A computer-based educational tool for generating experimental design examples

Abstract

In this article, challenges in designing example problems of multifactorial experiments for an academic purpose are identified, and an algorithm for simulating multifactorial experiments is presented. It is based on a proposed mathematical function that mimics a physical system, can be employed for any number of finite factors, and contains a unique peak. The proposed algorithm is implemented on an application developed with JavaScript, HTML and CSS, which has been tested in the teaching of the Response Surface Methodology.

Keywords

Experimental design, educational tool, generating examples, problem-based learning, multifactorial model.

1. Introduction

Almost all the fields involving experimentation use experimental design [1]–[4]. It is part of various undergraduate and graduate curriculum, ranging from engineering to biological sciences. The objective of experimental design is to minimize cost and time of experiments and maximize its yield. As an example, it can be used to find the values of factors such as pH, oxygen concentration, sugar concentration, with which the enzyme production is maximized. Different techniques can be employed to find the minimum number of experiments [5] . On the other hand, an improper experimental design may lead to inaccurate or false conclusions, as well as a loss of money, material, and time [6].

Solving numerical examples helps students to learn statistics or mathematics [7] , or to develop insight into a particular topic [8] . Moreover, students obtain good learning experience using visual numerical examples of experiments [9] . On the other hand, solving optimization problems and finding the most accurate mathematical model for a process in experimental design involves performing various experiments with different combinations of factors. Usually, conducting experiments on a real system is not feasible due to any of the following limitations: 1) experiments on a real system can be costly, 2) a considerable amount of time may take each experiment, and 3) a great variety of experiments is difficult to be carried out. Hence, a computer program, that generates responses for some specific factors, is an excellent alternative to mimic physical systems, and to develop experimental design.

In this article, a computer-based educational tool for generating experimental design examples is proposed. It generates a unique process that takes into account factor limits selected by the user, and outputs experimental data for a given combinations of parameters. Teachers may adopt this educational tool in order to generate numerical examples, and to highlight all characteristics of experimental design. Moreover, the proposed educational tool allows teachers to evaluate the knowledge acquired by students during a course, and to implement problem-based learning. In this pedagogy, a student learns the topic while solving a problem given by the teacher. The core of the educational tool is a novel mathematical function that finds the optimal solution of an experiment. The core of the educational tool is a novel mathematical function that is designed to provide a unique peak of an experiment subjected to a number of finite factors, thus providing the optimal solution of an experiment.

The paper has the following structure. Section 2 describes a multi-response model. The proposed mathematical function, employed by the computer-education tool, is presented in Section 3. The implementation of the proposed function is mentioned in Section 4. Section 5 describes the algorithm that generates an experiment of a multi-factorial process. Section 6 shows an application used to implement this algorithm. The paper ends with some concluding remarks.

1. Multi-response mathematical model

A numerical example for an experimental design is a multi-response mathematical model representing a physical process. This model is a set of static functions, i.e., it does not have derivative or integral terms, which maps the factors to the responses, and it is given by



where ,  are responses, ,  are factors, ,  are nonlinear functions mapping the  factors to the  responses, and ,  are zero mean random noise. All factors, , are constrained by upper and lower limits. Numerical examples should produce unique optimal responses, , for a set of factors within its limits.

The algorithm developed in this paper uses a single response mathematical model, which is described by a proposed function presented in the next section. Several functions of this kind are used to mimic a multi-response system, since a set of  singe response systems can represent a multi-response system.

1. Proposed mathematical function

A static mathematical function with a single peak is achieved by any of the following functions.







Functions  and are a quadratic polynomial and a multivariable Gaussian function, respectively. Function is obtained by replacing the multiplication operator with a summation one. Figures 1, 2 and 3 respectively represent functions, , and  for a two variable case. These functions have the following limitations:

1. Function  has a property that its slope increases linearly as it moves far away from the optimal point, i.e., the slope at any point is directly proportional to the distance from its optimal point, which is not always true in the real life.
2. Response surface methodology uses a second order fit algorithm. Hence, reaching an optimal solution for a system based on  requires very less effort, which is not recommended as a practice problem.
3. The solution to , , and  does not contain the terms of other factors where . Multi factorial example problems based on these functions are easy to solve, because optimizing one factor after another factor will work in these cases.

Multiplying function  by  and making a change of variable produce the following proposed

mathematical function:



where ,,  and  are rows of the matrices  and  respectively, , , , and .

Figure 4 and 5 depicts  with ,  and ,  respectively. The gradient of  is given by



where  is a positive definite matrix. If  and are non-singular, then matrices  and are positive definite, respectively [10]. Moreover,  and are positive semidefinite when  and  are singular. Therefore, if either, or both are non-singular, then matrix in is positive definite, and as a consequence a unique peek at,  is guaranteed.

In contrast to function, the gradient of function in is not directly proportional to the distance from its optimum point. Moreover, has the following advantages with respect to,  and:

1. The optimal value of an arbitrary factor is not constant throughout the factorial space.
2. The solution  contains the terms of other factors, where.

In the next section, a method is presented to adapt the function, defined at , in order to generate random experiments.

1. Implementation of the proposed function

A scaled version of the proposed function  in is given by

























In these equations, ,  ,  is a random variable such that  and , where  denotes expected value. Moreover,  is a random variable with the properties  and. On the other hand  is the function range,  is a noise factor,  is difficulty factor,  are the optimal combination of factors, where function  reaches its maximum value,  and  are lower and upper limits of the  factor,  and  are padding constants that, respectively, limit the maximum value of the function  and the optimal combination within the desired region.

It is worth mentioning that function, shown in , preserves all the mathematical properties of function. Moreover, multiplying function  in by a negative number transforms the single-peak function to a single-dip (opposite to a peak) function; this transformation requires modifying and such that the upper and lower limits are preserved.

Next section presents an algorithm that implements the proposed procedure.

5. Algorithm

The objective of the algorithm is to generate experiment results by simulating a multi-factorial process. Teacher gives the parameters required by the multi-factorial process such as upper and lower limit of the factors, and students perform experiments at different factors. The algorithm generates responses for the given ,  inputs using the equations and inequalities shown from to . The student objective is to minimize the number of experiments conducted in order to achieve the optimum value of the process.

Figure 6 shows the flowchart of the proposed algorithm. The teacher should define the values of  and , where  and  should be less than 0.5. Based on our experience, It is recommended to use . Parameter in (14) is a difficulty factor, the bigger the value is assigned, the harder it is to reach the optimum value. It is recommended to use a value of . The noise factor in (7) introduces noise into the system. It is important to mention that this algorithm can be implemented in any programming language.

6. Application

The triad of JavaScript (JS), Hypertext markup language (HTML) and Cascading Style Sheets (CSS) is used to create the application presented at the web page [11]. JS is used to implement the algorithm discussed in the previous section.

Master students of the Biological Sciences Faculty, Universidad Autónoma de Coahuila, are instructed to use this application. Figure 10 shows the screenshot of the application. It is used by the students in the classroom, at homework, and in projects related to the learning of Response Surface Methodology (RSM). Students are asked to perform experiments on this application by opening it in the Google Chrome web browser with the objective of maximizing the enzyme production in a biological process with two factors, which are pH and temperature that takes values between 1 to 13 and 0 to 100 Celsius degrees, respectively. The application contains a distance tool to provide how far the optimal values are from the given factors, as shown in Fig. 7. The graphic user interface allows performing multiple experiments at once, where a student can paste factorial information and copy the model response to the clipboard. Once the students have estimated the optimal values of pH and temperature, these values are compared with the optimal ones in order to evaluate the student's performance. It is worth mentioning that the optimal values of pH and temperature are not shown in the graphic user interface.

As an example of an experiment, letand the factors pH and temperature, respectively, and let  the enzyme production. The lower and upper limits of these factor are,, , and as shown in Fig. 7. The scaled version of function , described in equations (7) to (18), uses the parameters:, , , , , , , , , and , , , , , , , , and . Figure 7 shows the surface plot of. Moreover, Figures 8 and 9 show, respectively, the contour plot and surface plot of the function  without the noise term i.e., considering. The Response Surface Methodology finds the optimal values of and, and the results of each iteration are superimposed over these plots.

7. Conclusions

This article presented a computed educational tool, designed in HTML, CSS, and JavaScript, in order to generate random processes with factors provided by a teacher. It is based on a novel single response, unique peak multivariable mathematical function, denoted as. This function mimics a physical process designed for experiments, is adapted to generate experimental data for a selected range of factors, and can generate a multiple response system using several functions. It was shown that has the advantage over three mathematical functions, named as,  and, that the optimal value of an arbitrary factor is not constant throughout the factorial space. Students that learn the topic of Response Surface Methodology use the proposed education tool. It motivates the interaction between professor and students, and allows teachers to implement the problem-based learning, or to assign a unique problem to each student. It also helps students to work in groups to discuss its functionality. Moreover, teachers may employ it to teach any other optimization techniques such as the Taguchi methods.

Acknowledgments

The authors thank the anonymous reviewers for their helpful comments.

References

[1] R. A. Fisher, *The design of experiments*. Oliver And Boyd; Edinburgh; London, 1937.

[2] G. P. Quinn and M. J. Keough, *Experimental Design and Data Analysis for Biologists*. Cambridge University Press, 2002.

[3] D. C. Montgomery, *Design and Analysis of Experiments*. John Wiley & Sons, 2008.

[4] J. Antony, *Design of Experiments for Engineers and Scientists*. Elsevier Science, 2014.

[5] A. M. Sarotti, R. A. Spanevello, and A. G. Suarez, “An efficient microwave-assisted green transformation of cellulose into levoglucosenone. Advantages of the use of an experimental design approach,” *Green Chem.*, vol. 9, no. 10, pp. 1137–1140, 2007.

[6] M. F. W. Festing, “Principles: The need for better experimental design,” *Trends Pharmacol. Sci.*, vol. 24, no. 7, pp. 341–345, 2003.

[7] X. Zhu and H. A. Simon, “Learning mathematics from examples and by doing,” *Cogn. Instr.*, vol. 4, no. 3, pp. 137–166, 1987.

[8] A. Renkl, “Learning from worked-out examples: A study on individual differences,” *Cogn. Sci.*, vol. 21, no. 1, pp. 1–29, 1997.

[9] J. Hattie and G. C. R. Yates, *Visible Learning and the Science of How We Learn*. Taylor & Francis, 2013.

[10] A. S. Poznyak, *Advanced mathematical tools for automatic control engineers. Volume 1, Deterministic techniques*. Elsevier, 2008.

[11] S. K. Gadi, “Multifactorial experiment simulator - Suresh Kumar Gadi,” 2017. [Online]. Available: https://skgadi.com/tools/multifactorial-experiment-simulator/. [Accessed: 27-Aug-2017].