22nd International Conference on Modelling and Applied Simulation
20th International Multidisciplinary Modeling & Simulation Multiconference

2724-0037 © 2023 The Authors.
doi: xx.xxxx/xxxx

Determination of an optimal inventory system, and safety factor, in a MSME, through Monte Carlo simulation

Miguel Angel Andoaga-Mejia [1, *], Susana Casy Téllez-Ballesteros [2] and Hilda Reyna Solis-Vivanco [3]

[1,2,3] School of Engineering, National Autonomous University of Mexico (UNAM), Av. Universidad 3000, Ciudad Universitaria, Coyoacán, Cd. Mx., CP 04510, México

[*] Email address: miguel.andoaga@ingenieria.unam.edu

Abstract

For a newly opened MSME, defining an inventory investment strategy means the difference between surviving or closing operations. This article describes an inventory control system, which analyzes item classification and reorder point. Items are classified based on their utility and individual demand, using the ABC and FSN classifications, respectively. Both rankings are grouped into three macro categories: AA, BB and CC, to determine an inventory model for each group. The reorder point is evaluated between the periodic inventory review policy (Q, R) and (s, S) and the safety stock. Using a Monte Carlo simulation, which despite having little data to model the demand for each SKU, provides a useful guide to define an optimal inventory policy for an emerging business.

Keywords: Inventory model, inventory classification, Monte Carlo simulation, triangular distribution

1. Introduction

Inventory management is used to control the items that become obsolete and that are stored to supply the demand, these items correspond to perishable or short-lived products. (Gupta, 2020). Inventories in a company are a very relevant aspect, both in MSMEs and in larger businesses, since poor administration can generate discontent among customers; Therefore, having control in a sophisticated way implies having a greater supervision of the stock, reducing cost, and accelerating the fulfillment of the demand, especially if it behaves in a variable way.

This paper is about a Mexican MSME, focused on the sale of supplies for DIY, construction, and household needs. This MSME is replenishing its inventory weekly with direct purchase at its suppliers' stores. Units purchased for each product are based on the owner's perception, not necessarily product demand.

In addition, since the start of the business operation is new, the record of purchases and sales is small, so a demand forecast is very difficult with the few data collected. The objective of the paper is to describe an inventory policy that reduces the total inventory cost using Monte Carlo simulation.

First, we carried out a double criteria product classification analysis, this stage allowed us to identify the products with the highest profit and the best sellers.

Second, we use inventory theory to calculate the minimum quantity of each product. The third step was to determine the reorder point using a Monte Carlo simulation of the inventory over a year contemplating the inventory models (Q, R), (s, S), both with a periodic review system, and a safety factor k, to determine the extra units that cushion the difference between supply and demand. Finally, we evaluate the total annual cost of inventory and determine an optimal inventory system for purchasing items.

2. State of the art

Sipper (1998) described Pareto inventory classification to differentiate the strategic inventory support decision. The inventory policy was the A items must have a periodic system review, by other way the B and C items must have inventory continuous system review. The Parekh et al. (2008). research described a classification methodology with the objective of making the right decision and the right time in terms of inventory resources, it was describing the ABC, FSN, and VED inventory classification, an important step of the support inventory decision.

Cavaliere et al. (2008) developed a decision-making framework for managing maintenance as an engineering way of managing maintenance processes. Them was developing five sequential steps for inventory decision making: (1) part coding, (2) part classification, (3) part demand forecasting, (4) stock management policy, (5) policy test and validation.

Jimenez (2014) developed a bachelor thesis about an inventory control system in a pharmaceutical group. He was describing ABC double classification items, which means the use of two classification criteria. So, he proposes three classifications AA, BB and CC that cover the previous ones.

In 2001, Chandra and Kumar proposed an inventory management approach in the textile sector. First, they used the Silver and Peterson fixed order quantity systems models to determine inventory categories. Then they defined the order point using the Gaither model to form a pull production philosophy for each category.

In 2017, Muller, et Al., analyzed how data uncertainty distributions affects the modeling of life cycle, and then concluded the switch showing practically no effect on product system comparisons.

The stock management policy developed by Niles Joshi (2010) described the integrated risk-based inventory classification method (IRIC method). This methodology is more cost effective compared to only applying ABC analysis, At the analysis methodology used Monte Carlo simulation with triangular distribution demand input.

Table 1. Author Contributions.

Autor	Inventory Classification	Inventory system
Sipper	ABC classification	Several
Parekh et al	ABC, FSN, VED classification	None
Cavaliere	ABC, FSN, VED classification	Maintenance products inventory
Jimenez	Double classification items	None
Chandra and Kumar	Peterson-Silver classification	Textile products and define reorder point
Muller, et Al	None	Demand Distribution effects
Niles Joshi	None	Monte Carlo simulation

Source: own elaboration

Following the previous work presented, the article uses a two-criteria inventory classification of the item's MSME. The criteria were worked with the utility of each product, and its demand, to define inventory policies by group, simulation with Monte Carlo let us calculate the reorder point. The demand input variable is considered a triangular distribution function, as Muller mentions, it does not affect the performance of the inventory system. Finally, we describe the effect on its cost.

3. Materials and Methods

3.1. Classification of products in inventory

The database provided by the MSME is a Microsoft Excel® calculation book, which contains a record of all sales and purchases of products from January 11 to December 9, 2022, until that moment it had a catalog of 1,733 products, each one already identified with its respective SKU.

It was desired to know the "star" products of the business, the ones that generate the most utility and the best-selling ones, for this reason it begins with an individual classification for both criteria, however, it must be emphasized that said venture in the period that the database registers began selling in a store, from the start of operation of said establishment, on May 1, 2022. An additional filter was applied (also supported by the filter tool of Excel), to study the products that have been sold more than twice, this is so that the products will have a minimum of sales data so that the distribution used to model their demand in the simulation is useful.

After applying both filters, there are 28 products on the market, however two of these implied losses. These products were left out of the analysis because they did not generate utility, the object of this study. Once these products were discarded, the total individual profit in the period was recorded, as well as the units sold as a measure of their transaction volume. Once this was done, the total profit and the total units sold were obtained.

An ABC analysis was carried out to observe the contribution of each SKU to the total utility, which is the quotient of the individual utility between the total utility, expressed as a percentage, based on these values, the products with the highest contribution are ordered to the one with the lowest, this allows to find the products that are in certain ranges to assign them a label:

- A, for the products that contribute 20% of the total utility.
- B, for those who contribute the next 30% of the total profit.
- C, for the products that contribute the rest of the utility.

The SKUs under category A are the ones that should be paid the most attention, ensuring that there are always stocks in inventories, that they are easily accessible and with an area assigned for their conservation, those of category B should also be monitored, but to a lesser extent that those of category A, and those of category C imply a minimal attention.

The second classification is the FSN, which classifies the products according to the movement of their units in the inventory in Fast, Slow and Non-movement. This analysis is carried out with the units sold; the commercialized units of each SKU must be placed. The products are ordered according to their units sold, from the highest quantity to the lowest, and a classification is assigned: The products whose sold units are greater than the third quartile are category F, if said value is between the first quartile and the third quartile then they are classified as S, if their commercialized units are below the first quartile, they are category N.

Now that each SKU has been assigned an ABC and FSN category, both are crossed to group them into three macro categories: AA, BB and CC, so that each group has its own inventory policy to keep the latter manageable; the macro category assignment is shown in Figure 1.

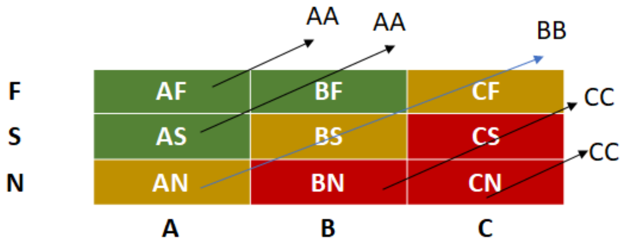

Figure 1. Macro category assignment according to ABC and FSN classifications.

The AA category are the most important products considering both criteria, this category includes the double classification AF, BF and AS. The items in the BB category are of medium importance, this category is the double classification AN, BS, and CF. Finally, the articles in consideration and the CC category are the ones that contribute the least to utility and the rest of the articles (double classification BN, CS and CN).

3.2. Inventory cost determination

The cost evaluation allows us to provide an indicator of comparison. Carrying and procurement cost calculations are described below.

3.2.1. *Carrying cost*

Carrying cost is calculated by dividing the annual rent, electricity, and telephone warehouse expenses by the total value of the inventory (see equation 1)

$$Carrying\ cost\ (i\%) = \frac{Inventory\ holding\ sum}{Total\ value\ of\ inventory} * 100 \quad (1)$$

The operating time at the warehouse was considered 7.96 months. The costs involved storing the items were multiplied by the equivalence factor between the months elapsed. Item carrying cost was estimated at 6.55 percent.

$$i = \frac{(67{,}680.00[\frac{\$}{year}] * 7.96\ [months] * \left(\frac{1[year]}{12\ [months]}\right)}{685{,}828.52[\$]} * 100 \quad (2)$$

3.2.2. *Procurement cost*

Procurement cost is calculated by salary of purchase staff, licenses, shipping cost and insurance delivery, software for purchases activities, among other costs.

The purchase and transportation time, the cost of delivery and insurance, as well as the cost of the ERP license, are the acquisition cost. For one year, the company spent an amount of $66,793.19, divided by 52 orders in the year, which is equivalent to 1284 ($/order) purchasing cost.

3.3. Demand probabilistic distribution.

Demand SKU registrations were very few. The average number of transactions for SKUs was 5 sales records. The sales transaction SKU range was between 3 and 11 records. Therefore, it was decided to use a triangular distribution for the probabilistic simulation of the demand. The advantage was that triangular distributions showed us good demand variance and can be used with little demand data, and the simplicity of its implementation in the simulation.

The demand analyzed was between May 1 and December 9, 2022, in said period there were 158 days in which the store was selling items, except non-working days. The triangular distribution parameters were minimum, maximum, and average units sold per item.

3.4. Economic order quantity and Reorder point.

Liang et. al. (2016) defines lead time as the time between the time a replacement order is placed and the time it arrives. For this company, inventory replenishment is time spent shopping at stores of suppliers and purchasing missing items.

The inventory model (Q, R) is a continuous review method (Wilson, 1934), that is, as soon as the inventory level is below R, Q units are ordered. A purchase order must be placed because it ensures that the item is going to arrive at lead time. The (Q, R) inventory system formulas are 3 and 4 (Hopp and Spearman, 2011).

$$Q = \sqrt{\frac{2A\bar{D}}{ic}} \tag{3}$$

Where:

Q = economic order quantity

$\bar{D}$ = average demand per unit of time

A = procurement cost

i = carrying cost

c= unit item cost

$$R = \bar{D}\tau + k\sigma\sqrt{\tau}$$
$$R = \bar{D}\tau + k\sigma_\tau \tag{4}$$

Where:

R = reorder point

τ = lead time

k =value in the standard normal table, this factor also let us evaluate the security factor

σ_τ = standard deviation of the lead-time demand

The inventory model (s, S) is a periodic review method, from time-to-time T the inventory level is reviewed, if it is less than s, the necessary units are requested to reach the target level of S units. For the system (s, S), a good approximation for the reorder point s and the target inventory quantity S is obtained with formula 5 (Sipper and Bulfin, 1998).

$$s = R$$
$$S = R + Q \tag{5}$$

Orders can only be ordered on the weekend in this simulation, when it is made the inventory order, so both inventory models will be considered as periodic review models.

3.5. Inventory simulation assumptions.

The aim is to simulate the behavior of the inventory for all items, week by week, throughout a year, if a (Q, R) or (s, S) model were implemented, with a randomly chosen safety factor k, being able to take values, with increments of 0.5, between zero and three.

The objective of the simulation is to evaluate the total inventory cost resulting from each method.

The input variables are:

- The demand data for each SKU, as parameters of its corresponding triangular distribution.
- The procurement cost.
- The carrying cost.
- The unit cost of each SKU.
- The average utility of each item.

The decision variables are:

- The inventory system used: (Q,R) or (s,S)
- The value of the safety factor k.

The performance indicator is the total annual cost of inventory.

The influence diagram of the model for the inventory is shown in figure 2.

The simulation assumptions are:

1. Inventory starts with zero units for all products.
2. Lead time is 2.5 days (half a week).
3. The supplier always has the product on time.
4. Inventory is checked at the end of each work week.
5. The weekly demand for each product is modeled by a triangular distribution, the parameter b, maximum value, cannot be less than 0.5, in which case it takes the value of 0.51.
   The fifth assumption occurs since the simulation model assigns a demand based on the distribution and rounds the result to the nearest integer. With a maximum value less than 0.5, there would never be weekly demand for the item that has it.

3.6. Input data file for simulation.

A spreadsheet is created in which the data that the simulation will use to calculate the total annual inventory cost according to a (Q, R) or (s, S) model and a model are placed. safety factor k.

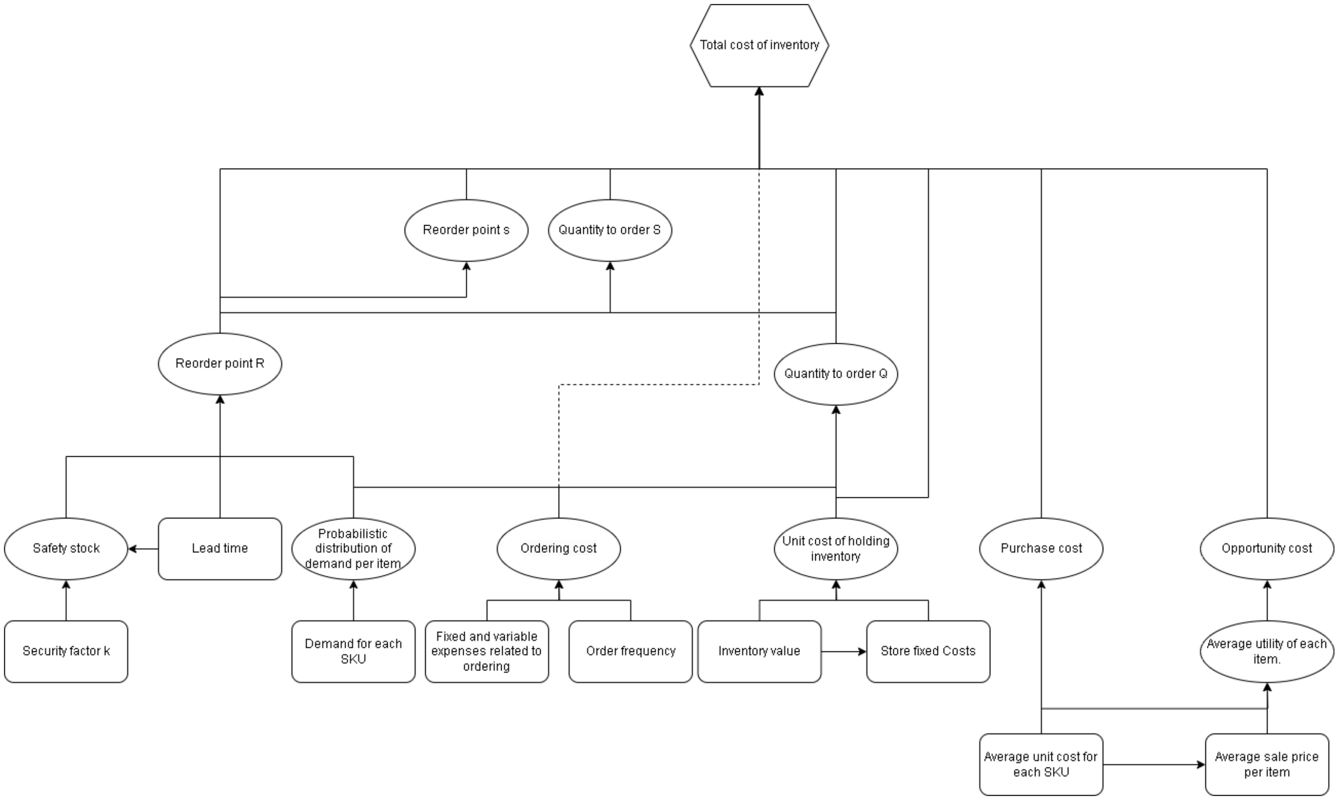

Figure 2. Influence diagram of the simulation model.

Source: Own elaboration

We generate a Microsoft Excel ® workbook, it has data from the input variables. This data has SKU code, average unit cost, average unit profit, minimum, maximum, mean, standard deviation, Q, R, and other information that software Python ® will use to simulate the inventory system.

These data are arranged in columns, the first data is taken from the calculations carried out previously and the last ones mentioned are filled with zeros, since the simulation requires it, as it was programmed.

The simulation result data table is not going to contain headers, it is going to save as a comma-separated (CSV) file called "Data".

3.7. Inventory simulation.

The simulation model is based on the equations of the inventory models (Q, R), (s, S), and on the total inventory cost formula considering ordering, buying, holding and opportunity costs.

The simulation is carried out over a period of one year, with weekly inventory updates. The model was written in the Python programming language.
The code is explained in detail below:

1. The numpy and random libraries are imported.
2. The user chooses the number of iterations in the simulation for each inventory system, entering an integer.
3. The iteration counter, n, is set to zero.
4. An empty Results list is created and with a while loop the iterations are carried out.
5. The simulation is started by establishing the model (Q, R), the data of each SKU previously prepared in the .csv file is loaded in a list called data, in this case, the file is in the same location as the file of python code, so it is not necessary to specify the location of Data.csv.
6. A safety factor value k is selected at random through a uniform distribution, being able to take values from 0 to 3 in steps of 0.5.
7. The value of R of each product is calculated, it is rounded to the nearest integer, if R is less than 1 then it takes the value of 1.
8. Weekly demand is assigned, using a triangular distribution for each SKU, rounded to the nearest integer.
9. It is verified for each SKU if the demand is greater than the inventory, in which case missing parts are added.

10. The respective demand is subtracted from the inventory of each product, if the result is negative then the inventory takes the value of zero.
11. The pieces that remained in inventory are added up, weekly, for each SKU, before buying, if that is the case.
12. It is checked for each product, if the inventory is below the reorder point, if so, a purchase order is posted and counted, the parts purchased, and the inventory is updated.
13. The week counter j is increased in one unit.
14. The total cost of inventory for each SKU is calculated by adding up the individual calculations of the cost to buy, the cost to order, the cost to hold (this is done considering the average inventory throughout the year) and the cost of opportunity.
15. A list called SKU is created, which contains the inventory system used, the safety factor k, the SKU code, and its respective total inventory cost. This is updated for all products, after adding this information to the Results list.
16. One iteration is counted.
17. At the end of the iterations requested for the system (Q, R), the value of the iteration counter, n, is reset to zero and the simulation begins using the system (s, S).
18. Step number five is repeated, resetting the values of the data list, and step number six: randomly calculating a value of k.
19. s and S are calculated, for each SKU, s is rounded to the nearest integer, if it is less than 1 then it takes the value of 1.
20. Repeat steps eight to eleven.
21. It is checked for each product, if the inventory is below the reorder point, if so, a purchase order is posted, the parts purchased, and the inventory is updated up to the target level S.
22. Repeat steps 13 to 16.
23. The simulation results are saved in a txt file.

Next, the Results.txt file is opened with MS Excel, in the wizard to import data delimited by text it is selected, then spaces as separators, data in columns as general format and finally the finish button is selected, with this we will open Results .txt as a workbook.

Meanwhile, in another Excel workbook, three worksheets have been set up, one for each category: AA, BB, and CC.

Four columns are placed on each sheet containing the inventory system used, the SKU code number, the value of k, and the total cost of inventory. In the AA tab we only have the data for SKU 1, in the BB tab, from SKU 2 to 10, and in CC from 11 to 26. In each tab with the AVERAGEIF function, there is the inventory, k and the SKU code established as conditions to obtain the average total cost of each product.

In another table, within the same sheet, three columns are placed, one with the system used, another with the k used and the sum of the total cost of crossing these data using the SUMIFS function. This sum is the total value of the policy with a given inventory system and value of k.

With this table it is possible to make a line graph and visually observe the behavior of the inventory policies.

3.8. Model verification.

The verification was the careful observation of the model according to how it was programmed, it must be verified that it is doing what was intended in the way it was devised, this is achieved by debugging the code with the help of Microsoft Visual Studio Code, with the extension of software Python®.

3.9. Model validation.

The developed model was presented to the user, with its assumptions and limitations, as well as its results. The advantage of monitoring sales and not just acquiring inventory without a strategy focused on profit was observed.

3.10. Simulation evaluation.

The number simulation iterations are evaluated with standard error. We calculate maximum standard error of the total annual inventory cost for each SKU in all possible combinations of model and safety factor.

A preliminary simulation was run with 1000 iterations, 500 for each model. We calculated the maximum standard error for each combination of model, factor of safety, and SKU, using Excel's STDEV function.

It should be noted that SKU 5 entailed a much higher variability than the rest of the articles, for this reason Table 1 is presented, which shows the maximum standard errors with and without considering SKU 5.

Table 2. Number of iterations vs maximum standard error ($)

Number of iterations for each model	Maximum standard error ($)	Maximum standard error without SKU 5 ($)
500	1271.64	811.42
1000	1146.28	771.05
2000	1219.78	748.78
4000	1197.86	763.38
8000	1172.53	756.19
16000	1164.45	757.07
20000	1138.42	749.21

Source: Own elaboration

In Table 2, we find that SKU 5 presents the highest standard deviation error. When we do the maximum standard error analysis and omit SKU 5, we see that the deviation error decreases in all iterations. Also, as interactions increase, the standard error decreases, with or without SKU 5 evaluation. Figure 3 shows the maximum value of the standard deviation of all calculated total annual costs, regardless of the model used or the value of k, against the number of iterations. The blue line considers all SKUs, the orange ignores the fifth SKU.

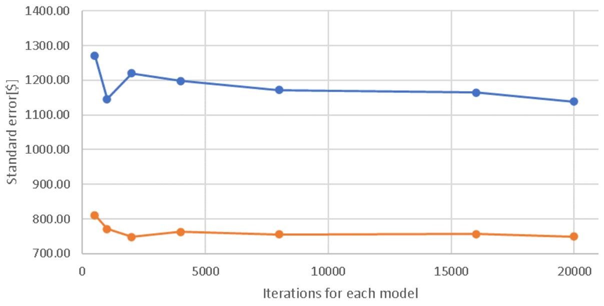

Figure 3. Standard deviation vs. simulation iterations number.

Source: Own elaboration

To reduce standard error, save resources and time, we decided doing 40,000 iterations, half for (Q, R) and the other for (s, S).

4. Results and Discussion

Once the number of iterations has been defined, the simulation results are described below. Only one product belongs to the AA macro category (SKU number 1). The inventory cost of this product increases as the factor of safety increases using the (Q, R) inventory model. Unlike the cost of inventory that is kept constant using the inventory model (s, S), the average cost of inventory is between 2,600 and 2,580 pesos from a value of k equal to one, as can be seen in figure 4.

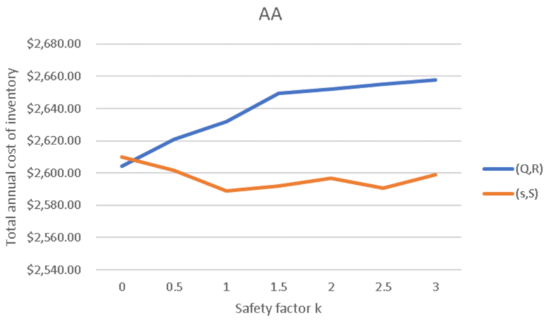

Figure 4. Inventory simulation cost vs safety factor k for item of classification AA

Source: Own elaboration.

We recommend using inventory (s, S) method for the macro category AA with a safety factor of one, that represents the greatest saving and better inventory service level. We note that the inventory cost (s, S) results are lower than the (Q, R) method. Except when k is equal to zero or the inventory model (Q, R) is evaluated without safety stock.

The items of the BB macro category are 9 SKU. Lower costs are observed with the (s, S) inventory model, the same as the AA category item, as can be seen in figure 5. We recommend using inventory (s, S) method for the macro category BB with a safety factor of 0.5, that represents the greatest saving and better inventory service level.

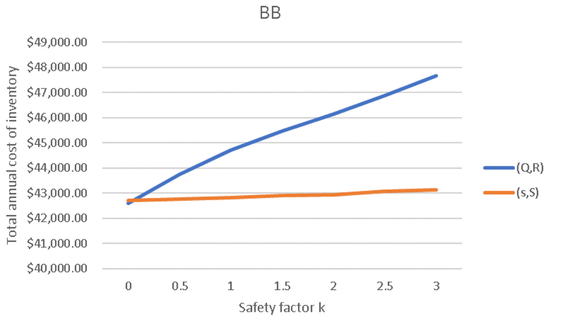

Figure 5. Inventory simulation cost vs safety factor k for item of classification BB

Source: Own elaboration

Finally, the remaining 16 items belong to the CC macro category. Figure 6 shows that for any value of k the cheapest cost is the inventory model (s, S). Purchasing staff could choose a model (s, S) without safety stock. It could ensure a lower inventory cost.

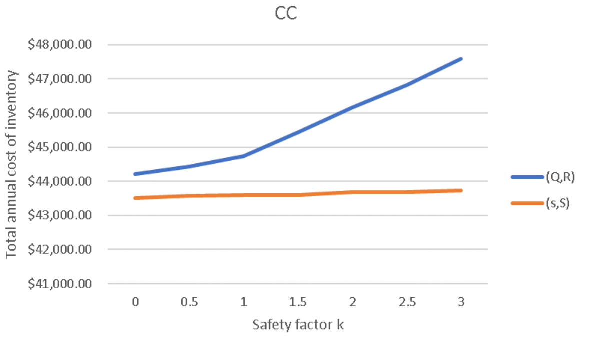

Figure 6. Inventory simulation cost vs safety factor k for item of classification CC

Source: Own elaboration.

5. Conclusions

There are several issues in MSMEs operation. We realize that data collection and simulation is not a common practice in some Mexican MSMEs. However, in this project the owner started a record of his purchases and sales, this allowed us to develop the simulation project and identify other areas for improvement. The lessons learned from the project are that the decision maker must begin to trust and make use of their own data analysis and scientific management.

The next steps to what is elaborated in this project is that the business continues with the data collection, and begins to collect from more areas in case the need to use them arises, the model presented in the text can be refined with more data and deliver more valid results, it could also be evaluated what changes are implied by modifying the time in which the T inventory is reviewed, as well as using other distributions for the problematic items, or all of them in general, such as the empirical one.

With this simulation project it was possible to identify the best policy to manage inventories. The double classification item made it possible to identify the products that generate utility from those that do not, as well as the products that give the company more utility. A free access tool was developed on the software Python for the management of inventory levels and safety stock of MSMEs. The application of simulation and double classification allowed to improve inventory management.

Acknowledgements

I want to thank Ricardo Torres, my teacher. For all the knowledge that he transmitted to me in class at the time and, more recently, for their valuable contribution to the realization of this article, with their constant supervision, advice, and comments.

Appendix A. Python code

```
#Python code
import numpy as np
import random
iterations=int(input("Number of iterations for each model: "))
n=0
Results=[]
data=np.loadtxt('data.csv', skiprows=1, delimiter=',')
#(R,Q)
while n<iterations:
    data=np.loadtxt('data.csv', skiprows=1, delimiter=',')
    k=(random.randrange(0,31,5))/10
    for i in range(0,26):
data[i][8]=round(data[i][8]+(k*np.sqrt(0.5)*data[i][6]),0)
        if data[i][8]<1:
            data[i][8]=1
    for i in range(0,26):
        for j in range(1,53):
data[i][11]=round(random.triangular(data[i][3], data[i][4], data[i][5]),0)
        if data[i][11]>data[i][10]:
            data[i][12]=data[i][12]+data[i][11]-data[i][10]
        data[i][10]=data[i][10]-data[i][11]
        if data[i][10]<0:
            data[i][10]=0
        data[i][15]=data[i][15]+data[i][10]
        if data[i][10]<data[i][8]:
            data[i][14]=data[i][14]+1
            data[i][13]=data[i][13]+data[i][7]
            data[i][10]=data[i][10]+data[i][7]
        j=j+1
    for i in range(0,26):
        Purchase_cost=data[i][1]*data[i][13]
        Ordering_cost=1284.484447*data[i][14]
Holding_cost=0.065495339*data[i][1]*(data[i][15]/52)
        Opportunity_cost=data[i][2]*data[i][12]
Inventory_Cost=Purchase_cost+Ordering_cost+Holding_cost+Opportunity_cost
        SKU=["(Q,R)",k,data[i][0],Inventory_Cost]
        Results.append(SKU)
    n=n+1
n=0
while n<iterations:
    data=np.loadtxt('data.csv', skiprows=1, delimiter=',')
    k=(random.randrange(0,31,5))/10
    for i in range(0,26):
data[i][8]=round(data[i][8]+(k*np.sqrt(0.5)*data[i][6]),0)
        if data[i][8]<1:
            data[i][8]=1
        data[i][9]=data[i][7]+data[i][8]
    for i in range(0,26):
        for j in range(1,53):
```

```
   data[i][11]=round(random.triangular(data[i][3],
data[i][4], data[i][5]),0)
       if data[i][11]>data[i][10]:
         data[i][12]=data[i][12]+data[i][11]-data[i][10]
       data[i][10]=data[i][10]-data[i][11]
       if data[i][10]<0:
         data[i][10]=0
       data[i][15]=data[i][15]+data[i][10]
       if data[i][10]<data[i][8]:
         data[i][14]=data[i][14]+1
         data[i][13]=data[i][13]+data[i][9]-data[i][10]
         data[i][10]=data[i][9]
       j=j+1
     for i in range(0,26):
       Purchase_cost=data[i][1]*data[i][13]
       Ordering_cost=1284.484447*data[i][14]
Holding_cost=0.065495339*data[i][1]*(data[i][15]/52)
       Opportunity_cost=data[i][2]*data[i][12]
Inventory_Cost=Purchase_cost+Ordering_cost+Holdi
ng_cost+Opportunity_cost
       SKU=["(s,S)",k,data[i][0],Inventory_Cost]
       Results.append(SKU)
     n=n+1
   np.savetxt('Results.txt',Results,fmt='%s')
#Python code end
```

References

Cavalieri, S., Garetti, M., Macchi M., and Pinto R. (2008) A decision-making framework for managing maintenance spare parts, Production Planning & Control: *The Management of Operations*, 19:4, 379-396

Chandra, C., and Kumar, S. (2001). Taxonomy of inventory policies for supply-chain effectiveness. *Int. J. Retail. Distrib. Manag.*, Vol 29, Num 4: 164-175

Gupta, M., Tiwari, S., and Jaggi, C.K. (2020). Retailer's ordering policies for time-varying deteriorating items with partial backlogging and permissible delay in payments in a two-warehouse environment. *Ann. Oper. Res.* 295: 139–161.

Hopp, W. J., and Spearman, M. L. (2011), Factory Physics. *Waveland Press*, 3: 720.

Jiménez, G. R. (2014). Planning system, inventory control and production control in a pharmaceutical group, (Bachelor's thesis). *National Autonomous University of Mexico*.

Liang, X., Ma, L., Wang, H., and Yan, H. (2016). Inventory Management with Alternative Delivery Times. *Springer*, 106.

Parekh, S., Lee, J. and Kozman, T. (2008). A decision support system for inventory management.

Sabri, E. H., Gupta, A. P., and Beitler, M. A. (2006). Purchase Order Management Best Practices: Process, Technology, and Change Management. *J. Ross Publishing*, 15: 248.

Sipper, D., and Bulfin, R. L. (1997). Planning, control and integration. *MHHE*

Wilson, R.H. (1934). A scientific routine for stock control. *Harv. Bus. Rev*, 13:116-128