Second assignment (40%)

Simulation for Business Decision Making

Alexander J. Ohrt
UPC
alexander.johansen.ohrt@estudiantat.upc.edu

1 Abstract.

A brief and concise description of what is going to be analyzed, the problem and the solution proposed.
Review the abstract based on the new analysis obtained on the DOE.

2 System description, introduction.

This section **only contains the description of the system to be modeled** (not the problem, not the data), other elements must be described on the subsequent sections of the document.

3 Problem description.

Describe the problem you want to solve using simulation. Be as clear and concise as possible.

3.1 Hypotheses/Restrictions in the model.

3.1.1 Structural and Simplifying Hypotheses.

In this assignment we will be focused on the execution and experimentation with the model.

4 Model specification.

We quickly review the model we defined in the first assignment. I think the model was defined in a reasonable way and I do not want to make any changes to it at this point. Of course it is a very small model, with a lot of restrictions, which means that it might not be extremely useful in a real practical application. However, in the scope of this assignment I believe it is fitting to continue to use this simple model, in order to facilitate better understanding of the concepts to come. Moreover, this will ensure that the experimental design is kept as simple as possible — we will not be including any artifacts that make the experiment complex and difficult to understand.

5 Definition of the experimental framework.

We explain the design of experiments (DOE) to be used and detail the process to execute the replications.

First of all, it is necessary to clearly define the objectives of the experimentation. Following the discussion in Assignment 1, the goal is to figure out how to maximize the number of customers, among the customers that enter the store, that can be successfully served during the opening hours of the store. Thus, we monitor the proportion of successful check-outs during each simulation of the supermarket. The final proportion after concluding the daily simulation is recorded. In technical terms, we define the *response variable* as the proportion of successful check-outs during the opening hours of the supermarket.

The Design of Experiments (DOE) framework that will be used here is a 2^4 factorial design. The factors that will be tested are

- Number of manual cashiers; few or many.
- Number of automatic cashiers; few or many.
- Speed of manual cashiers, i.e. the speed of the employees; low speed or high speed.
- Speed of automatic cashiers; low speed or high speed.

This yields 2^4 = 16 scenarios which need to be analyzed. We limit the levels in each factor to two, since the experiment can quickly explode and become very large, even for this small problem, when including more levels. Obviously, this does not give great precision to the experimental design and we cannot find statistical evidence about any other levels besides the ones we choose. However, the idea is that we can define one level as high and one level as low, in each factor, and therefore still be able to explore the response surface, which will allow some understanding of the changes in the response variable depending on the levels. Thus, the main advantage of restricting the levels of each factor to two is that one can (hopefully) find which factors have the largest impact, with a fraction of the cost of a full factorial design with more levels available to each factor. This facilitates further investigation of the factors that are found to have a large impact later on, if this is needed.

The levels should be chosen with great care, taking cost, availability and many other factors into account. Since we don't have any information about such factors, we choose the levels based on experience we gained with the system in Assignment 1. The levels are chosen as follows

- Number of manual cashiers (A): A^- = 3 and A^+ = 5
- Number of automatic cashiers (B); B^- = 1 and B^+ = 3
- Speed of manual cashiers (C): C^- = U[4,8] and C^+ = U[2,6]
- Speed of automatic cashiers (D); D^- = Exp(6) and D^+ = Exp(4)

Notice that the levels in the two last factors, concerning the speed of the types of cashiers, are specified using the distribution of time that settling of payment takes. In Assignment 1, we defined the settling time at the manual cashiers as following a uniform distribution from 3 to 7 minutes, i.e. U[3,7]. In this case we decide to test (expected) settling times shorter and longer than the time that was specified in part 1. In practice, these changes could for example be interpreted as either hiring more experienced cashiers, training the cashiers to be more efficient or increasing the efficiency of the cashier-machine, which leads to the high level of the factor, or hiring less experienced cashiers, which leads to the low level of the factor. A similar description can be done concerning the settling time of the automatic cashiers, which was defined to follow an exponential distribution with mean 6 minutes in Assignment 1, i.e. Exp(6). Here we define the high factor level as Exp(4) and the low factor level as Exp(6), where the low level could mean keeping the same automatic cashiers in the supermarket and the high level could mean buying newer and more efficient automatic cashiers.

The system is modeled as a terminate system, because there is a cyclic state that represents the end of the system; the store opens in the morning and closes after 10 hours. The closing time represents the state where we terminate the simulations, and new simulations can be started afterwards. Because of this, the *n* replications will be executed using independent repetitions, simply by simulating the system *n* times for each combination of factor levels and recording the results.

We assume no transient period in the simulations, because the system is terminate and the entire opening interval contains correct values. This means that the replications are recorded as described, without the need to worry about erasing loading periods in each replication. How many replications are needed for this experimental design? There is no clear theoretically backed answer to this question. We follow the heuristic often used in computing of beginning with 10 replications to get started. Then we need to check that this is enough to get the a priori specified statistical confidence in the results. A priori we decide that we want a confidence interval with upper and lower limits within 5% of the sample mean, with a 95% confidence level. This information will be used to calculate the needed number of replications, after first having run the experiment for 10 replications.

As a sidenote to the experimental design, blocking will not be necessary since the experiment will be performed by me, where all replications are performed using Experiments in GPSS.

The table showing the 16 different scenarios is shown below.

Table 1: Experimental design matrix

"A"	"B"	"C"	"D"
"-"	"-"	"-"	"-"
"+"	"-"	"-"	"-"
"-"	"+"	"-"	"-"
"+"	"+"	"-"	"-"
"-"	"-"	"+"	"-"
"+"	"-"	"+"	"-"
"-"	"+"	"+"	"-"
"+"	"+"	"+"	"-"
"-"	"-"	"-"	"+"
"+"	"-"	"-"	"+"
"-"	"+"	"-"	"+"
"+"	"+"	"-"	"+"
"-"	"-"	"+"	"+"
"+"	"-"	"+"	"+"
"-"	"+"	"+"	"+"
"+"	"+"	"+"	"+"

6 Coding.

We implement the experimentation in GPSS, using the implementation of the model from Assignment 1. The implementation is delivered together with this report. Notice that I have not been able to get the simulation to run properly, but I deliver it anyway, in hopes of being relatively close to a solution that will work, i.e. minimal tweaks of the code would hopefully have it working as expected.

The number of replications for each scenario can be calculated by using the formula $n' = n(h/h')^2$, where n' is the total number of replications needed, n is the initial number of replications used (10 in this case), h is the half-range of the confidence interval obtained with the initial number of replications and h' is the desired half-range (5% of the obtained sample mean in this case). Once we have the data correctly sampled for each scenario, we calculate the mean response for each scenario and apply Yates algorithm to determine the interaction and the effects of each factor.

6.1 Data.

Since I have not been able to codify the experiment for several repetitions correctly I will simply fabricate some mean values and “pretend” like they were the results from the simulated system. This is done in order to continue with the analysis, i.e. apply Yates algorithm to calculate some effects. If I am able to get the code working in the end, I can simply switch out the data and redo the calculations. The results are shown in the table below.

Table 2: Results for each scenario (fabricated)

"A"	"B"	"C"	"D"	"y"
"-"	"-"	"-"	"-"	0.88
"+"	"-"	"-"	"-"	0.93
"-"	"+"	"-"	"-"	0.91
"+"	"+"	"-"	"-"	0.96
"-"	"-"	"+"	"-"	0.91
"+"	"-"	"+"	"-"	0.94
"-"	"+"	"+"	"-"	0.97
"+"	"+"	"+"	"-"	0.98
"-"	"-"	"-"	"+"	0.95
"+"	"-"	"-"	"+"	0.94
"-"	"+"	"-"	"+"	0.96
"+"	"+"	"-"	"+"	0.98
"-"	"-"	"+"	"+"	0.97
"+"	"-"	"+"	"+"	0.98
"-"	"+"	"+"	"+"	0.98
"+"	"+"	"+"	"+"	0.995

6.2 Effect Calculations

Using the data shown in table 2, we calculate the effects using Yates algorithm. The results are shown below.

Table 3: Yates algorithm on first fabricated data

Scenario	y	1	2	3	4	Effect
Mean	0.88	1.81	3.68	7.48	15.23	0.951875
A	0.93	1.87	3.8	7.75	0.17	0.02125
B	0.91	1.85	3.83	0.14	0.24	0.03
AB	0.96	1.95	3.92	0.03	0.02	0.0025
C	0.91	1.89	0.1	0.16	0.21	0.02625
AC	0.94	1.94	0.04	0.08	-0.05	-0.00625
BCD	0.97	1.945	0.01	-0.02	0.02	0.0025
ABC	0.98	1.975	0.02	0.04	-0.04	-0.005
D	0.95	0.05	0.06	0.12	0.27	0.03375
AD	0.94	0.05	0.1	0.09	-0.11	-0.01375
BD	0.96	0.03	0.05	-0.06	-0.08	-0.01
ABD	0.98	0.01	0.03	0.01	0.06	0.0075
CD	0.97	-0.01	0	0.04	-0.03	-0.00375
ACD	0.975	0.02	-0.02	-0.02	0.07	0.00875
BCD	0.98	0.005	0.03	-0.02	-0.06	-0.0075
ABCD	0.995	0.015	0.01	-0.02	0	0

We use the value in the Mean row as a reference. Obtaining a value of the same magnitude as this implies that the respective factor (or the respective interaction) has a considerable effect. However, as we can see from the table above, there are no effect values of the same magnitude as the reference value, meaning that we cannot state that any of the effects are significant.

For illustration purposes, we change the result values (the y vector) drastically, in order to see if we can obtain any significant effects and perform some interpretations of them. Keep in mind that all conclusions drawn based on this data cannot be used for our system, since the data is fabricated. The new results are shown below.

Table 4: Yates algorithm on second fabricated data

Scenario	y	1	2	3	4	Effect
Mean	0.7	1.55	3.24	6.97	13.09	0.818125
A	0.85	1.69	3.73	6.12	0.67	0.08375
B	0.8	1.78	2.7	0.37	0.15	0.01875
AB	0.89	1.95	3.42	0.3	0.05	0.00625
C	0.83	1.5	0.24	0.31	1.21	0.15125
AC	0.95	1.2	0.13	-0.16	0.39	0.04875
BCD	0.97	1.64	-0.1	-0.17	0.47	0.05875
ABC	0.98	1.78	0.4	0.22	-0.83	-0.10375
D	0.9	0.15	0.14	0.49	-0.85	-0.10625
AD	0.6	0.09	0.17	0.72	-0.07	-0.00875
BD	0.5	0.12	-0.3	-0.11	-0.47	-0.05875
ABD	0.7	0.01	0.14	0.5	0.39	0.04875
CD	0.65	-0.3	-0.06	0.03	0.23	0.02875
ACD	0.99	0.2	-0.11	0.44	0.61	0.07625
BCD	0.86	0.34	0.5	-0.05	0.41	0.05125
ABCD	0.92	0.06	-0.28	-0.78	-0.73	-0.09125

In this case we can make some more interesting observations. We can see that the absolute values of the effects of factor C and D, as well as the effect of the interaction between A, B and C are of similar magnitude as the reference value. Thus, we observe that increasing the speed of the manual cashiers to the higher level of factor C yields a larger response, i.e. a larger proportion of successfully attended customers. The opposite can be said about the high level of factor D and the interaction between A, B and C; having D in the high level or having A, B and C in their respective high levels both lead to a lower response, i.e. the store is able to successfully attend a smaller proportion of customers. A similar conclusion can be made for the interaction between A, B, C and D, as its absolute value is almost as big as for the interaction between A, B and C. For the record, I do not think this makes sense at all in the real system. These interpretations are a result of the random assignment of the fabricated results I have done in this case. The purpose of this fabrication is not to learn anything about the system, but to simply show how the calculations can be interpreted. As already stated, the same calculations should be done when the correctly simulated values are recorded.

Notice that with the values of each of the scenarios from each of the repetitions, one should calculate confidence intervals of the effects also. Since confidence intervals and hypothesis tests require essentially the same calculations, one could use these results to statistically test whether or not an effect is significant.

7 Model validation.

We propose some methods to perform validation and verification of the model. In addition, we describe the behaviour of the model.

An informal static validation technique that could be used is Face validation. In this technique we ask experts to analyze the conceptual model description and they are (often) able to recognize the correctness of the model. The focus lies on the conceptual model, without taking any implementation into account. Thus, this technique (and similar techniques within the same group) are focused on the analysis of the structural and simplifying assumptions of the model, as well as on analyzing the correctness of the relations. Experts will often be able to point out errors in the assumptions, either if there are any missing assumptions or any assumptions that oversimplify the model. In my case I would contact experts in the supermarket sector, or even the more general industrial system sector, in order to get some feedback on the conceptual model I have set up and on the assumptions I have made. Then, I would use their feedback to make necessary changes to the model. Perhaps I could discuss possible changes with them also, taking my restrictions into account (e.g. when it comes to allowed complexity, budget or time to make the model). Some other examples of similar techniques in this category are; experts could audit the simulation documentation or perform documentation consistency checking, which entails analyzing the completeness of the documentation, in order to uncover flaws in the simulation study.

An informal dynamic validation technique that could be used is a Turing test. In this test, we let the simulator generate results, which later are merged with system data. This data can then be examined by experts, in order to uncover if the data the simulator generates is similar to, or in accordance with, real system data. In this class of tests, we review both the structure of the model and the correctness of the data we use, i.e. we review the structural, simplifying and data assumptions in the simulation study. In this case, I would gather data at a supermarket during its original set up, i.e. with the number of cashiers, size of the store, number of aisles, etc. as given by the supermarket originally. Then I would set up the simulation model to mirror this system (a digital twin) and try to reproduce the data collected. When analyzing the generated data, I would use statistical methods to check if the distribution of the generated data matches the distribution of the collected data. Moreover, I would use experts in the field to review the data, in order to use their expertise and experience to conclude if it makes sense with real-life cases. Thus, with this type of methodology we could check if our model is implemented correctly to match the existing supermarket, in order to gain confidence in the model before changing the configurations in order to optimize it. In addition, the methodology allows checking data assumptions and simplifying hypotheses, by analyzing the data. Notice that this test becomes formal when using statistical methods (and is informal when simply asking experts for their opinions on the data). Thus, the methodology I have described here is a mix of an informal and formal validation technique. The formal part of the methodology I have described is highly reminiscent of Predictive validation – which is classified as a formal dynamic technique.

A formal static model validation technique could be Cause-Effect Graphing, or more specifically State Transition analysis. In this validation method one is focused on analyzing how the model changes from one state to another, depending on the inputs in each specific state. In this case, since we have both a Petri Net and SDL-model from Assignment 1, we can do this analysis by creating a reachability tree for the former or a reachability graph for the latter. The creation of these trees or graphs would be able to identify possible errors or shortcomings in the conceptual model. Notice that we in this case are focused on understanding if the model itself is correct, without taking any form of implementation or coding into account. Hence, this technique mainly provides insights on the structural assumptions.

The verification process can be performed via different formal techniques. In this process we are not focused on the conceptual model anymore, even though problems within the conceptual model also can be detected with these techniques, but we are focused on verifying that the implementation of the model is correct. Thus, this provides insights on the structural assumptions. One possible formal dynamic technique is Extreme Condition Testing, which is supposed to show that the model structure and outputs are credible even when using extreme and unlikely combinations of variables. In this case, I could for instance set the number of cashiers to a very large number, like 50, and have a look at the results. Moreover, I could set the processing time of a server to a very large value, like a mean of 100, or a very small value, like a mean of 0.2, and see if the results are as expected.

8 Results /Conclusions.

For conciseness we only tested four factors of two levels in this DOE. There exists several other factors one could test for this system, for instance the arrival time of the customers. Notice that this cannot be optimized by the store owner (at least not explicitly or directly), which makes it less interesting from an optimization standpoint, but it could still be interesting to learn how it would affect the proportion of successfully attended customers.

Since we were not able to get the implementation of the experiments in GPSS to work, we cannot conclude anything about the simulated system based on data simulated from it. However, some conclusions were made from the fabricated data, in order to highlight *how* conclusions can be made. When the correctly simulated data is gathered, similar conclusions can be made about the system of interest.