Universitat Pompeu Fabra Barcelona

# #Case 1:
# Choosing ordering goods under uncertainty

Applied Machine Learning and Optimization
6th May 2022
Team 4

Marc Fernández
Fernando Camara
Carla Torruella

## 1. Executive summary

*Situation of the caserta warehouse*

The results for 2021 were underwhelming. During the year, the service level at Caserta was 75.72% (or 276 out of 365 days), leaving 24.38% of the days with stock outs (89 out of 365 days). Not being able to satisfy the demand on 89 days translates into a large opportunity cost of 4,463,278 kgs of demand not satisfied. Furthermore, the Caserta Warehouse had an average stock of 359,941 kgs, and a redundant stock –over 10 days[1] worth of daily demand– on 38.63% of the days (141 out of 365), resulting in an average redundant stock of 356,018 kgs of coffee beans and the same amount in stagnant money. These results imply that, for 63% of the year, the warehouse had either too much stalled money in the form of stock, or too little stock to satisfy the demand.

A deeper analysis of the purchasing policy shows no consistency. The orders were made seemingly at random, and oftentimes there would be clustered around a date, while there were extensive periods of time without orders at other points throughout the year. For instance, may saw only 2 orders

During the study, we are going to assume a minimum desired service level of 95% to present the best possible policy. Using this strategy will help Beanie Limited sell more products and increase customers' satisfaction.

Under no restrictions on minimum order quantity (MOQ), and a fluctuating lead time, our study suggests that the best policy is buying once the stock is lower than 401,194 kgs (8 days of demand), and buys just enough so that if the order was to arrive on that same day[2] the stock would get to 902,686 (18 days of demand). This would maintain an average stock level of 315,315 kgs (44,626 kgs less than in 2021) with a service level of 95,77%. It would also be potentially beneficial to buy just enough so that the order would be 802,387 minus the current stock, which would mean a stock level of 260081 kgs, but would not guarantee a 95% service level.

With a MOQ restriction, the best policy would be to buy every time the stock level is below 8 days of demand (same as under no restrictions), and the order quantity would be 500,000 (enough to cover roughly 10 days of demand). This would yield a service level of 96% and an average stock of 298,226 kgs.

Lastly, we do not recommend accepting the offer to have a guaranteed lead time of 15 days with an MOQ restriction, as the best policy under these assumptions would translate into an average stock of 382,953 kgs, with a service level of 96%.

[1] The 10 day mark satisfies 95% of Diemen0s lead times and is consistent with the results of the optimal policy for this situation (Level 1 in the analysis of the results)

[2] See section "purchase policy", under "Methodology", for a better understanding on the proposed formula

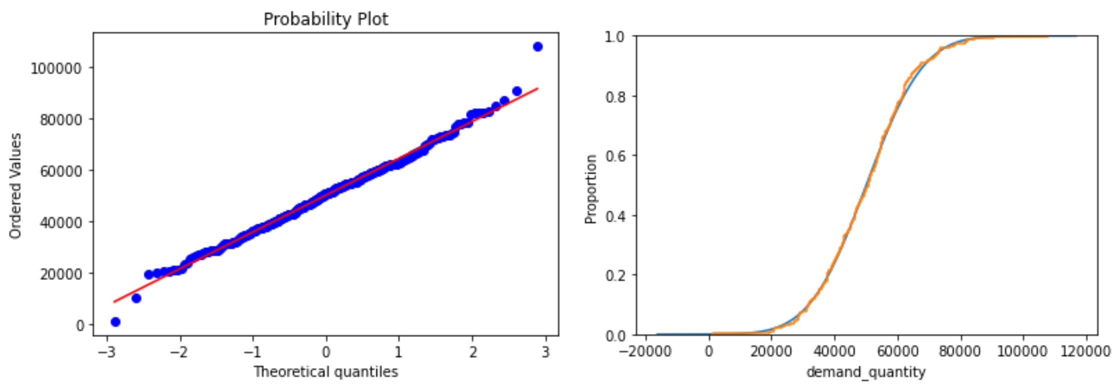
## 2. Methodology

*Simulation process*

In order to work out a solution for the problem at hand, we run several simulations where we modeled the elements that take part in the choice of a purchasing policy: (i) the demand, (ii) the lead time from Diemen to Caserta, (iii) and different values for when to buy and which quantities.

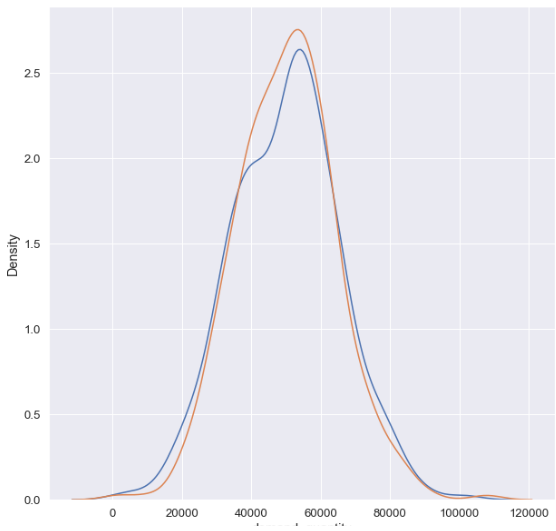
*Demand simulation*

In order to simulate the demand we did some exploratory statistics to gain insights on how it behaved. On a close inspection, we realized that the periods are not correlated, and no stationality is present (the distribution is, thus, iid); indeed it behaves similarly to a $N(50149.2,\ 14220.77)$, but it presents mesokurtic features (kurt 0.59), and slightly fat tailed to the right (skew 0.156). It is noticeable, however, that it is not a normal distribution at the 95% CI, but it is at the 99%. That happens probably because of the presence of the three outliers on the distribution.




QQplot and ECDF of a theoretical normal with the same Mean and SD as the observed distribution vs the empirical data.
QQplot: red is the theoretical quantiles and blue the observed values
ECDF: Light blue is the theoretical ECDF and orange is the empirical distribution

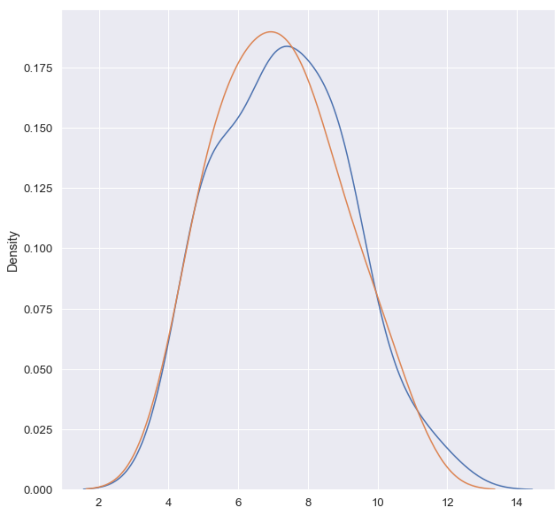
Given the outcome of this analysis, we decided to resample the empirical distribution, but forcing lower and upper limits to make sure no extreme values are present (for instance a negative demand); in this case, the lower bound is 1,000, and the upper bound is 11,000. The results of our demand generation *versus* the empirical data are quite similar:




| | Simulated | Empirical |
|---|---|---|
| **Count** | 365 | 365 |
| **Mean** | 49968.5 | 50149.2 |
| **Standard deviation** | 15295.05 | 14220.77 |
| **Min** | 2392.43 | 1380.99 |
| **25%** | 39358 | 40200.06 |
| **50%** | 51202.21 | 50873.13 |
| **75%** | 59796.67 | 59385.01 |
| **Max** | 101505.51 | 107790.97 |

*Lead time simulation*

Same as with the demand generation function, for the lead time generator we did some data exploration to see how the variable behaved. In this case we cannot reject the null hypothesis of normality using a Jarque-Bera test at any of the standard confidence levels (*p = 0.53)*. In this case, and because of the JB analysis, we decided to generate the data using a random normal distribution with the same mean and standard deviation as the empirical lead time ($N[7.3,\ 1.84]$) and restricted it to a lower and upper bound [4, 12], but this time we restricted the values to the nearest integer.




| | Simulated | Empirical |
|---|---|---|
| **Count** | 60 | 60 |
| **Mean** | 7.17 | 7.3 |
| **Standard deviation** | 1.79 | 1.84 |
| **Min** | 4 | 4 |
| **25%** | 6 | 6 |
| **50%** | 7 | 7 |
| **75%** | 8 | 9 |
| **Max** | 11 | 12 |

*Purchase Policy*

The fact that the data is not autocorrelated for any lags implies that the best forecast we can create is to use the sample unconditional mean. Because of this restriction, the best purchasing policy we can create doesn't take into account past values of demand to make an estimate for any number of periods ahead. On top of that, we assumed only one order can be made at a time, otherwise it could affect lead times at Diemen (more purchase orders translate into more workload, and possibly a bottleneck).

For the moving parameters of the purchase policy, we deemed best to estimate two values: a *trigger,* and a *goal.* The trigger refers to the point at which an order is made, while the *goal* makes reference to the amount of stock we would have at the end of the day if the lead time was 0. The reason why we set a goal, instead of a fixed value is quite simple: once the stock crosses the trigger point it ends up at some random value. The goal is simply a dynamic way of stating the amount to buy.

One last aspect to take into account is that we will be working in terms of *daily demand* instead of stock in absolute terms. This considerably simplifies the simulations, given that it provides more coherent steps to iterate over, and creates a better framework to think about the stock. Because the demand isn't autocorrelated at any lag, it is also consistent with the best forecast we can make (the unconditional mean). A unit of daily demand corresponds to 50,149 kgs of coffee beans.

Formula for the purchase policy amount once the stock crosses the trigger point:

$$Purchase\ amount\ =\ goal\ -\ stock_i$$

*where:*

$$stock_i:\ Caserta's\ stock\ on\ day\ i$$

When the MOQ rule is applied, the formula is adjusted to account for it.

$$Purchase\ amount\ =\ max(goal\ -\ stock_i,\ stock_i\ +\ 500,000)$$

*Simulation methodology*

In order to evaluate each stock policy and find the one that maximizes the company's results, we have run several simulations where we changed the daily demand for the trigger and goal variables, which we have repeated randomly for 500 periods of one year each. Doing this, we can present results with a high confidence of their accuracy. The iterations go from 6 to 11 days of demand for the trigger parameter, and from 11 to 19 for the goal, with a 1 day step up for both parameters and a minimum difference (goal - trigger) of 4 days, which is the minimum lead time (values below that are irrelevant because we know for sure they wouldn't arrive on time). In the case of 15 day lead time, the iterations were changed to 14-19, and to 25-30 respectively.

To calculate the initial stock that we have used for the simulation we remove the first 21 days to be able to start with a random and positive stock but at the same time consistent with our purchasing policy.

After each simulation we compute the mean stock and the service level of the 500 years simulated in order to get the two metrics that will help us make the decision of our choice of policy.

*Making the choice*

There are two main ways to approach the results of the simulations in order to make the best choice. The first one is to set a minimum service level and choosing the policy that holds the least stock while satisfying that restriction; and the second is to (again) set a minimum service level and choosing the policy that is the most cost effective, that is, the one that yields the lowest stock level per % service level. Although the distinction is slim, it is worth considering as the two ways of looking at the problem don't necessarily lead us to the same conclusion and ranking of policies. For instance, consider the case where we have choice A, with a service level of 95% and an average stock of 10,000; but choice B had a service level of 98% with an average stock of 10,100. If we were to minimize stock levels, we would choose option A, as it would be lower in *absolute terms*, but if we wanted to optimize the cost per % of service level, option B we'd be better off choosing option B (105 €/SL vs 103 €/SL).

Throughout the analysis, we will assume a minimum service level of 95%, which satisfies the demand on 346 out of 365 days, leaving 19 days where there are stock outs.

## 3. Analysis of the results

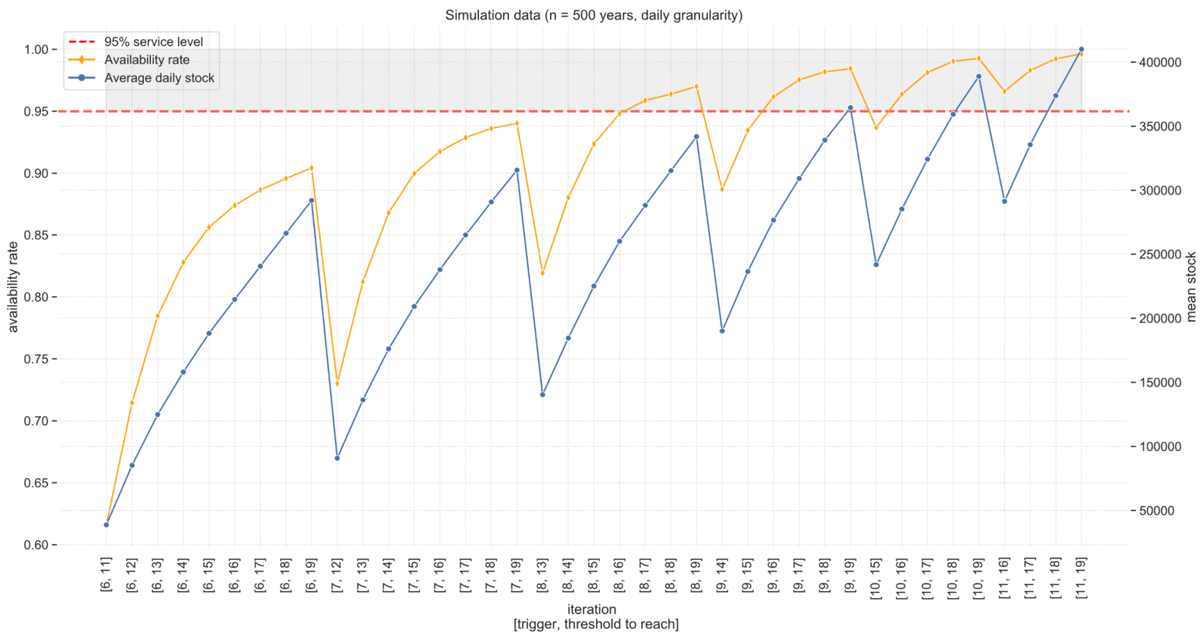
**LEVEL 1. Analyzing Lisa's situation**

*(See the executive summary)*

**LEVEL 2. Ordering policy**

The two main rationales explained in the methodology section are consistent in this scenario –The policy choice that has the lowest absolute cost is also the most cost effective–. The chosen policy choice is to set a trigger on 8 days of demand, and a goal of 18 days. That translates into buying the day the stock dips below 401,194 kgs, with a goal of 902,686 kgs.

This policy would yield a service level around 95.87%. Note that, because of small deviations, the *observed* service level on a given year could be slightly higher, but also slightly lower –this is precisely the reason why the graph below differs slightly to the data on the appendix–. The policy suggested is best to ensure a minimum service level of 95% is guaranteed, although a policy could be as aggressive as a [8, 16] (trigger, goal), although it would not *guarantee* a service level above 95%, it is an observed value in some simulations, and it would save up around 55,234 € in stock.

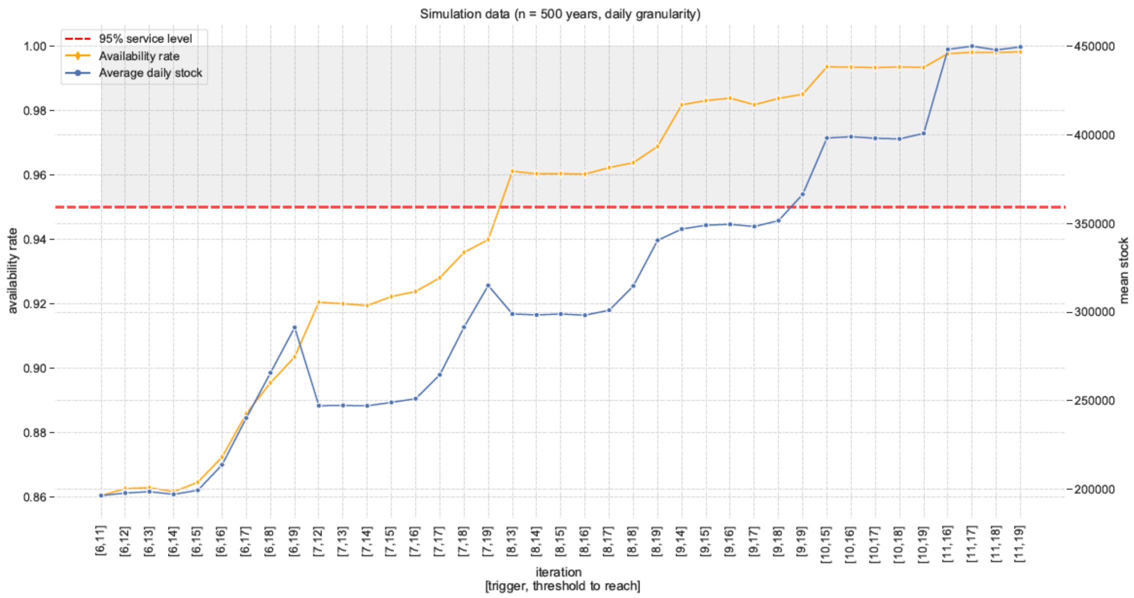



**LEVEL 3. Minimum Order Quantity of 500,000 kgs of beans.**

In this paragraph we have a new restriction, a Minimum Order Quantity (MOQ) of 500,000 kgs of beans. This restriction affects the previously proposed policy.

The new policy we propose is ordering when there's 401,194 kgs of beans left, this is, stock to supply 8 days; and the goal will be 802,387 kgs, a stock to supply for 16 days. Same as with the first case, where no MOQ rule was present, the two decision criteria are consistent with each other.

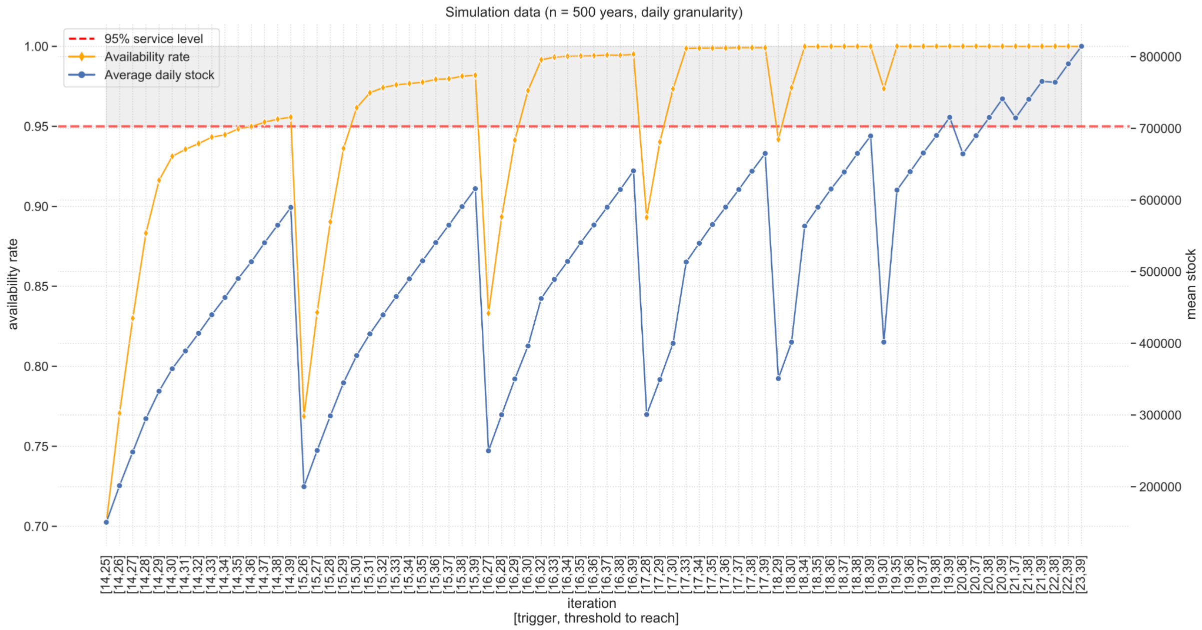
Although we are suggesting [8,16] (trigger, goal), other equally efficient intervals will be those from [8,13] to [8,16]. The reason for the interval to be equally as good is that a MOQ of 500k translates into roughly 10 days of demand, so whenever goal minus trigger is less than 10 days, the MOQ rule forces it to buy the equivalent of 10 days of stock, thus in reality we are actually observing the same data and the same purchase amounts. It's also worth noticing that these intervals are not completely flat, but small deviations are present. This is, again, because of the randomness of the lead time and the demand, but seeing how small these variations are hints us towards how thin the confidence interval would be (thanks to having a sample size of 500 years).




**LEVEL 4. 7 days target lead- time or 15- day lead time**

In this section, Caserta is offered a choice between a lead time of exactly 15 days with 100% certainty or the previous scenario with a target lead time of 7 days. Here the MOQ rule mentioned before still applies.

After running the simulation, the optimum order policy is at the combination [15, 30], meaning, we have to place an order when we have 752,238 units of stock left, translated to a demand enough to cover for 15 days. Our goal is a quantity of 1,504,476 units, which translates to a demand of 30 days.


Simulation data (n = 500 years, daily granularity)
95% service level
Availability rate
Average daily stock
availability rate
1.00
0.95
0.90
0.85
0.80
0.75
0.70
mean stock
800000
700000
600000
500000
400000
300000
200000
iteration
[trigger, threshold to reach]

Universitat Pompeu Fabra Barcelona

## First level

| TrackingIteration_ | Availability_rate | Mean_stock | TrackingIteration_ | Availability_rate | Mean_stock |
|---|---|---|---|---|---|
| **[6, 11]** | 0,383972255 | 38805,75186 | **[8, 16]** | 0,92370193 | 260081,4578 |
| **[6, 12]** | 0,616027745 | 85191,1281 | **[8, 17]** | 0,948126517 | 288287,1967 |
| **[6, 13]** | 0,714487648 | 124900,8977 | **[8, 18]** | 0,958771867 | 315315,6657 |
| **[6, 14]** | 0,784643959 | 158059,7901 | **[8, 19]** | 0,963839777 | 341917,8709 |
| **[6, 15]** | 0,827871861 | 188217,0286 | **[9, 14]** | 0,970030846 | 190044,8317 |
| **[6, 16]** | 0,856405564 | 214661,3277 | **[9, 15]** | 0,88692808 | 236479,8533 |
| **[6, 17]** | 0,874036412 | 240650,0557 | **[9, 16]** | 0,934615743 | 276695,7105 |
| **[6, 18]** | 0,886484295 | 266295,5306 | **[9, 17]** | 0,961675643 | 309192,0221 |
| **[6, 19]** | 0,895754461 | 292098,8428 | **[9, 18]** | 0,975361739 | 339164,4854 |
| **[7, 12]** | 0,904186368 | 90801,17487 | **[9, 19]** | 0,981793876 | 364502,0883 |
| **[7, 13]** | 0,730222824 | 136327,5516 | **[10, 15]** | 0,984297697 | 241795,8342 |
| **[7, 14]** | 0,812344881 | 176074,8922 | **[10, 16]** | 0,93686206 | 285310,8791 |
| **[7, 15]** | 0,867976836 | 209184,5542 | **[10, 17]** | 0,96368637 | 324302,1849 |
| **[7, 16]** | 0,899847141 | 237914,3281 | **[10, 18]** | 0,981317218 | 359319,069 |
| **[7, 17]** | 0,917527298 | 265017,8752 | **[10, 19]** | 0,990269613 | 388916,4888 |
| **[7, 18]** | 0,92876984 | 290883,8346 | **[11, 16]** | 0,992663858 | 291380,1208 |
| **[7, 19]** | 0,936105982 | 315786,6689 | **[11, 17]** | 0,966047742 | 335605,8387 |
| **[8, 13]** | 0,940280844 | 140348,315 | **[11, 18]** | 0,982977301 | 373868,7189 |
| **[8, 14]** | 0,819335857 | 184477,6008 | **[11, 19]** | 0,992313213 | 410027,603 |
| **[8, 15]** | 0,880348015 | 225054,3508 | | | |

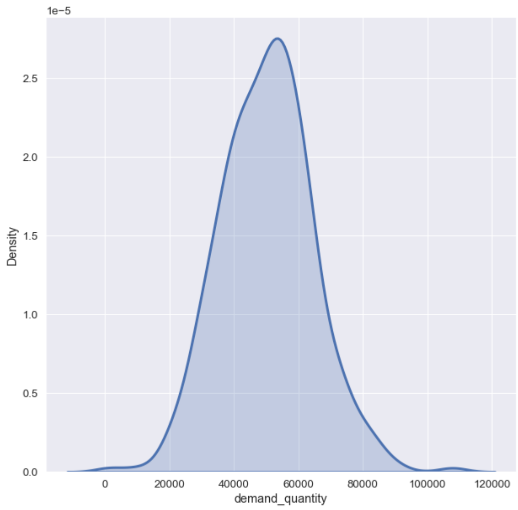
## Second level

| TrackingIteration | Availability_rate | Mean_stock | TrackingIteration_ | Availability_rate | Mean_stock |
|---|---|---|---|---|---|
| **[6,11]** | 0.8603831887837564 | 196517.26449649647 | **[8,16]** | 0.9602401915396037 | 298226.15356651036 |
| **[6,12]** | 0.8625747174297752 | 197953.52381186924 | **[8,17]** | 0.9622728343587861 | 301054.1968380475 |
| **[6,13]** | 0.8628431796889124 | 198733.95685874863 | **[8,18]** | 0.9637959467677691 | 314717.2346794477 |
| **[6,14]** | 0.8615994871822968 | 197220.42952754613 | **[8,19]** | 0.9688309838319974 | 340466.699468793 |
| **[6,15]** | 0.8645196991031169 | 199517.56745661725 | **[9,14]** | 0.9817610028435084 | 346795.8656864502 |
| **[6,16]** | 0.8723653716558643 | 213900.14299664737 | **[9,15]** | 0.9830868776743498 | 348980.622859357 |
| **[6,17]** | 0.8857830057911145 | 240245.9850298284 | **[9,16]** | 0.9838593915220715 | 349475.99485966103 |
| **[6,18]** | 0.8954202530119821 | 265755.042986297 | **[9,17]** | 0.9817719604867385 | 348237.35133667756 |
| **[6,19]** | 0.9034631631428712 | 291318.8121975601 | **[9,18]** | 0.9837662515546156 | 351566.6728876669 |
| **[7,12]** | 0.9204584677927471 | 247224.3762283075 | **[9,19]** | 0.9850537746341517 | 366431.8528605738 |
| **[7,13]** | 0.919981810312238 | 247348.14103409118 | **[10,15]** | 0.9935788210671649 | 398134.95850319264 |
| **[7,14]** | 0.9194174916858882 | 247136.09233850503 | **[10,16]** | 0.9934637658132489 | 398886.1028870583 |
| **[7,15]** | 0.922195254244717 | 249070.4574511378 | **[10,17]** | 0.9933322740944878 | 397989.31213667756 |
| **[7,16]** | 0.9237895913346957 | 251149.14053637336 | **[10,18]** | 0.9935349904942445 | 397620.953068489 |
| **[7,17]** | 0.9280521145512023 | 264641.1342901023 | **[10,19]** | 0.9933815834890232 | 400747.4724715327 |
| **[7,18]** | 0.935925181212025 | 291488.2029041052 | **[11,16]** | 0.9976769796352201 | 448099.0086185652 |
| **[7,19]** | 0.9399028057045491 | 315007.12396194367 | **[11,17]** | 0.9980769336131184 | 449946.5528380473 |
| **[8,13]** | 0.9611332394628563 | 298945.9148806666 | **[11,18]** | 0.9980111877537379 | 447816.6987506808 |
| **[8,14]** | 0.9603497679719046 | 298387.78089740913 | **[11,19]** | 0.9982303406183398 | 449508.3704660532 |
| **[8,15]** | 0.9603881197232099 | 298922.5321031926 | | | |

## Third level

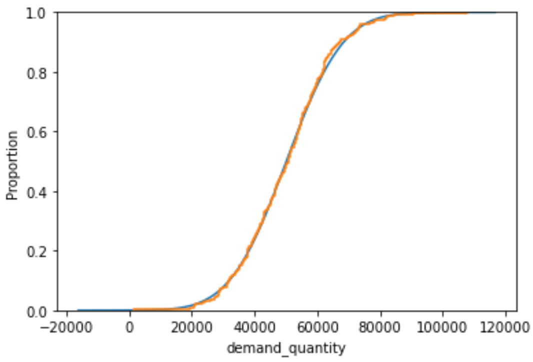
| iteration_tracking | availability_rate | mean_stock | iteration_tracking | availability_rate | mean_stock |
|---|---|---|---|---|---|
| **[14,25]** | 0,702549296 | 150444,7273 | **[16,36]** | 0,994121224 | 565332,1935 |
| **[14,26]** | 0,770623654 | 201488,3788 | **[16,37]** | 0,994554051 | 590058,8614 |
| **[14,27]** | 0,829959292 | 248357,9288 | **[16,38]** | 0,994362293 | 614686,3387 |
| **[14,28]** | 0,883142214 | 294924,7948 | **[16,39]** | 0,995047145 | 640800,0396 |
| **[14,29]** | 0,916305521 | 333317,2315 | **[17,28]** | 0,893102711 | 300670,4087 |
| **[14,30]** | 0,931284619 | 364510,715 | **[17,29]** | 0,940258929 | 349411,5002 |
| **[14,31]** | 0,935678634 | 389313,5675 | **[17,30]** | 0,973477025 | 399840,7685 |
| **[14,32]** | 0,939267262 | 413951,368 | **[17,33]** | 0,998685083 | 513331,6277 |
| **[14,33]** | 0,943283239 | 439741,7874 | **[17,34]** | 0,998833011 | 539508,8748 |
| **[14,34]** | 0,94476252 | 463873,4689 | **[17,35]** | 0,998860405 | 565845,4735 |
| **[14,35]** | 0,948466204 | 490262,8999 | **[17,36]** | 0,998926151 | 590205,3067 |
| **[14,36]** | 0,949803036 | 513672,5135 | **[17,37]** | 0,999167219 | 614690,485 |
| **[14,37]** | 0,95266846 | 540124,7448 | **[17,38]** | 0,999150783 | 640336,3615 |
| **[14,38]** | 0,954476471 | 565059,4073 | **[17,39]** | 0,999106952 | 665016,9374 |
| **[14,39]** | 0,955747558 | 589776,6289 | **[18,29]** | 0,941809436 | 350817,2997 |
| **[15,26]** | 0,76883208 | 200080,7497 | **[18,30]** | 0,974189271 | 401532,5437 |
| **[15,27]** | 0,833646539 | 250546,1844 | **[18,34]** | 0,999835635 | 563679,9293 |
| **[15,28]** | 0,890335907 | 298738,7947 | **[18,35]** | 0,999846593 | 590080,8457 |
| **[15,29]** | 0,936221038 | 344948,0185 | **[18,36]** | 0,999912339 | 615549,8951 |
| **[15,30]** | 0,961631812 | 382952,9983 | **[18,37]** | 0,999923296 | 639210,2979 |
| **[15,31]** | 0,970945809 | 413094,0958 | **[18,38]** | 0,999912339 | 665094,0809 |
| **[15,32]** | 0,974211187 | 439611,7031 | **[18,39]** | 0,999923296 | 689407,9243 |
| **[15,33]** | 0,975931537 | 465245,1664 | **[19,30]** | 0,973564686 | 401613,1012 |
| **[15,34]** | 0,976736923 | 489800,4686 | **[19,35]** | 0,999994521 | 613917,6328 |
| **[15,35]** | 0,977597098 | 515021,0815 | **[19,36]** | 1 | 639570,9466 |
| **[15,36]** | 0,979361279 | 540431,0414 | **[19,37]** | 0,999994521 | 665724,5967 |
| **[15,37]** | 0,979733839 | 565149,0678 | **[19,38]** | 0,999983564 | 690121,0632 |
| **[15,38]** | 0,98135557 | 590924,0877 | **[19,39]** | 0,999994521 | 715375,488 |
| **[15,39]** | 0,981919889 | 615869,2424 | **[20,36]** | 1 | 664471,1711 |
| **[16,27]** | 0,833054827 | 250068,2794 | **[20,37]** | 1 | 689820,5956 |
| **[16,28]** | 0,893447877 | 300400,378 | **[20,38]** | 1 | 715200,359 |
| **[16,29]** | 0,941431397 | 350262,5962 | **[20,39]** | 1 | 741217,0197 |
| **[16,30]** | 0,972375781 | 396294,8271 | **[21,37]** | 1 | 714629,1159 |
| **[16,32]** | 0,991568094 | 462463,4583 | **[21,38]** | 1 | 740497,9239 |
| **[16,33]** | 0,993217219 | 489230,7713 | **[21,39]** | 1 | 765461,5712 |
| **[16,34]** | 0,993765101 | 514158,5835 | **[22,38]** | 1 | 764298,1311 |
| **[16,35]** | 0,993945902 | 540315,4522 | **[22,39]** | 1 | 790050,118 |

In this paragraph, we will show some metrics that will explain Lisa's situation. Lisa made a total of 60 product orders. The lead time average is 7.3 days and 95% of the product orders were served within the first 10 days. Lisa had a negative stock on 24.4% of the days, this is, a service level of 75.6%. The demand had an average of 50,149 kgs/day and a standard deviation of 14,220 kgs/day.
In this histogram we can see the number of days in which the demand is between certain values throughout the year.

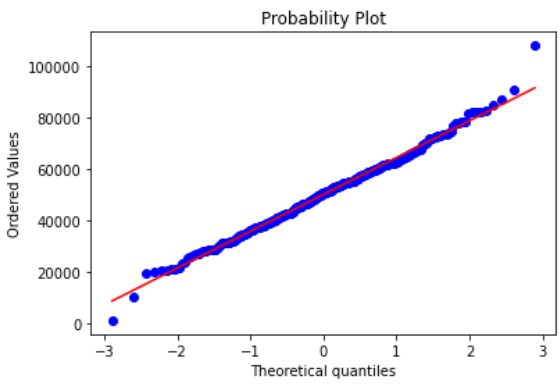



As it might be seen is a graph that approximates a normal distribution.
Different metrics have been performed to see if we can treat it as a normal distribution.

- The kurtosis is 0.59.
- The skew is 0.156
- The jarque bera has a p-value of 0.03. With a confidence level of 95% we reject the null hypothesis that this is a normal distribution but with a 99% of level we don’t reject the null hypothesis and we assume that's a normal distribution.
- The orange is empirical data and the blue is a theoretical normal data with the same mean and standard deviation as the empirical distribution.




- It is not a normal distribution, the kurtosis is too low. But we will assume it behaves as one because it's fairly close. So, we assume it is a normal distribution.




We performed a CI for a normal distribution at 95% confidence and obtained the following interval: (48690 , 51608). There's about 0.4 days (in average) between the bottom and top brackets, so using the mean will give us a fairly good approximation.

In the following graph, we can see the amount of stock in relation to the days of the year.

A green line has been drawn at 10 and a red line at 0. All stock below 0 is customers who want to buy but cannot. All stock above 10 is surplus stock.

The 10 day cutoff satisfies 95% of the POs served. This is why we think that all stock in excess of the 10-day demand is unnecessary.

On average, the amount in $ of overstock for the days that was stock surplus is 356,018$ and the amount in $ for the days that there were negative stock is 235,185$.