IMPROVING CUSTOMER WAITING TIME FOR MEDICINE-RETRIEVAL CENTER

Mateo Restrepo Sierra

Mathematical Engineering
Universidad EAFIT
Medellín, Colombia

Juan Sebastián Cárdenas-Rodríguez

Mathematical Engineering
Universidad EAFIT
Medellín, Colombia

David Plazas Escudero

Mathematical Engineering
Universidad EAFIT
Medellín, Colombia

ABSTRACT

Neuromédica is a Colombian pharmacy which provides treatment for people with neurological diseases. Recently, Neuromédica started attending patients from other pharmacy which led to a significant increase in the waiting time. In this pharmacy, people are classified and attended due to certain priorities. The data, given by Neuromédica, is analyzed using boxplots, Kruskal-Wallis and Kolmogorov-Smirnov tests with Python's library Scipy. The objective of this work is to determine the number of assistants and queue logistic such that the waiting time has a significant reduction, with the purpose to provide a satisfactory level of service. A discrete-event simulation model was created and implemented in Python. A heuristic approach to minimize the waiting time is used. Additionally, a sensitivity analysis is made on the assumed distributions.

Keywords: Complex system modeling, queue theory, discrete-event simulation, simulation-optimization,statistical analysis of input-output data, white and black-box validation, conceptual modeling and python.

1 INTRODUCTION

Health care is one of the few systems that involve the great majority of the population, therefore there is an increasing demand on all the provided services by health institutions. This is the case of the system analyzed in this work: medication retrieval. These kinds of systems have complex interactions and often evolve into long queues with large waiting times, which creates dissatisfaction among the patients. Thus, health institutions have to face this issue and strive for solutions.

Among the different approaches to treat this problem, Discrete-Event Simulation (DES) emerges as a possible solution. Different authors, such as (Thorwarth and Arisha 2009) and (Jun, Jacobson, and Swisher 1999), provide a review of DES applied to health care. For the specific case of pharmacies, (Reynolds, Vasilakis, McLeod, Barber, Mounsey, Newton, Jacklin, and Franklin 2011) developed a DES model to analyze the automatization of some processes. More references can be found in the references within the mentioned work. Some more references in applications of queue theory in health care (and some in pharmacy) can be found in the excellent review provided by (Lakshmi and Iyer 2013).

Following the ideas of the above-presented authors, we propose a DES model for the medication retrieval process in the Colombian context, focusing on queue logistics. The main justification for this methodology is that it provides a tool for experimentation without having to use the real system, it takes less time and does not bring risks (Reynolds, Vasilakis, McLeod, Barber, Mounsey, Newton, Jacklin, and Franklin 2011).

2 CONCEPTUAL MODEL

2.1 Problem Definition

Neuromédica is a health services institution (IPS for its abbreviation in spanish), specialized in health care for adult population focused on neurological disorders; this is a specialized center for Grupo SURA. Their Almacentro complex is in charge of the distribution of medicine for their patients. Recently, they started receiving patients from DEMPOS (another medicine supplier in Colombia) and they are worried about the significant increase in the waiting time of their costumers.

In this complex, patients arrive and take a turn accordingly to their service type and priority, then they enter a queue until they are called into a specific counter; generally, they are called once and the medicine is provided instantly, but sometimes they are recalled a second time. In total, there are 15 counters. Each operator has some service types assigned with certain priorities, but they can also call patients outside these rules. It is important to remark that a patient enters the system exactly when he takes the turn.

In Figure 1, the general activity scheme is presented; it depicts the flow of a patient throughout the system; and in Figure 2 we present a minimalist representation of the system at hand.

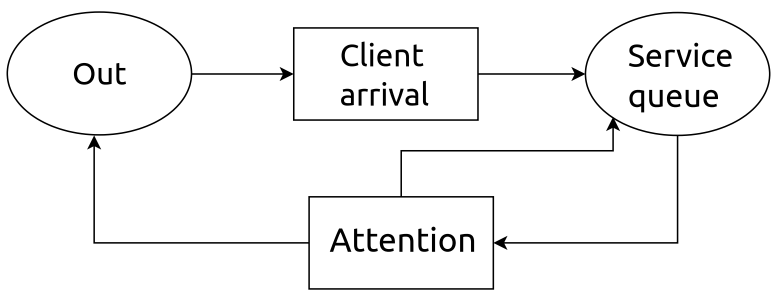


Figure 1: Activity Cycle Diagram

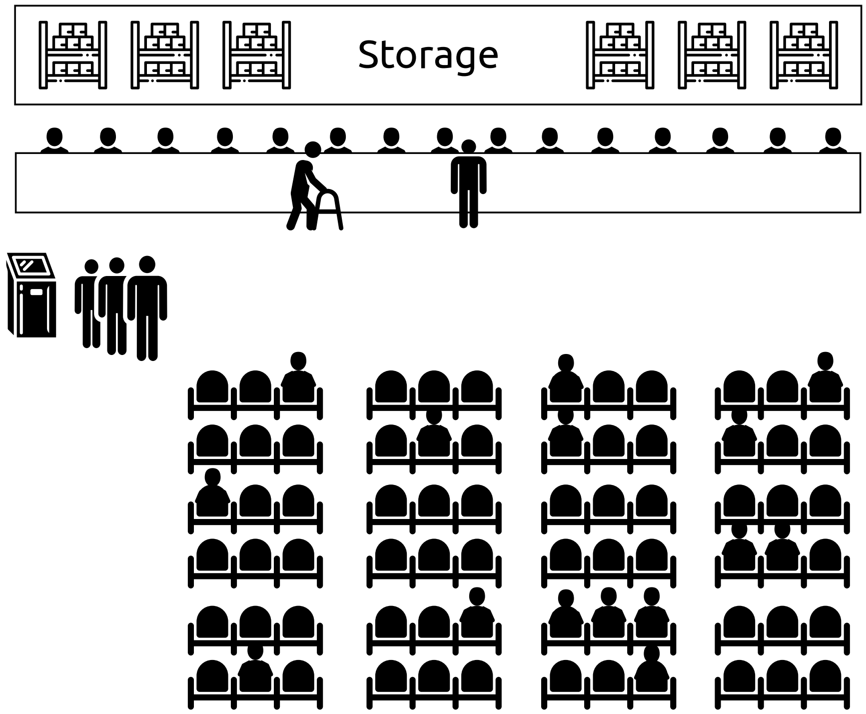


Figure 2: System diagram.

The complex attends the services presented in Table 1, each service has associated an ID; on the other hand, each module attends specific IDs with certain priorities, as shown in Table 2. For example, “Admissions 1” attends, first, service 4 which represents “Appointment confirmation & PAC, MP and policy SURA”; then it attends service 5 “Procedure after appointment”; and finally, it calls service 9 “Preferential appointment confirmation”. It is important to mention that each module can attend outside this policies, for instance, if there are not patients waiting in line with ID 4, 5 or 9, “Admissions 1” can call patients with other IDs. Moreover, the attendants can voluntarily call, at any moment, any patient waiting in line.

ID	**Name**
3	General Pharmacy
4	Appointment confirmation & PAC, MP and policy SURA
5	Procedure after appointment
7	Preferential Pharmacy
8	Pharmacy PAC, MP y SURA policy
9	Preferential Appointment confirmation
10	Scheduled retrieval

Table 1: Types of services

Module	IDs
Admissions 1	4,5,9
Admissions 2	4,9,5
Pharmacy 1	3,7,8,10
Pharmacy 2	3,8,7,10
Pharmacy 3	7,3,8
Pharmacy 4	7,3,8
Pharmacy 5	3,7,8,10
Pharmacy 6	3,8,7,10
Pharmacy 7	3,7,8,10
Pharmacy Retrieval	10
Pharmacy 9	3,8,7,10
Pharmacy 10	3,7,8,10
Pharmacy 11	3,7,8,10
Pharmacy 12	3,7,8,10
Pharmacy 13	3,8,7,10
Pharmacy 14	3,7,8,10

Table 2: Policy for each module

2.2 Modelling Objectives

2.2.1 Model Purpose

To determine the priorities of each module in order to reduce the queue time of patients.

2.2.2 Specific Objectives

- To select the most appropriate data to be considered in the model.
- To apply statistical tests to filtered data in order to contrast different hypothesis.
- To validate and verify the implemented model using the available data.
- To explore multiple feasible strategies for queue priorities in the modules.

2.2.3 General Project Objectives

- *Time-Scale*: a final report must be available in 5 weeks.
- *Flexibility*: high, since our implementation in Python allows us to make modifications easily.
- *Ease-of-use*: no, due to its implementation in Python.

We decided to implement the model in Python since it allows us to develop a more realistic model, fitting the available data to a wider set of distributions and obtaining a better fit. On the other hand, the Python model is more adaptative for our further experimentation and optimization; although, this kind of implementation makes the model not usable by everyone. Moreover, Python is open-source.

2.3 Model Inputs & Outputs

2.3.1 Experimental Factors/Model Inputs

- Number of modules with priority by hours.
- Queue logistics.
- Type of service.

2.3.2 Outputs

To determine achievement of objectives:

- Waiting time for each type of patient.
- Average time in system.
- Reduce the percentage of patients waiting.

To determine reasons for failure to meet objectives:

- Average number of patients waiting by service type per hour.
- Percentage of people waiting is not reduce.
- Leisure time.

2.3.3 Parameters

- Inter-arrival times for each type of patient and hour of the day and day of the week.
- Duration of the attention by type of service and attendant.
- Probability of quitting.
- Type of services.
- Inter-service times for each attendant.

2.4 Model Content

2.4.1 Model Scope

Component	Include/Exclude	Justification
Entities:		
Customer of each type	Include	This entity is the main character of the simulation.
Attendants	Include	This entity is the one that interact with the costumers and its behavior determines the waiting time.
Medicine retrieval attendants	Exclude	The modelling and simulation of this work assumes that these personnel is always available and the dispatch process is out of the scope of this work.
Medicine authorization attendants	Exclude	Just as the previous entity, the medicine authorization process is out of the scope of this work.
Activities:		
Medicine retrieval	Include	This is the activity that each customer is going to be involved.
Medicine authorization	Exclude	There is no way two merge both systems.
Queues:		
Dispatch queue	Include	Jointly with the last queue, is the focus of this work.
Authorization queue	Exclude	Separate system.
Parking lot queue	Exclude	Separate system.
Delivery services	Exclude	Separate system.
Resources:		
Attendants for dispatch	Exclude	Number of people that dispatch the medicine.
Medicine	Exclude	We assume that there is always available medicine.

Table 3: Model scope.

2.4.2 Model Level of Detail

Component	Detail	Include/Exclude	Justification
Entities:			
Customer of each type	Customer inter-arrival time varying by hours of the day.	Include	The system at hand clearly has different arrival times for each hour on a day, allowing us to make a more realistic simulation.
	Customer inter-arrival time varying by days of the week.	Include	It is important to focus on the busiest days in order to make a more relevant study. It is important to focus on the busiest days in order to make a more relevant study.
	Customer inter-arrival time varying by minutes on each hour.	Exclude	We believe that considering the inter-arrival time by hours is enough for making significant simulation. Modelling the system by minutes is considered unmanageable.
Activities:			
Medicine Dispatch	Medicine dispatch duration	Include	It has a big impact on the waiting time of the customers.
	Cost per working hours of this attendants	Exclude	The company did not provide this information and it's not relevant for this work.
Queues:			
Authorization and dispatch queue	Customer desertion	Include	Modelled as a constant probability. It is important since this clears a spot in the queue.
	Different types of queues for each type of customers	Include	It is a feasible factor to control.
	Queue maximum capacity	Include	Customers will leave if capacity is exceeded.
	Skippers	Exclude	We are aware of these cases, but this will not be included in the model.
Resources			
Dispatch attendants	Number of attendants per hour	Include	It is considered per hour for future experimentation purposes.
Authorization	Number of people that authorized the medicine	Exclude	As the authorization process is out of the scope of this work, the number of people that authorized the medicine.

Table 4: Model level of detail

2.4.3 Assumptions

- There will always be medicine available.
- There are no skippers.
- Attendants can call other priorities if they are available.
- Attendants cannot voluntarily call patients.

2.4.4 Simplifications

- The patients will not be called a second time, only once.
- Quitters are treated as a single constant probability.
- As we will discuss later in Section 7, we only consider the data provided for the last week of August.

3 BACKGROUND

In the research done by (Shimshak, Damico, and Burden 1981), they propose several queueing models to evaluate the effectiveness of them and improve waiting times, considering the theoretical background discussed in (Cobham 1954), where he claims the queue models can be classified in two types: single-channel and multiple-channel systems. It is important to remark that these results constitute a theoretical and preliminary background in the queue theory with priorities.

Returning to the ideas proposed in (Shimshak, Damico, and Burden 1981), they use three different queueing schemes:

1. Multiple server model queue with no priorities.
2. Priority discipline queueing model with two priorities.
3. Priority discipline queueing model with preemptive service.

The main idea of said work is to analyze different the described alternatives to determine the effective work distribution of New Jersey Hospital pharmacy. Finally, a more recent application was introduced by (Ndukwe, Omale, and Opanuga 2011), where they apply queueing disciplines to improve the waiting times of outpatients in a Nigerian hospital, reducing the average waiting time from 167min to only 55.1min.

4 MODEL DATA

The data was provided directly by Neuromédica, it covers all records of August 2019. The data includes for each patient:

- Date.
- The arrival time.
- Waiting time.
- Time of service's start.
- Service duration.
- Time of service's end.
- Attendant.
- Counter.
- Type of service.
- Completion.

Furthermore, they provided another document that contains the types of services with their ID and their corresponding counters (see Tables 1 and 2). The names of the different attendants including their assigned services were provided as well.

It is important to highlight that the patient's ID was not granted by Neuromédica for security reasons; thus, it was not possible to extend the system with SURA's authorization procedure.

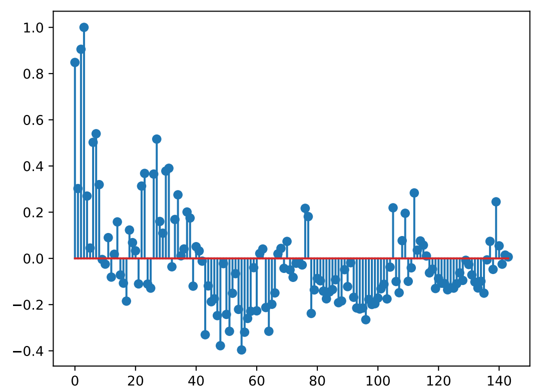
All data in this work is handled in minutes.

4.1 Arrivals

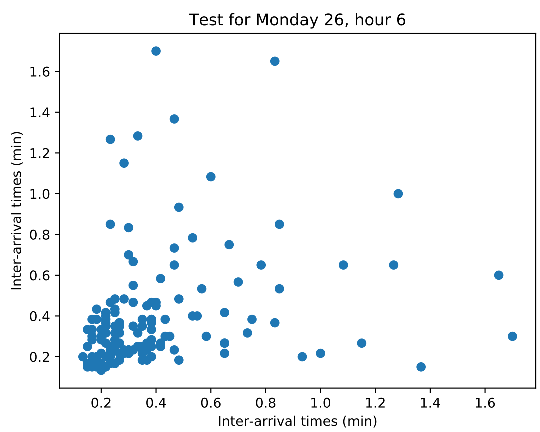
Using the dataset available, the inter-arrival times were initially calculated and then filtered for each day of the week by hours. The resulting data were grouped by days of the week, for instance, we separated the data for each Monday of August at 6 am, 7 am, etc.

As we will discuss later in Section 7, the model did not pass the validation process and we decided to reduce the data to one week: last week of August, since it is the only complete week of the month, due to Holidays and different schedules in the Complex. This decision reduces the used data from 27k to 6.7k. Therefore all the results and figures that are going to be presented use this amount of data.

In Figures 3a and 3b, we present the autocorrelation and scatter plot respectively. These plots show that there's no significant autocorrelation in the data since the autocorrelation plot is not strictly above the x-axis and the scatter plot shows no obvious behavior.



(a) Autocorrelation plot inter-arrival times.



(b) Scatter plot for inter-arrival times.

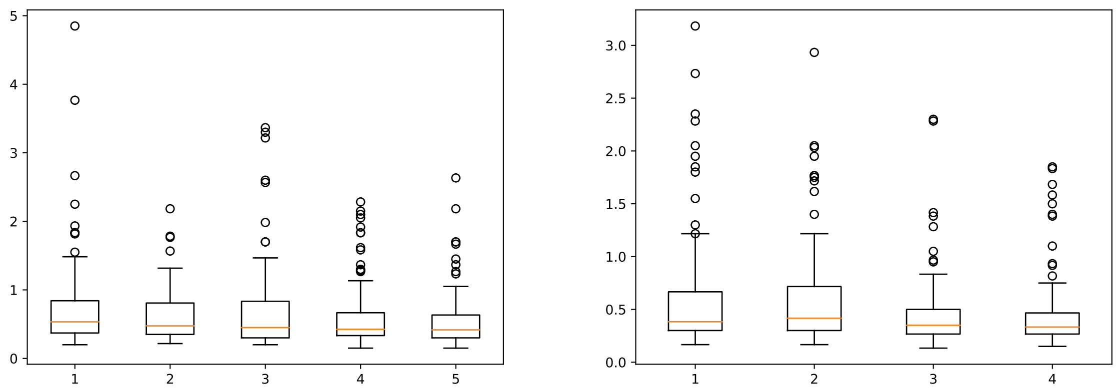
Figure 3: Data for Monday 26 at 6 a.m.

For the implementation, the distributions of the inter-arrival times are required. Therefore, for each day of the week and each hour of the day, it was checked to see if they come from the same distribution, hence a homogeneity test, e.g. for each Monday of the month, at 6 am, we compared if the inter-arrival times come from the same distribution. The homogeneity was tested using a Kruskal-Wallis test (Wackerly, Muñoz, Humbertotr, et al. 2010, pp. 765-767).

In the same way, a Kolmogorov-Smirnov test (Banks, J., B.L., and D. 2005, pp. 230-231) was applied to determine the best distribution that fits the inter-arrival time data, for each hour in the month. The test was performed with all the available theoretical distributions available in the stats module in the SciPy package.

In the case of the rejection of homogeneity by the Kruskal-Wallis test, we selected the distribution using boxplots, selecting the one that better characterizes the general behavior. In the other case, we selected the best fit distribution for the first day. See Appendix A for the detailed fitting.

For instance, we present the box plots of the data for Thursdays at 7 a.m. in Figure 4a. The Kruskal-Wallis test for this data concluded that there is not sufficient evidence to reject the null hypothesis, i.e. the data comes from the same distribution. Therefore the data can be considered homogeneous. The best-fit distribution is shown in Figure 4b.



(a) Homogeneous (Thursdays at 7 a.m.). (b) Non-homogeneous (Thursdays at 6 a.m.).

Figure 4: Boxplots for inter-arrival times.

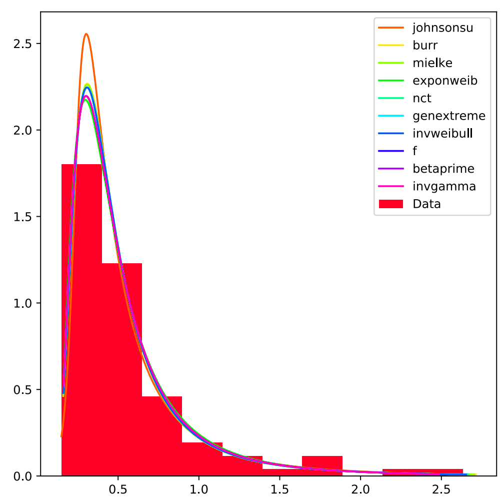


Figure 5: Distribution fitting for Thursdays at 7 a.m.

5 IMPLEMENTATION

We implemented the simulation model using the programming language Python. Python was chosen over a simulation software as this problem does not have many components but each of these components has many backend behaviors that are not shown, hence simulation software such as Simul8 would not be as useful. Furthermore, Python is open-source hence it could be used and tested by anyone who needed it.

For this implementation, we did not use a simulation package, instead, we programmed each of the components such as queues, event handling and more. We decided to take this approach, instead of using a simulation library as SimPy, for three main reasons. Firstly, to manage all the priorities that each module has and use it accordingly. Secondly, this approach is easy to run several times the simulation without using multi-processing packages. Lastly, so we can have more control over the system and the debugging process.

This implementation is easily modified, as it is object-oriented. Hence, changing behavior of any of the components would be done by changing the methods of each object and obtaining in this manner the desired change. Furthermore, visualizing the simulation can be done through outputs in the terminal but, a graphical user interface was not implemented.

```
luisinavasquez  finished attending Customer 378 at 9:01.
Customer 405 of type Farmacia General arrived at 9:01
luisinavasquez  started attending Customer 403 at 9:01.
leidydiaz  started attending Customer 387 at 9:01.
jenosita30  finished attending Customer 399 at 9:01.
sceballos  finished attending Customer 397 at 9:01.
jenosita30  started attending Customer 405 at 9:01.
angelaimitola  finished attending Customer 400 at 9:01.
leidydiaz  finished attending Customer 387 at 9:01.
sceballos  started attending Customer 401 at 9:01.
dianabetancur  started attending Customer 372 at 9:01.
angelaimitola  started attending Customer 402 at 9:01.
luisinavasquez  finished attending Customer 403 at 9:02.
Customer 406 of type Preferencial Farmacia arrived at 9:02
sezaocho  finished attending Customer 386 at 9:02.
dianabetancur  finished attending Customer 372 at 9:02.
Customer 407 of type Preferencial Farmacia arrived at 9:02
Customer 408 of type Preferencial Farmacia arrived at 9:03
vmonsalve  started attending Customer 389 at 9:03.
Customer 409 of type Farmacia General arrived at 9:03
dvalderrama  started attending Customer 395 at 9:03.
angelaimitola  finished attending Customer 402 at 9:03.
angelaimitola  started attending Customer 408 at 9:03.
Customer 410 of type Farmacia PAC, MP y Poliza Sura arrived at 9:03
sceballos  finished attending Customer 401 at 9:03.
Customer 411 of type Farmacia General arrived at 9:03
dvalderrama  finished attending Customer 395 at 9:03.
Customer 412 of type Farmacia General arrived at 9:03
vmonsalve  finished attending Customer 389 at 9:04.
Customer 413 of type Farmacia General arrived at 9:04
sceballos  started attending Customer 409 at 9:04.
```

Figure 6: Examples of output of the simulation.

On the contrary, it is easily used as all of the important aspects of the simulation are packed in different methods of the main class. In this manner, extracting information, testing the model and extracting the outputs are just done by calling those respective functions.

6 RESULTS

We refer to (Robinson 2004, Ch. 9). The system in consideration has a fixed schedule since they open every business day at 6 a.m. and closes at 7 p.m. (Saturdays from 7 a.m. to 1 p.m.), therefore it can be considered as a terminating simulation model. Consequently, the output of the simulation is transient, hence the distribution of the output is constantly changing. For instance, the number of patients is not the same for each day.

The initial condition is assumed as zero because the system does not start with patients inside of it, although, in the complex, patients start arriving before the opening hour and start accumulating. Therefore, it is clear that the distributions in the first hour for all days are biased and this issue cannot be addressed with the available data.

How many times do we have to run the model in order to obtain trustworthy results? When the model is being simulated, it is desired to obtain the highest achievable accuracy, but this often requires a lot of simulation runs and, therefore, computational resources and time. Given that these objectives are conflictive, the idea is to find a balance between them. To achieve this goal, we use the strategy proposed in (Byrne 2013) and (Currie and Cheng 2016).

This methodology is based on confidence intervals; we select a maximum allowed deviation δ, in terms of a proportion of the mean. The simulation has to be previously run n_0 times, to estimate the standard deviation. The appropriate number must satisfy

$$n \geq \left(\frac{t_{\alpha/2;n_0-1} \times \sigma}{\delta} \right)^2 \tag{1}$$

Calculating this value for our model, using the test described previously we obtain a large value of n. The reason for this is because the fit distributions can generate a big amount of outliers, therefore it is difficult to find stability for the means.

Figure 7 is a time series of the average waiting time of the model for a trial of 5 weeks, which shows clearly that the nature of the output is transient since it changes constantly.

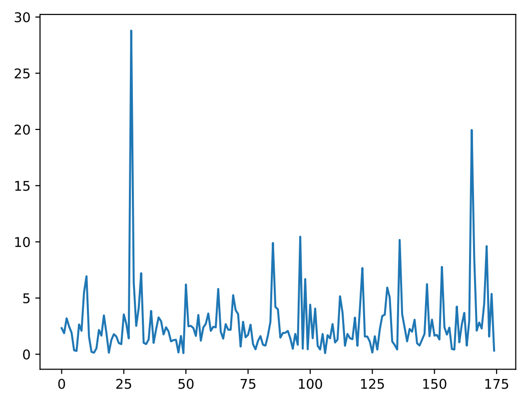


Figure 7: Time series for average waiting time in a week

7 VALIDATION AND VERIFICATION

It is important to remark that some of the procedures that are going to be present in this section are doing twice. Model 1 uses the data of all the month provide by Neuromédica that we are going to refer to it as the model 1 and the other one, i.e model 2 only uses the data of the last of the month.

Although a model cannot be formally validated, we use different forms of validation: black and white validation. White validation is a more empirical procedure, checking that numerous parts of the model operate properly. On the other hand, black-box validation is intended to check the general behavior of the model, comparing it with the real system using confidence intervals (Robinson 2004, Ch. 12).

Furthermore, the waiting time for each of the type of customers was tested with the waiting time obtained in the simulation by the Kruskal-Wallis test.

7.1 White-Box Validation

We used several white-box validation methodologies since the model in consideration does not have a lot of components. The first test consists of submitting the model to extreme conditions, such as:

- One attendant.
- All modules open.

The results using one attendant in the Model 2 are shown in Figure 8 which presented the total number of people that are waiting in the system for one week. This validates the model since it shows that the waiting times augment during the day when it is used only one attendant. It can see a repetitive pattern in the graphic, this represents when the system is closed, which provides a general idea that the model is working well.

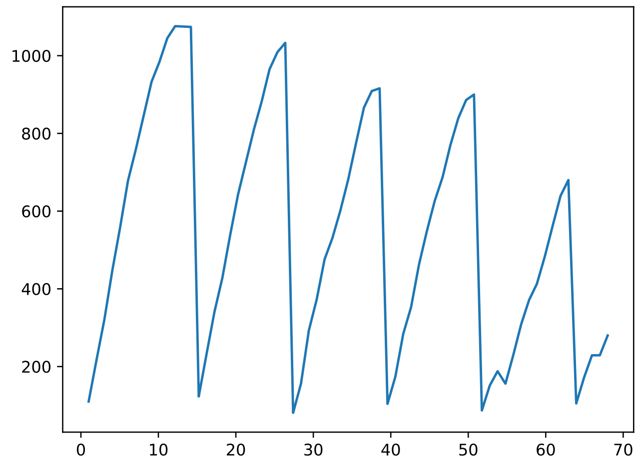
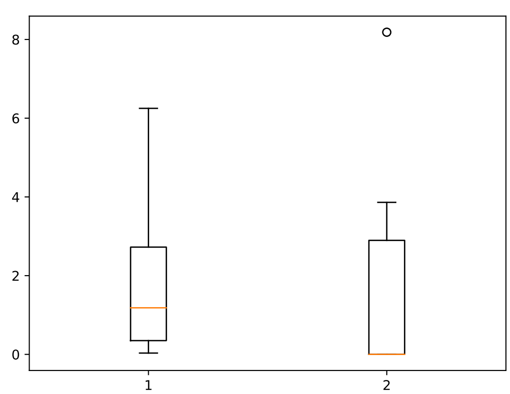


Figure 8: Number of patients for each hour with one attendant

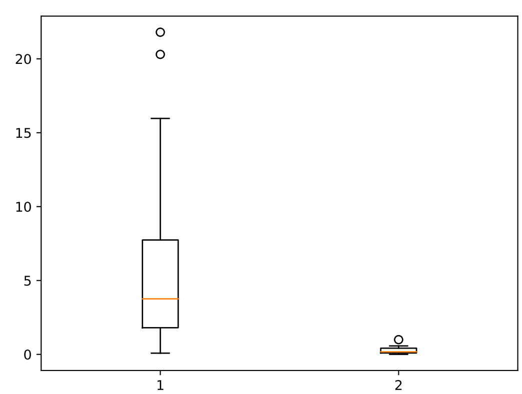
In Figs. 9, 10 it can be seen two different box plots for each of the model compared with the one obtained in the data set. In both, the first box plot determines one in which the test assured that both where homogeneous and the second box plot is non-homogeneous.

It is important to see that in the second model, more outliers are generated. This was done with the intention of simulating more human behavior, as the attendants generally have some erratic behavior which which worsen the system. On the other hand, the second model has a lot more cases in which the data is homogeneous making it preferable to use.

Lastly, it is important to remark that the second model is not homogeneous in the type of client ”Farmacia General” which is the one with the biggest waiting times in the real system. Hence, it is known that although the model works better than the previous one, it is not a fully working simulation model.

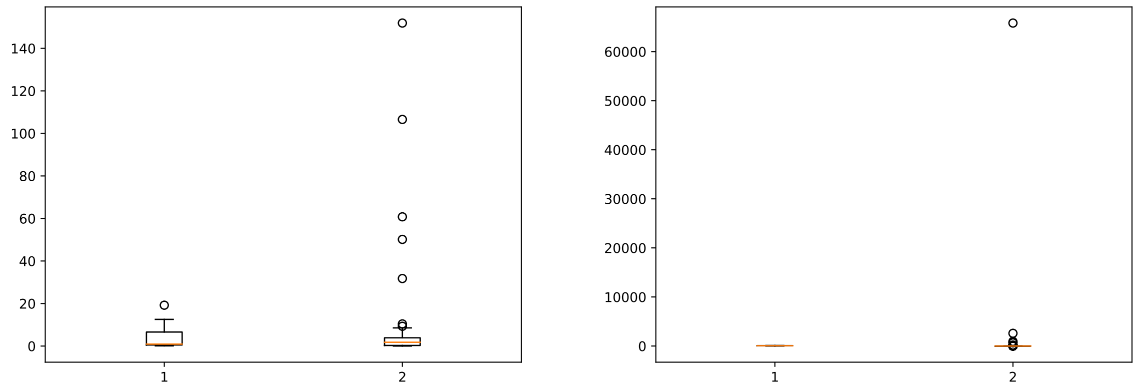


(a) Homogeneous (Pref Confirmación de citas).



(b) Non-Homogeneous (Trámites después de Citas).

Figure 9: Boxplots validation for Model 1.



(a) Homogeneous (Trámites después de Citas). (b) Non-Homogeneous (Farmacia General).

Figure 10: Boxplots validation for Model 2.

Furthermore, whilst the model was being implemented, multiple runs were performed to check initial conditions, arrivals/attention distributions and general proper behavior of the system.

7.2 Black-Box Validation

As previously stated, the general behavior of the model can be analyzed using the corresponding confidence interval to check the difference of the means. The formula its calculation is (Robinson 2004, pp. 218):

$$\bar{X}_S - \bar{X}_R \pm t_{2n-2,\alpha/2}\sqrt{\frac{S_S^2 + S_R^2}{n}} \tag{2}$$

$\bar{X}_S =$ mean of simulated output data

$\bar{X}_R =$ mean of real system output data

$S_S =$ standard deviation of simulated output data

$S_R =$ standard deviation of real system output data

$n =$ number of observations (this must be the same for the simulated and real system data)

$t_{2n-2,\alpha/2} =$ value from Student's t distribution with $2n-2$ degrees of freedom and a significance level of $\alpha/2$

So the obtained result has more sense, it is going to be used the type of pacient "Confirmacion de citas". This was chosen because this type of pacient is the one that the simulation is capable of characterizing. Hence, all experiments are going to be analyzed through the waiting time of this customer.

With this test, it was found that the interval for the difference of the means is:

$$[-3.74976, 1.57127] \tag{3}$$

8 SENSITIVITY ANALYSIS AND EXPERIMENTATION

8.1 Sensitivity Analysis

To see the robustness of the model an sensitivity analysis was achieved. This procedure consists of a variation of some of the parameters of the model between -10% to 10%.

The model, as explained before, some parameters where not able to be fitted because the data was too small. In this manner, we chose to see how the parameters of this elected distributions affected the output of the model. The result if this test is shown in Fig. 11.

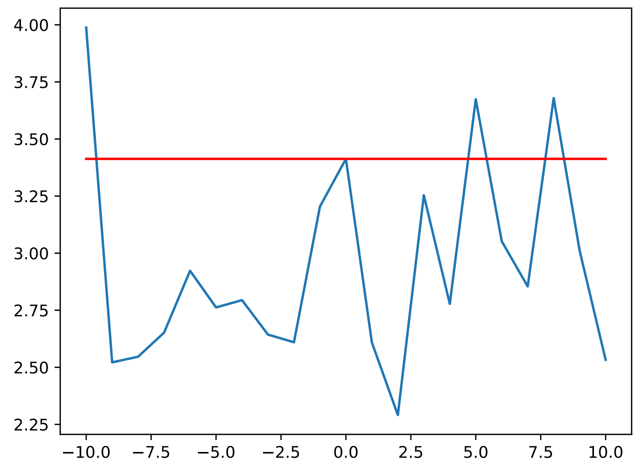


Figure 11: Sensitivity analysis

It is seen that, although the simulation does change while changing the parameters of this distributions, this does not present a major problem as the output of the model does not change drastically. Hence, it is confirmed that the suppositions are not affecting in a damaging manner the simulation.

8.2 Optimization

The priority that each module attend (Table 2) is not only is a key factor to consider in the queue logistics, but also a highly feasible strategy to implement in the real system since it only requires the adjustment on the queueing platform. The idea is to find the best combinations of priorities per (open) module to minimize the average queue time. The optimization scheme we propose takes the initial "solution", i.e. Table 5, and apply a heuristic approach based on local search.

Set	**Amount**	**Priority**
1	1	[10]
2	1	[4, 9, 5]
3	1	[4, 5, 9]
4	2	[7, 3, 8, 10]
5	4	[3, 8, 7, 10]
6	5	[3, 7, 8, 10]

Table 5: Sets and priorities.

We use a standard Variable Neighborhood Search (VNS) (Hansen, Mladenović, and Urošević 2006), using classic neighborhood operators:

- In-Set Swap: Swaps two priorities within the same set (Figure 12).

$$\text{set } k\ [p_1\ p_2 \cdots p_i \cdots p_j \cdots p_n]$$

$$\text{set } k\ [p_1\ p_2 \cdots p_j \cdots p_i \cdots p_n]$$

Figure 12: In-set swap operator.

- Best Insertion: Takes a node and attempts to find the best insertion within the set (Figure 13).

$$\text{set } k\ [p_1\ p_2 \cdots p_{i-1}\ p_i\ p_{i+1} \cdots p_j\ \ p_{j+1} \cdots p_n]$$

$$\text{set } k\ [p_1\ p_2 \cdots p_{i-1}\ p_{i+1} \cdots p_j\ p_i\ \ p_{j+1} \cdots p_n]$$

Figure 13: Best-insertion in-set operator.

- Inter-Set Swap: Same as in-set, but attempting between two nodes in different sets (14).

$$\text{set } k_1 [p_1\ p_2 \cdots p_i \cdots p_n]$$

$$\text{set } k_2 [q_1\ q_2 \cdots q_j \cdots q_m]$$

Figure 14: Inter-set swap operator.

We refer to (Paraskevopoulos, Repoussis, Tarantilis, Ioannou, and Prastacos 2008) and (Fosin, Carić, and Ivanjko 2014) for the local search operators. If the third operator swaps a priority into a set that already had that priority, one is chosen randomly and deleted. Otherwise, we assume that the number of priorities will not change, e.g. the number of appearances in all sets of policy “3” remains constant on each iteration and so forth.

After the optimization is done, an ANOVA test is performed in order to check if the mean of the output data is significantly improved after the optimization (Wackerly, Muñoz, Humbertotr, et al. 2010, pp. 662-665).

The result of this heuristic is 6. Furthermore, the evolution of the waiting time for the selected type of patient is shown in the Fig. 15

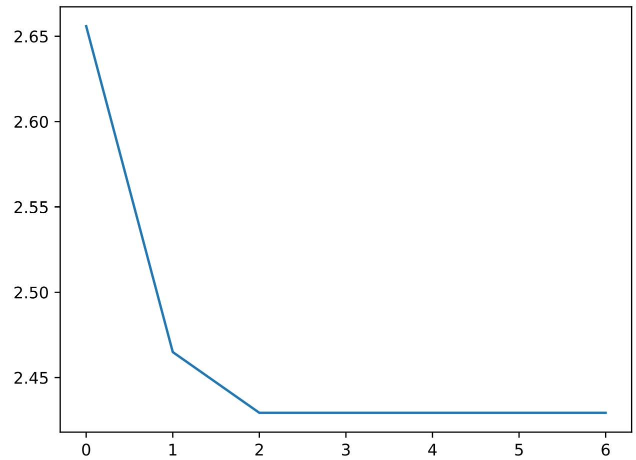


Figure 15: Waiting time for selected pacient changing over the iterations of VNS

Set	Amount	Priority
1	1	[5,4,9]
2	1	[10]
3	1	[8,10,7,3]
4	1	[7,3,10,8]
5	1	[8,10,3,7]
6	1	[3,7,8,10]
7	1	[8,3,7,10]
8	1	[7,8,3,10]
9	1	[8,7,3,10]
10	1	[10,7,3,8]
11	1	[3,8,7,10]
12	1	[7,3,8,10]

Table 6: Found table of priorities.

It is important to notice that the best solution found was by using all the resources in all the different ways possible, so each one attend a different and unique priority. This is due to using all the resources to attend all the possible priorities, therefore reducing this waiting time.

9 CONCLUSIONS

It was successfully analyzed the received data, using statistical tests which allowed to fit data to use in the simulation, find auto-correlation and test homogeneity between data. It is important to remark that in the fitting of the data the distributions have a left bias as most of the times are positive and close to zero.

It was successfully implemented the simulation model in Python, which partially represents the Neuromédica pharmacy medicine retriveal system. The model did not fully validate with the tests done, which presents an important limitation of the implemented model; even though, the results given are not useful for the real system, this article proposes a methodology for solving this types of problems.

For further work, a model using continuous simulation could be used as the inter-arrival times are so close to zero that the system could be analyzed supposing a continuous flux of patients. Also, a graphical user interface can be done so it is easily explained without the use of code.

REFERENCES

Banks, J., C. J., N. B.L., and N. D.. 2005. *Discrete-event System Simulation*. Prentice-Hall international series in industrial and systems engineering. Pearson Prentice Hall.

Byrne, M. D. 2013. "How many times should a stochastic model be run? An approach based on confidence intervals". In *Proceedings of the 12th International conference on cognitive modeling, Ottawa*.

Cobham, A. 1954. "Priority assignment in waiting line problems". *Journal of the Operations Research Society of America* 2 (1): 70–76.

Currie, C. S., and R. C. Cheng. 2016. "A practical introduction to analysis of simulation output data". In *2016 Winter Simulation Conference (WSC)*, 118–132. IEEE.

Fosin, J., T. Carić, and E. Ivanjko. 2014. "Vehicle routing optimization using multiple local search improvements". *automatika* 55 (2): 124–132.

Hansen, P., N. Mladenović, and D. Urošević. 2006. "Variable neighborhood search and local branching". *Computers & Operations Research* 33 (10): 3034–3045.

Jun, J., S. H. Jacobson, and J. R. Swisher. 1999. "Application of discrete-event simulation in health care clinics: A survey". *Journal of the operational research society* 50 (2): 109–123.

Lakshmi, C., and S. A. Iyer. 2013. "Application of queueing theory in health care: A literature review". *Operations research for health care* 2 (1-2): 25–39.

Ndukwe, H., S. Omale, and O. Opanuga. 2011. "Reducing queues in a Nigerian hospital pharmacy".

Paraskevopoulos, D. C., P. P. Repoussis, C. D. Tarantilis, G. Ioannou, and G. P. Prastacos. 2008. "A reactive variable neighborhood tabu search for the heterogeneous fleet vehicle routing problem with time windows". *Journal of Heuristics* 14 (5): 425–455.

Reynolds, M., C. Vasilakis, M. McLeod, N. Barber, A. Mounsey, S. Newton, A. Jacklin, and B. D. Franklin. 2011. "Using discrete event simulation to design a more efficient hospital pharmacy for outpatients". *Health care management science* 14 (3): 223–236.

Robinson, S. 2004. *Simulation: The Practice of Model Development and Use*. Wiley.

Shimshak, D. G., D. G. Damico, and H. D. Burden. 1981. "A priority queuing model of a hospital pharmacy unit". *European journal of operational Research* 7 (4): 350–354.

Thorwarth, M., and A. Arisha. 2009. "Application of discrete-event simulation in health care: a review".

Wackerly, D. D., R. Muñoz, J. Humbertotr et al. 2010. *Estadística matemática con aplicaciones*. Number 519.5 W3.

A Appendix

A.1 Distribution Fitting

As mentioned in Section 4, the fitting of distributions was performed in Python using SciPy. The fitting was performed with the Kolmogorov-Smirnov test against all the available theoretical distributions. In Table 7 the best fit distributions for each day and each hour are presented. We use the conventions in SciPy. The distribution name matches the online documentation and the parameters have the following convention: [*distribution parameters*, location, scale]. All the parameters are used according to the probability density functions presented in SciPy documentation.

	Service	**Distribution**	**p-value**	**Parameters**
Att 16	Farmacia General	fatiguelife	0.99057	[1.62781, 0.13875, 0.17738]

	Preferencial Farmacia	norm	-	[0.49259, 0.52202]
Att 13	Farmacia General	nct	0.89895	[1.06936, 2.09336, 0.10515, 0.12759]
	Preferencial Farmacia	norm	-	[0.62333, 0.4897]
Att 11	Pref Confirmacion de Citas	ncf	0.99829	[0.33131, 15.96713, 6.41388, -0.00012, 0.20353]
	Confirmacion de citas	johnsonsu	0.97981	[-6.28171, 1.2888, -0.77047, 0.06161]
	Tramites despues de Citas	genexpon	0.96818	[67084310183.16647, 98784169004.66412, 115812276725.65875, -0.0, 470313120150.21686]
	Confirmacion Citas PAC, MP y Poliza Sura	norm	-	[2.37121, 4.89016]
Att 24	Farmacia General	chi	0.70929	[0.61509, -0.0, 5.73994]
	Farmacia PAC, MP y Poliza Sura	norm	-	[6.93333, 0.6364]
	Preferencial Farmacia	johnsonsb	0.98175	[0.56148, 0.71294, 0.31001, 11.24679]
Att 8	Farmacia General	invgauss	0.58183	[0.77192, -0.27563, 3.82245]
	Farmacia PAC, MP y Poliza Sura	norm	-	[5.08333, 2.07418]
	Preferencial Farmacia	loglaplace	0.99818	[1.41177, 0.20017, 2.03994]
Att 4	Farmacia General	johnsonsu	0.69515	[-2.1781, 0.66373, 0.12293, 0.04356]
	Farmacia PAC, MP y Poliza Sura	foldcauchy	0.99746	[1.12262, 0.03333, 0.13716]
	Preferencial Farmacia	johnsonsu	0.39988	[-1.48111, 0.55217, 0.16105, 0.03756]
	Entrega Programada	norm	-	[0.80417, 1.14465]
Att 12	Farmacia General	norm	-	[0.36515, 0.12615]
	Farmacia PAC, MP y Poliza Sura	norm	-	[0.47222, 0.05358]
	Preferencial Farmacia	johnsonsu	0.51968	[-1.36186, 0.81497, 0.24959, 0.05623]
Att 17	Farmacia PAC, MP y Poliza Sura	johnsonsu	0.8192	[-1.13296, 0.73728, 0.25054, 0.0506]

	Preferencial Farmacia	johnsonsu	0.48031	[-1.23664, 0.64863, 0.23181, 0.04298]
	Entrega Programada	norm	-	[0.80417, 0.35833]
Att 14	Farmacia General	johnsonsu	0.12823	[-1.29847, 0.64395, 0.20378, 0.0373]
	Preferencial Farmacia	johnsonsu	0.8928	[-0.70984, 0.42431, 0.30924, 0.02746]
Att 23	Farmacia General	johnsonsu	0.38527	[-1.27092, 0.5069, 0.22385, 0.04632]
	Farmacia PAC, MP y Poliza Sura	johnsonsu	0.90203	[-0.87856, 0.88658, 0.21585, 0.11953]
	Preferencial Farmacia	nct	0.44984	[1.28693, 2.29516, 0.16159, 0.05815]
Att 15	Farmacia General	exponpow	0.99261	[0.65811, 0.16667, 0.6911]
	Entrega Programada	norm	-	[0.51667, 0.02357]
Att 3	Farmacia General	fisk	0.99853	[1.47967, 0.09026, 0.35672]
	Farmacia PAC, MP y Poliza Sura	norm	-	[0.64762, 0.32595]
	Preferencial Farmacia	nct	0.65052	[1.71681, 3.05122, 0.09044, 0.07231]
Att 22	Farmacia General	mielke	0.99987	[12.38816, 0.84125, 0.10811, 0.01271]
	Farmacia PAC, MP y Poliza Sura	johnsonsu	0.927	[-0.553, 0.60398, 0.34656, 0.08497]
	Preferencial Farmacia	nct	0.71151	[0.95348, 1.92681, 0.19793, 0.08532]
	Entrega Programada	halfcauchy	0.96395	[0.33333, 0.45007]
Att 18	Farmacia General	johnsonsu	0.85574	[-1.5017, 0.79096, 0.20804, 0.07408]
	Farmacia PAC, MP y Poliza Sura	nct	0.94507	[1.06208, 1.73919, 0.27715, 0.15981]
	Preferencial Farmacia	norm	-	[0.24722, 0.09511]
Att 6	Farmacia General	johnsonsu	0.58099	[-1.42415, 0.61752, 0.21081, 0.04098]
	Farmacia PAC, MP y Poliza Sura	norm	-	[0.33667, 0.08283]

	Preferencial Farmacia	nct	0.41817	[1.06735, 1.89027, 0.19151, 0.06387]
Att 7	Farmacia General	nakagami	0.86245	[0.29563, -0.0, 10.97817]
	Farmacia PAC, MP y Poliza Sura	norm	-	[5.46167, 7.22277]
	Preferencial Farmacia	norm	-	[4.32917, 3.7765]
	Entrega Programada	norm	-	[4.58, 9.56146]
Att 21	Farmacia General	johnsonsu	0.97683	[-0.90044, 0.89261, 0.19759, 0.06815]
	Farmacia PAC, MP y Poliza Sura	norm	-	[0.41833, 0.45498]
	Preferencial Farmacia	alpha	0.96406	[2.72941, -0.09125, 1.18675]
	Entrega Programada	norm	-	[0.34167, 0.3182]
Att 20	Pref Confirmacion de Citas	nakagami	0.98859	[0.31051, 0.36667, 2.96453]
	Confirmacion de citas	mielke	0.8015	[1.12627, 2.92181, 0.11352, 3.57187]
	Confirmacion Citas PAC, MP y Poliza Sura	norm	-	[1.83182, 1.38647]
	Tramites despues de Citas	johnsonsb	0.96377	[0.98592, 0.70016, 0.08997, 11.45658]
Att 5	Farmacia PAC, MP y Poliza Sura	invgamma	0.9992	[2.05702, -0.8456, 7.75335]
	Preferencial Farmacia	norm	-	[2.70167, 2.46779]
	Entrega Programada	norm	-	[5.02, 8.27435]
Att 10	Farmacia General	nct	0.93934	[0.65037, 1.83056, 0.16605, 0.04542]
	Farmacia PAC, MP y Poliza Sura	johnsonsu	0.99824	[-1.21136, 0.51816, 0.21761, 0.03192]
	Preferencial Farmacia	johnsonsu	0.74634	[-1.09901, 0.59292, 0.24489, 0.0431]
	Entrega Programada	norm	-	[3.29111, 5.83623]
Att 2	Farmacia General	weibull_min	0.99138	[0.75254, 0.06667, 3.25703]

	Farmacia PAC, MP y Poliza Sura	laplace	0.94197	[4.05, 2.19116]
	Preferencial Farmacia	exponpow	0.99944	[0.60618, 0.23333, 4.92841]
	Entrega Programada	norm	-	[1.93889, 3.60495]
Att 9	Farmacia General	johnsonsu	0.88688	[-5.09161, 1.07842, -0.24934, 0.04755]
	Farmacia PAC, MP y Poliza Sura	norm	-	[4.29545, 4.93202]
	Preferencial Farmacia	norm	-	[2.11389, 2.31354]
	Entrega Programada	norm	-	[4.25, 5.26326]
Att 19	Farmacia General	johnsonsb	0.50391	[8.74138, 1.38249, -0.49468, 1834.28761]
	Preferencial Farmacia	nakagami	0.91117	[0.37578, 0.23333, 4.78737]
Att 1	Farmacia General	gengamma	0.45301	[10.30977, 0.35725, -0.08809, 0.00339]
	Farmacia PAC, MP y Poliza Sura	norm	-	[2.55, 1.18782]
	Preferencial Farmacia	norm	-	[2.55833, 1.37886]
	Entrega Programada	norm	-	[0.59375, 0.63076]

Table 7: Attendants distribution fitting by hours and service.

Date	**Hour**	**Distribution**	**p-value**	**Parameters**
Monday 26	6:00-7:00	lognorm	0.92735	[0.91273, 0.11933, 0.18633]
	7:00-8:00	johnsonsu	0.78289	[-1.79996, 1.03526, 0.19108, 0.06859]
	8:00-9:00	alpha	0.87654	[2.87705, -0.0595, 1.17314]
	9:00-10:00	foldcauchy	0.86375	[1.18657, 0.16667, 0.09681]
	10:00-11:00	johnsonsu	0.82017	[-1.16378, 0.85848, 0.20883, 0.06324]
	11:00-12:00	invgamma	0.61487	[2.46322, 0.00954, 0.64648]
	12:00-13:00	beta	0.99136	[1.04874, 97492228.74675, 0.13313, 37741066.13559]
	13:00-14:00	burr	0.8261	[2.37289, 41.07014, 0.05157, 0.06283]

	14:00-15:00	johnsonsu	0.85535	[-1.03125, 0.96418, 0.31182, 0.11123]
	15:00-16:00	foldcauchy	0.84493	[1.49348, 0.11667, 0.14705]
	16:00-17:00	loglaplace	0.78622	[1.39784, 0.12699, 0.23968]
	17:00-18:00	johnsonsb	0.99517	[1.50618, 0.81644, 0.14092, 4.33617]
	18:00-19:00	lomax	0.81476	[28.64119, 0.21667, 15.74325]
Tuesday 27	6:00-7:00	alpha	0.86891	[2.20287, 0.00396, 0.74495]
	7:00-8:00	johnsonsu	0.64997	[-1.36029, 1.01153, 0.24694, 0.07437]
	8:00-9:00	johnsonsb	0.6316	[8.07477, 1.37157, 0.13379, 94.46084]
	9:00-10:00	johnsonsu	0.80292	[-1.23989, 1.0254, 0.23958, 0.08885]
	10:00-11:00	johnsonsu	0.55124	[-1.26769, 0.92585, 0.25384, 0.05091]
	11:00-12:00	johnsonsu	0.91897	[-1.03936, 0.85618, 0.26707, 0.06578]
	12:00-13:00	exponweib	0.99446	[56.65721, 0.29109, 0.15841, 0.0016]
	13:00-14:00	burr	0.742	[2.33412, 2.56205, 0.09438, 0.17955]
	14:00-15:00	loglaplace	0.77161	[1.53339, 0.17669, 0.25664]
	15:00-16:00	johnsonsu	0.91211	[-1.80142, 1.01405, 0.21918, 0.06851]
	16:00-17:00	fisk	0.99743	[1.59009, 0.21202, 0.23038]
	17:00-18:00	gengamma	0.99938	[1.02027, 0.90964, 0.21667, 0.77741]
	18:00-19:00	gamma	0.987	[0.73154, 0.21667, 1.7865]
Wednesday 28	6:00-7:00	johnsonsu	0.79231	[-2.13376, 0.94118, 0.18939, 0.03901]
	7:00-8:00	johnsonsu	0.53049	[-1.33043, 0.7488, 0.24046, 0.04609]
	8:00-9:00	johnsonsu	0.91021	[-1.46346, 0.8543, 0.27322, 0.03896]
	9:00-10:00	betaprime	0.97344	[346.46788, 3.54218, 0.06161, 0.00335]
	10:00-11:00	fisk	0.98687	[2.31011, 0.13201, 0.22175]
	11:00-12:00	betaprime	0.87292	[40.95951, 1.908, 0.09518, 0.01308]

	12:00-13:00	foldcauchy	0.97268	[2.26431, 0.13333, 0.11112]
	13:00-14:00	johnsonsu	0.95613	[-1.5493, 0.99582, 0.27035, 0.08264]
	14:00-15:00	johnsonsu	0.94757	[-1.88627, 1.16978, 0.20176, 0.0945]
	15:00-16:00	johnsonsu	0.79988	[-1.55693, 0.82651, 0.2533, 0.05144]
	16:00-17:00	foldcauchy	0.98139	[0.87319, 0.2333, 0.18948]
	17:00-18:00	johnsonsu	0.99974	[-1.90268, 1.00767, 0.27625, 0.17265]
	18:00-19:00	fatiguelife	0.81624	[1.67466, 0.24489, 0.69849]
Thursday 29	6:00-7:00	foldcauchy	0.82509	[1.99899, 0.13333, 0.09163]
	7:00-8:00	johnsonsu	0.80807	[-1.54583, 0.95217, 0.22393, 0.08193]
	8:00-9:00	exponweib	0.57104	[98.6175, 0.46902, 0.08717, 0.01065]
	9:00-10:00	johnsonsu	0.92695	[-1.6203, 0.97081, 0.23532, 0.0816]
	10:00-11:00	nct	0.90124	[2.10262, 2.07561, 0.19427, 0.07789]
	11:00-12:00	johnsonsu	0.90969	[-1.48168, 0.98104, 0.2434, 0.07366]
	12:00-13:00	alpha	0.76294	[1.57887, 0.0345, 0.6142]
	13:00-14:00	johnsonsu	0.94077	[-1.52673, 0.89291, 0.24903, 0.08479]
	14:00-15:00	alpha	0.89964	[2.2197, 0.01165, 0.90375]
	15:00-16:00	mielke	0.83486	[128.74968, 2.46165, -0.01524, 0.08747]
	16:00-17:00	burr	0.98	[1.50408, 2.21761, 0.14066, 0.17991]
	17:00-18:00	johnsonsb	0.99367	[3.36774, 0.81745, 0.20265, 22.44792]
	18:00-19:00	f	0.92346	[5.20197, 1.67781, 0.18672, 0.33884]
Friday 30	6:00-7:00	invgauss	0.80879	[2.2255, 0.16826, 0.24136]
	7:00-8:00	ncf	0.29092	[2.99982, 3.72833, 270.22994, 0.12008, 0.00278]
	8:00-9:00	foldcauchy	0.87901	[1.78653, 0.15, 0.11516]
	9:00-10:00	johnsonsu	0.49965	[-1.03596, 0.85416, 0.27657, 0.0822]

	10:00-11:00	powerlognorm	0.6541	[0.9033, 0.82938, 0.16589, 0.25331]
	11:00-12:00	johnsonsu	0.82742	[-1.11264, 0.7743, 0.28324, 0.0516]
	12:00-13:00	alpha	0.97102	[2.34122, -0.07718, 1.17781]
	13:00-14:00	alpha	0.63484	[1.34062, 0.03509, 0.56573]
	14:00-15:00	johnsonsu	0.78728	[-1.55962, 0.76181, 0.27247, 0.04114]
	15:00-16:00	johnsonsu	0.77648	[-1.46487, 0.81995, 0.257, 0.04651]
	16:00-17:00	johnsonsu	0.83958	[-1.32962, 0.77857, 0.28046, 0.06247]
	17:00-18:00	johnsonsu	0.84887	[-1.1286, 0.69075, 0.35424, 0.0634]
	18:00-19:00	chi2	0.88327	[1.32993, 0.18333, 0.78942]
Saturday 31	7:00-8:00	lomax	0.25193	[3.22539, 0.11667, 0.76482]
	8:00-9:00	invgamma	0.92703	[1.92871, 0.13517, 0.55124]
	9:00-10:00	loglaplace	0.78948	[1.19366, 0.21467, 0.28533]
	10:00-11:00	fisk	0.88252	[1.7456, 0.1734, 0.27987]
	11:00-12:00	johnsonsu	0.97212	[-2.05118, 0.95597, 0.18942, 0.08072]
	12:00-13:00	johnsonsu	0.38118	[-1.45688, 0.65743, 0.2781, 0.0595]

Table 8: Inter-arrival times distribution by hours.