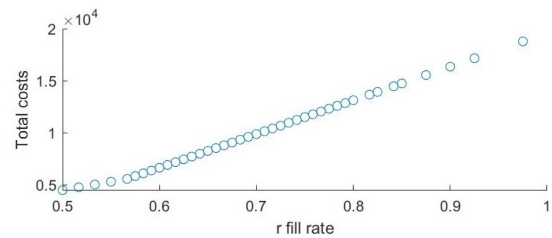

Figure 28: Croston's method

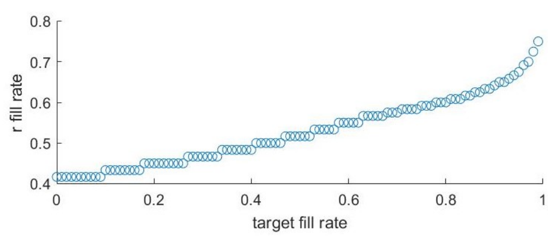

Figure 29: Truncated normal based model

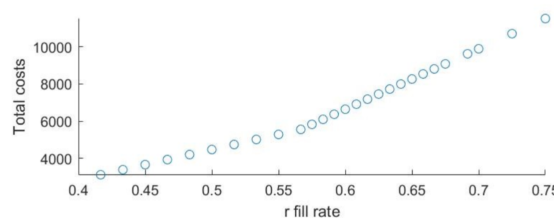

Figure 30: Truncated normal based model

E Optimization results

E.1 Normal model and Poisson model

Table 10: This table contains the results of the current system (C), the normal model (N) and the Poisson model (P).

		C			N			P		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
112	15	1,000	1,000	1504	-37	1,000	1,000	-37	1,000	1,000
113	47	0,943	0,944	9175	542	0,944	0,944	542	0,944	0,944
114	60	0,898	0,878	49888	2881	0,894	0,872	2881	0,894	0,872
115	17	0,895	0,895	94276	9096	0,895	0,895	9096	0,895	0,895
122	17	1,000	1,000	2712	-18	1,000	1,000	-32	1,000	1,000
123	43	1,000	1,000	12908	2424	1,000	1,000	2424	1,000	1,000
124	36	0,989	0,989	44224	13887	0,983	0,983	13887	0,983	0,983
125	6	0,952	0,952	23715	-3632	1,000	1,000	-3632	1,000	1,000
132	155	0,976	0,981	17529	-2090	0,972	0,972	-2090	0,972	0,972
133	357	0,969	0,968	93390	18523	0,961	0,959	18523	0,961	0,959
134	462	0,950	0,948	412925	60800	0,944	0,938	60800	0,944	0,938
135	31	0,935	0,935	138242	11027	0,935	0,935	11027	0,935	0,935
142	65	1,000	1,000	7123	-2680	0,992	0,992	-2680	0,992	0,992
143	166	0,977	0,978	44628	9485	0,971	0,968	9485	0,971	0,968
144	159	0,949	0,923	139985	24907	0,942	0,920	24907	0,942	0,920
145	9	1,000	0,962	35338	4188	1,000	0,962	4188	1,000	0,962
152	16	1,000	1,000	1427	-245	1,000	1,000	-245	1,000	1,000
153	40	0,944	0,949	10694	2166	0,956	0,956	2166	0,956	0,956
154	76	0,931	0,931	68700	11627	0,922	0,922	11627	0,922	0,922
155	9	0,955	0,955	77282	39633	0,955	0,955	39633	0,955	0,955
162	50	1,000	1,000	5993	-592	1,000	1,000	-592	1,000	1,000
163	158	0,981	0,981	40336	10438	0,973	0,973	10438	0,973	0,973
164	205	0,930	0,920	203485	39737	0,925	0,914	39737	0,925	0,914
165	45	0,959	0,952	254970	355	0,952	0,946	-3047	0,959	0,952
212	66	0,855	0,847	8604	4348	0,824	0,981			
213	123	0,896	0,866	47129	-945	0,845	0,894			
214	88	0,845	0,839	172149	-5920	0,810	0,847			
215	27	0,808	0,776	301035	-2972	0,688	0,800			
222	113	0,953	0,979	19695	13155	0,857	0,986			
223	129	0,948	0,933	62411	15319	0,913	0,950			
224	60	0,939	0,907	90279	13182	0,932	0,905			
232	499	0,878	0,885	71385	44397	0,844	0,984			
233	735	0,887	0,862	360604	66577	0,839	0,883			
234	526	0,885	0,838	1009533	50300	0,836	0,835			

Table 10: This table contains the results of the current system (C), the normal model (N) and the Poisson model (P).

		C			N			P		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
235	39	0,835	0,657	402269	-37373	0,567	0,649			
242	285	0,943	0,950	39619	24615	0,878	0,991			
243	391	0,917	0,887	160871	37582	0,869	0,908			
244	202	0,893	0,878	261082	15954	0,881	0,873			
252	30	0,921	0,918	5865	3603	0,786	0,972			
253	76	0,857	0,798	33150	3767	0,831	0,901			
254	82	0,880	0,861	152802	17828	0,831	0,859			
255	12	0,919	0,933	125528	37946	0,892	0,933			
262	594	0,697	0,827	113312	60022	0,719	0,966			
263	803	0,773	0,792	304882	49286	0,731	0,860			
264	315	0,856	0,839	634043	80511	0,841	0,832			
265	38	0,828	0,807	388464	25290	0,795	0,856			
312	15	0,776	0,766	9252	1353	0,700	0,756			
313	11	0,729	0,809	81006	-2043	0,660	0,829			
322	25	0,629	0,889	14976	4959	0,686	0,862			
323	6	0,875	0,548	8385	-370	0,667	0,599			
332	145	0,773	0,863	80204	30080	0,598	0,888			
333	43	0,696	0,743	94218	-2123	0,588	0,751			
334	9	0,676	0,722	44279	-59	0,619	0,732			
342	35	0,798	0,906	16113	6773	0,548	0,859			
343	9	0,742	0,866	102969	-10863	0,756	0,857			
352	7	0,538	0,736	1942	-77	0,500	0,886			
362	287	0,750	0,850	131782	-849	0,588	0,901			
363	39	0,688	0,681	76429	17514	0,640	0,724			
413	14	0,967	0,969	2156	142	0,958	0,969			
414	26	0,967	0,971	37309	2307	0,967	0,971			
415	18	1,000	1,000	110768	-52449	1,000	1,000			
423	8	1,000	1,000	3446	1025	1,000	1,000			
424	11	0,977	0,980	15350	1274	0,977	0,980			
432	14	0,947	0,947	1476	1208	1,000	1,000			
433	52	0,978	0,982	14766	3191	0,973	0,977			
434	91	0,953	0,959	116889	12396	0,946	0,953			
435	62	0,924	0,940	525604	10330	0,924	0,940			
442	8	1,000	1,000	1048	879	1,000	1,000			
443	36	0,976	0,985	9737	3430	0,983	0,993			
444	39	0,981	0,978	47602	9729	0,974	0,978			
445	6	0,900	0,923	32096	-6205	1,000	1,000			

Table 10: This table contains the results of the current system (C), the normal model (N) and the Poisson model (P).

		C		N				P		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
453	8	1,000	1,000	2267	363	1,000	1,000			
454	26	0,944	0,955	34334	-529	0,947	0,955			
455	7	1,000	1,000	42668	2534	1,000	1,000			
463	5	0,941	0,952	1100	23	1,000	1,000			
464	30	0,962	0,976	40737	2616	0,962	0,976			
465	41	0,949	0,959	352887	-7852	0,962	0,972			
Total	8610	0,878	0,886	8608955	797603	0,844	0,915	249006	0,955	0,950

E.2 Empirical model and Willemain's method

Table 11: Results of the current system (C), empirical model (E), and Willemain's bootstrap (W)

		C			E			W		
class	# items	rCSL	fill rate	costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
112	15	1,000	1,000	1504	-37	1,000	1,000	-48	1,000	1,000
113	47	0,943	0,944	9175	542	0,944	0,944	542	0,944	0,944
114	60	0,898	0,878	49888	2881	0,894	0,872	2464	0,911	0,894
115	17	0,895	0,895	94276	9096	0,895	0,895	6610	0,895	0,895
122	17	1,000	1,000	2712	-32	1,000	1,000	-86	1,000	1,000
123	43	1,000	1,000	12908	2673	0,995	0,995	1867	1,000	1,000
124	36	0,989	0,989	44224	13887	0,983	0,983	13887	0,983	0,983
125	6	0,952	0,952	23715	-3632	1,000	1,000	-3632	1,000	1,000
132	155	0,976	0,981	17529	-2090	0,972	0,972	-2090	0,972	0,972
133	357	0,969	0,968	93390	18382	0,961	0,959	17271	0,961	0,958
134	462	0,950	0,948	412925	60800	0,944	0,938	62954	0,944	0,938
135	31	0,935	0,935	138242	11027	0,935	0,935	11027	0,935	0,935
142	65	1,000	1,000	7123	-2680	0,992	0,992	-2680	0,992	0,992
143	166	0,977	0,978	44628	9485	0,971	0,968	8635	0,976	0,973
144	159	0,949	0,923	139985	25549	0,938	0,915	25322	0,938	0,915
145	9	1,000	0,962	35338	4188	1,000	0,962	4188	1,000	0,962
152	16	1,000	1,000	1427	-245	1,000	1,000	-245	1,000	1,000
153	40	0,944	0,949	10694	2166	0,956	0,956	2191	0,949	0,949
154	76	0,931	0,931	68700	11627	0,922	0,922	8924	0,927	0,927
155	9	0,955	0,955	77282	39633	0,955	0,955	39633	0,955	0,955
162	50	1,000	1,000	5993	-592	1,000	1,000	-598	1,000	1,000
163	158	0,981	0,981	40336	10438	0,973	0,973	9708	0,975	0,973
164	205	0,930	0,920	203485	40524	0,922	0,910	39417	0,925	0,913
165	45	0,959	0,952	254970	355	0,952	0,946	-3047	0,959	0,952

Table 11: Results of the current system (C), empirical model (E), and Willemain's bootstrap (W)

		C			E			W		
class	# items	rCSL	fill rate	costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
212	66	0,855	0,847	8604	4040	0,853	0,985	3963	0,853	0,980
213	123	0,896	0,866	47129	-1217	0,714	0,863	-4276	0,735	0,862
214	88	0,845	0,839	172149	-12531	0,754	0,839	-25445	0,770	0,833
215	27	0,808	0,776	301035	-4241	0,675	0,824	-6984	0,675	0,835
222	113	0,953	0,979	19695	12343	0,929	0,994	12403	0,929	0,993
223	129	0,948	0,933	62411	11099	0,933	0,980	8097	0,910	0,952
224	60	0,939	0,907	90279	6128	0,914	0,926	-9652	0,918	0,901
232	499	0,878	0,885	71385	44039	0,856	0,989	41300	0,898	0,991
233	735	0,887	0,862	360604	33716	0,844	0,918	6899	0,851	0,900
234	526	0,885	0,838	1009533	-13852	0,781	0,842	-103308	0,790	0,833
235	39	0,835	0,657	402269	-117448	0,546	0,665	-250877	0,631	0,743
242	285	0,943	0,950	39619	23598	0,951	0,998	22039	0,988	1,000
243	391	0,917	0,887	160871	29774	0,867	0,925	18173	0,880	0,912
244	202	0,893	0,878	261082	8919	0,849	0,881	-18209	0,838	0,868
252	30	0,921	0,918	5865	3274	0,929	0,994	2970	0,929	0,996
253	76	0,857	0,798	33150	931	0,819	0,925	-926	0,808	0,908
254	82	0,880	0,861	152802	-3481	0,820	0,884	-14077	0,793	0,858
255	12	0,919	0,933	125528	33185	0,892	0,933	27818	0,919	0,950
262	594	0,697	0,827	113312	60567	0,662	0,968	56484	0,801	0,975
263	803	0,773	0,792	304882	30460	0,736	0,878	6193	0,752	0,871
264	315	0,856	0,839	634043	46865	0,830	0,831	-40340	0,837	0,832
265	38	0,828	0,807	388464	9234	0,803	0,860	-6357	0,769	0,818
312	15	0,776	0,766	9252	-313	0,667	0,810	-189	0,667	0,802
313	11	0,729	0,809	81006	-4503	0,528	0,801	-7489	0,566	0,805
322	25	0,629	0,889	14976	4294	0,627	0,896	4344	0,608	0,893
323	6	0,875	0,548	8385	-505	0,417	0,585	-290	0,417	0,577
332	145	0,773	0,863	80204	18806	0,736	0,958	18236	0,732	0,957
333	43	0,696	0,743	94218	-11772	0,489	0,737	-11991	0,505	0,735
334	9	0,676	0,722	44279	-5	0,571	0,720	-170	0,643	0,731
342	35	0,798	0,906	16113	4347	0,839	0,951	4053	0,823	0,938
343	9	0,742	0,866	102969	-2329	0,644	0,904	3258	0,733	0,892
352	7	0,538	0,736	1942	-46	0,300	0,805	-19	0,300	0,786
362	287	0,750	0,850	131782	52867	0,871	0,794	47242	0,873	0,799
363	39	0,688	0,681	76429	-1043	0,503	0,779	-2509	0,534	0,782
413	14	0,967	0,969	2156	142	0,958	0,969	142	0,958	0,969
414	26	0,967	0,971	37309	2307	0,967	0,971	2781	0,967	0,971
415	18	1,000	1,000	110768	-52449	1,000	1,000	-66682	1,000	1,000

Table 11: Results of the current system (C), empirical model (E), and Willemain's bootstrap (W)

		C			E			W		
class	# items	rCSL	fill rate	costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
423	8	1,000	1,000	3446	1200	1,000	1,000	944	1,000	1,000
424	11	0,977	0,980	15350	-1079	0,977	0,980	-1079	0,977	0,980
432	14	0,947	0,947	1476	1208	1,000	1,000	1208	1,000	1,000
433	52	0,978	0,982	14766	3128	0,973	0,982	2863	0,973	0,977
434	91	0,953	0,959	116889	12651	0,942	0,950	12651	0,942	0,950
435	62	0,924	0,940	525604	10330	0,924	0,940	10330	0,924	0,940
442	8	1,000	1,000	1048	879	1,000	1,000	879	1,000	1,000
443	36	0,976	0,985	9737	3430	0,983	0,993	3430	0,983	0,993
444	39	0,981	0,978	47602	10232	0,974	0,978	5622	0,974	0,978
445	6	0,900	0,923	32096	3927	0,900	0,923	3927	0,900	0,923
453	8	1,000	1,000	2267	363	1,000	1,000	363	1,000	1,000
454	26	0,944	0,955	34334	-529	0,947	0,955	308	0,947	0,955
455	7	1,000	1,000	42668	2534	1,000	1,000	2534	1,000	1,000
463	5	0,941	0,952	1100	23	1,000	1,000	23	1,000	1,000
464	30	0,962	0,976	40737	2616	0,962	0,976	2616	0,962	0,976
465	41	0,949	0,959	352887	-7852	0,962	0,972	-7852	0,962	0,972
Total	8610	0,878	0,886	8608955	512207	0,849	0,920	-2413	0,866	0,917

E.3 Croston's method and the truncated normal model

Table 12: Results of the current system (C), Croston's method (Cr) and the truncated normal model (TN)

		C			Cr			TN		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
112	15	1,000	1,000	1504	-37	1,000	1,000	-37	1,000	1,000
113	47	0,943	0,944	9175	542	0,944	0,944	542	0,944	0,944
114	60	0,898	0,878	49888	2881	0,894	0,872	2881	0,894	0,872
115	17	0,895	0,895	94276	9096	0,895	0,895	9096	0,895	0,895
122	17	1,000	1,000	2712	-14	1,000	1,000	-18	1,000	1,000
123	43	1,000	1,000	12908	2694	0,995	0,995	2424	1,000	1,000
124	36	0,989	0,989	44224	13887	0,983	0,983	13887	0,983	0,983
125	6	0,952	0,952	23715	-3632	1,000	1,000	-3632	1,000	1,000
132	155	0,976	0,981	17529	-2089	0,975	0,975	-2090	0,972	0,972
133	357	0,969	0,968	93390	19010	0,960	0,958	18363	0,961	0,959
134	462	0,950	0,948	412925	60441	0,948	0,942	57184	0,944	0,938
135	31	0,935	0,935	138242	11027	0,935	0,935	11027	0,935	0,935
142	65	1,000	1,000	7123	-2680	0,992	0,992	-2680	0,992	0,992
143	166	0,977	0,978	44628	9485	0,971	0,968	9401	0,971	0,968
144	159	0,949	0,923	139985	24186	0,940	0,920	24907	0,942	0,920
145	9	1,000	0,962	35338	4188	1,000	0,962	4188	1,000	0,962
152	16	1,000	1,000	1427	-245	1,000	1,000	-245	1,000	1,000
153	40	0,944	0,949	10694	2191	0,949	0,949	2166	0,956	0,956
154	76	0,931	0,931	68700	12786	0,927	0,927	11627	0,922	0,922
155	9	0,955	0,955	77282	39633	0,955	0,955	39633	0,955	0,955
162	50	1,000	1,000	5993	-592	1,000	1,000	-592	1,000	1,000
163	158	0,981	0,981	40336	10487	0,975	0,975	10438	0,973	0,973
164	205	0,930	0,920	203485	40057	0,924	0,912	39094	0,925	0,916
165	45	0,959	0,952	254970	62	0,959	0,946	355	0,952	0,946
212	66	0,855	0,847	8604	3274	0,912	0,997	4374	0,824	0,984
213	123	0,896	0,866	47129	-2348	0,718	0,862	45	0,849	0,894
214	88	0,845	0,839	172149	-495	0,758	0,836	-1658	0,782	0,845
215	27	0,808	0,776	301035	-12864	0,740	0,835	-501	0,727	0,800
222	113	0,953	0,979	19695	12202	0,929	0,994	13236	0,857	0,988
223	129	0,948	0,933	62411	13496	0,910	0,960	15946	0,910	0,959
224	60	0,939	0,907	90279	9189	0,896	0,913	14626	0,910	0,898
232	499	0,878	0,885	71385	44413	0,850	0,978	44553	0,862	0,987
233	735	0,887	0,862	360604	35080	0,843	0,906	60725	0,857	0,904
234	526	0,885	0,838	1009533	7597	0,836	0,843	55017	0,836	0,845
235	39	0,835	0,657	402269	-229118	0,667	0,751	-40342	0,589	0,649
242	285	0,943	0,950	39619	23962	0,927	0,996	24608	0,902	0,992
243	391	0,917	0,887	160871	27193	0,875	0,932	36149	0,877	0,923

Table 12: Results of the current system (C), Croston's method (Cr) and the truncated normal model (TN)

		C			Cr			TN		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
244	202	0,893	0,878	261082	7284	0,844	0,869	25356	0,875	0,881
252	30	0,921	0,918	5865	2967	0,929	0,996	3572	0,786	0,976
253	76	0,857	0,798	33150	681	0,814	0,914	4146	0,814	0,900
254	82	0,880	0,861	152802	11491	0,834	0,867	15503	0,837	0,876
255	12	0,919	0,933	125528	29762	0,892	0,933	37946	0,892	0,933
262	594	0,697	0,827	113312	60317	0,693	0,965	61386	0,656	0,960
263	803	0,773	0,792	304882	35042	0,725	0,870	52322	0,729	0,866
264	315	0,856	0,839	634043	55113	0,839	0,832	95673	0,828	0,831
265	38	0,828	0,807	388464	15118	0,778	0,860	8198	0,803	0,874
312	15	0,776	0,766	9252	-655	0,233	0,762	1174	0,733	0,766
313	11	0,729	0,809	81006	792	0,679	0,825	1127	0,679	0,824
322	25	0,629	0,889	14976	3444	0,588	0,899	4754	0,667	0,872
323	6	0,875	0,548	8385	-37	0,542	0,601	-600	0,708	0,621
332	145	0,773	0,863	80204	15888	0,724	0,935	28633	0,644	0,902
333	43	0,696	0,743	94218	-18127	0,451	0,739	-1932	0,577	0,760
334	9	0,676	0,722	44279	-5125	0,548	0,715	-62	0,524	0,714
342	35	0,798	0,906	16113	5071	0,774	0,927	6555	0,565	0,878
343	9	0,742	0,866	102969	341	0,711	0,895	-469	0,733	0,857
352	7	0,538	0,736	1942	-46	0,400	0,858	-8	0,500	0,877
362	287	0,750	0,850	131782	60160	0,883	0,761	76906	0,677	0,787
363	39	0,688	0,681	76429	-88	0,559	0,747	14994	0,665	0,746
413	14	0,967	0,969	2156	142	0,958	0,969	142	0,958	0,969
414	26	0,967	0,971	37309	1012	0,967	0,971	2307	0,967	0,971
415	18	1,000	1,000	110768	-62216	1,000	1,000	-52449	1,000	1,000
423	8	1,000	1,000	3446	917	1,000	1,000	1540	1,000	1,000
424	11	0,977	0,980	15350	1274	0,977	0,980	-1079	0,977	0,980
432	14	0,947	0,947	1476	1208	1,000	1,000	1208	1,000	1,000
433	52	0,978	0,982	14766	3007	0,973	0,977	3274	0,964	0,977
434	91	0,953	0,959	116889	12438	0,942	0,950	12651	0,942	0,950
435	62	0,924	0,940	525604	10330	0,924	0,940	10330	0,924	0,940
442	8	1,000	1,000	1048	879	1,000	1,000	879	1,000	1,000
443	36	0,976	0,985	9737	3430	0,983	0,993	3430	0,983	0,993
444	39	0,981	0,978	47602	11738	0,974	0,972	8566	0,974	0,978
445	6	0,900	0,923	32096	1269	1,000	1,000	-9698	1,000	1,000
453	8	1,000	1,000	2267	363	1,000	1,000	363	1,000	1,000
454	26	0,944	0,955	34334	308	0,947	0,955	-529	0,947	0,955
455	7	1,000	1,000	42668	2534	1,000	1,000	2534	1,000	1,000
463	5	0,941	0,952	1100	23	1,000	1,000	23	1,000	1,000

Table 12: Results of the current system (C), Croston's method (Cr) and the truncated normal model (TN)

		C			Cr			TN		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
464	30	0,962	0,976	40737	2616	0,962	0,976	2616	0,962	0,976
465	41	0,949	0,959	352887	-2534	0,949	0,966	-7852	0,962	0,972
Total	8610	0,878	0,886	8608955	453074	0,854	0,917	888129	0,846	0,916

E.4 Revised Willemain's method

Table 13: Results of the current system (C) & Revised bootstrap with bound determined by average lead time demand (L) and determined by mean (M).

		C			L			M		
class	# items	rCSL	r fill rate	Costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
312	15	0,776	0,766	9252	829	0,731	0,880	850	0,769	0,897
313	11	0,729	0,809	81006	4513	0,698	0,799	3661	0,717	0,808
322	25	0,629	0,889	14976	-2359	0,531	0,894	-2978	0,531	0,890
323	6	0,875	0,548	8385	-66	0,294	0,658	-34	0,294	0,658
332	145	0,773	0,863	80204	24267	0,796	0,906	26566	0,770	0,898
333	43	0,696	0,743	94218	-5628	0,485	0,742	-5971	0,485	0,746
334	9	0,676	0,722	44279	-717	0,548	0,763	-294	0,581	0,756
342	35	0,798	0,906	16113	1379	0,910	0,988	1536	0,866	0,987
343	9	0,742	0,866	102969	-3852	0,676	0,901	-1548	0,618	0,897
352	7	0,538	0,736	1942	-117	0,357	0,833	-4	0,214	0,838
362	287	0,750	0,850	131782	12180	0,746	0,742	13184	0,713	0,738
363	39	0,688	0,681	76429	-10480	0,324	0,673	-10623	0,316	0,686
Total	631	0,744	0,832	651925	18062	0,701	0,802	22460	0,676	0,800

Table 14: Results of the current system (C) & Revised bootstrap with bound determined by re-order point (S) and determined by *EOQ* and re-order point (QS).

		C			S			QS		
class	# items	rCSL	r fill rate	costs	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
312	15	0,776	0,766	9252	1038	0,769	0,904	790	0,769	0,927
313	11	0,729	0,809	81006	6464	0,679	0,808	3357	0,679	0,830
322	25	0,629	0,889	14976	-2736	0,490	0,894	-2176	0,510	0,890
323	6	0,875	0,548	8385	-517	0,294	0,665	-195	0,294	0,660
332	145	0,773	0,863	80204	27463	0,732	0,895	24466	0,781	0,908
333	43	0,696	0,743	94218	-2041	0,558	0,742	-5715	0,497	0,745
334	9	0,676	0,722	44279	-55	0,581	0,755	-72	0,581	0,757
342	35	0,798	0,906	16113	1601	0,866	0,979	1624	0,881	0,983
343	9	0,742	0,866	102969	-676	0,618	0,872	-877	0,647	0,883
352	7	0,538	0,736	1942	-22	0,357	0,825	-22	0,357	0,825
362	287	0,750	0,850	131782	26863	0,713	0,726	12927	0,746	0,735
363	39	0,688	0,681	76429	-1796	0,338	0,733	-9690	0,346	0,698
Total	631	0,744	0,832	651925	55588	0,673	0,796	22532	0,698	0,802

E.5 Joint results

Table 15: Results joint classes of the current system (C), Normal model (N) and Poisson model (P)

		C			N			P		
Dem	# items	costs	rCSL	r fill rate	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
1	2244	1790450	0,960	0,956	252422	0,955	0,949	249006	0,955	0,950
2	5220	4764710	0,852	0,855	516471	0,814	0,901			
3	631	661556	0,744	0,832	44298	0,602	0,862			
4	502	1392239	0,961	0,968	-15587	0,963	0,971			
Critic										
1	527	924251	0,887	0,876	-43697	0,855	0,901	12482	0,923	0,913
2	454	298101	0,941	0,949	61206	0,907	0,955	12648	0,994	0,994
3	3220	3383313	0,904	0,894	267185	0,865	0,912	88260	0,954	0,951
4	1410	898211	0,933	0,925	117795	0,898	0,936	35900	0,964	0,953
5	389	556660	0,906	0,895	118616	0,878	0,921	53181	0,943	0,943
6	2610	2548420	0,801	0,844	276498	0,771	0,902	46535	0,953	0,948
Price										
2	2441	551561	0,833	0,891	188806	0,784	0,960	-5676	0,984	0,985
3	3299	1576656	0,883	0,877	225454	0,849	0,903	43578	0,967	0,965
4	2503	3575596	0,911	0,893	353428	0,892	0,888	153839	0,937	0,927
5	367	2905142	0,910	0,890	29916	0,872	0,899	57265	0,948	0,943
Total	8610	8608955	0,878	0,886	797604	0,844	0,915	249006	0,955	0,950

Table 16: Results of joint classes of the current system (C), Empirical model (E) and Willemain's bootstrap (W)

		C			E			W		
Dem	# items	costs	rCSL	r fill rate	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
1	2244	1790450	0,960	0,956	253945	0,954	0,949	242216	0,956	0,950
2	5220	4764710	0,852	0,855	205402	0,805	0,913	-274112	0,832	0,907
3	631	661556	0,744	0,832	59798	0,751	0,839	54476	0,755	0,839
4	502	1392239	0,961	0,968	-6938	0,961	0,970	-24992	0,961	0,969
Critic										
1	527	924251	0,887	0,876	-56283	0,815	0,895	-94612	0,825	0,897
2	454	298101	0,941	0,949	46378	0,921	0,970	26804	0,915	0,958
3	3220	3383313	0,904	0,894	68921	0,863	0,925	-183696	0,874	0,921
4	1410	898211	0,933	0,925	119319	0,913	0,945	78637	0,923	0,940
5	389	556660	0,906	0,895	89412	0,881	0,931	69474	0,874	0,923
6	2610	2548420	0,801	0,844	244460	0,787	0,896	100980	0,826	0,896
Price										
2	2441	551561	0,833	0,891	224228	0,831	0,957	209168	0,878	0,959
3	3299	1576656	0,883	0,877	136584	0,842	0,918	63116	0,850	0,910
4	2503	3575596	0,911	0,893	213508	0,873	0,891	-35333	0,875	0,887
5	367	2905142	0,910	0,890	-62113	0,868	0,901	-239364	0,875	0,907
Total	8610	8608955	0,878	0,886	512207	0,849	0,920	-2412	0,866	0,917

Table 17: Joint results of revised Willemain's method with bound depending on re-order point S (S), Croston's method and the truncated normal model (TN)

		S			Cr			TN		
Dem	# items	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate	Savings	rCSL	r fill rate
1	2244				253365	0,955	0,950	247920	0,955	0,950
2	5220				149356	0,813	0,909	530880	0,812	0,908
3	631	55588	0,673	0,796	61617	0,745	0,815	131072	0,654	0,817
4	502				-11264	0,961	0,969	-21743	0,962	0,970
Critic										
1	527	7503	0,731	0,863	-60877	0,818	0,896	-32956	0,855	0,902
2	454	-3253	0,452	0,85	53420	0,912	0,963	61086	0,903	0,958
3	3220	25367	0,687	0,855	-34020	0,872	0,922	258540	0,873	0,920
4	1410	926	0,815	0,957	116345	0,909	0,945	131191	0,904	0,942
5	389	-22	0,357	0,825	102424	0,885	0,927	116708	0,876	0,924
6	2610	25067	0,668	0,727	275781	0,794	0,890	353559	0,765	0,890
Price										
2	2441	54208	0,715	0,797	227427	0,834	0,948	266169	0,789	0,948
3	3299	1435	0,481	0,752	144315	0,841	0,913	234559	0,853	0,912
4	2503	-55	0,581	0,755	268677	0,886	0,890	378568	0,888	0,891
5	367				-187345	0,884	0,912	8833	0,878	0,900
Total	8610	55588	0,673	0,796	453074	0,854	0,917	888129	0,846	0,916