

AUTOMATIC EVALUATION AND DATA GENERATION FOR ANALYTICAL CHEMISTRY INSTRUMENTAL ANALYSIS EXERCISES**Arsenio Muñoz de la Peña^{a,*}, David Muñoz de la Peña^b, María P. Godoy-Caballero^a, David González-Gómez^c, Fabio Gómez-Estern^d and Carlos Sánchez^b**^aDepartamento de Química Analítica, Facultad de Ciencias, Universidad de Extremadura, Avda. Elvas s/n, Badajoz, Spain^bDepartamento de Ingeniería de Sistemas y Automática, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Camino de los Descubrimientos, s/n, Sevilla, Spain^cDepartamento de Didáctica de las Ciencias Experimentales y de las Matemáticas, Universidad de Extremadura, Avda. de la Universidad s/n, Cáceres, Spain^dEscuela Técnica Superior de Ingeniería, Universidad Loyola Andalucía, C/ Energía Solar, 1. Edificios F y G, Sevilla, Spain

Recebido em 18/02/2014; aceito em 25/06/2014; publicado na web em 28/08/2014

In general, laboratory activities are costly in terms of time, space, and money. As such, the ability to provide realistically simulated laboratory data that enables students to practice data analysis techniques as a complementary activity would be expected to reduce these costs while opening up very interesting possibilities. In the present work, a novel methodology is presented for design of analytical chemistry instrumental analysis exercises that can be automatically personalized for each student and the results evaluated immediately. The proposed system provides each student with a different set of experimental data generated randomly while satisfying a set of constraints, rather than using data obtained from actual laboratory work. This allows the instructor to provide students with a set of practical problems to complement their regular laboratory work along with the corresponding feedback provided by the system's automatic evaluation process. To this end, the Goodle Grading Management System (GMS), an innovative web-based educational tool for automating the collection and assessment of practical exercises for engineering and scientific courses, was developed. The proposed methodology takes full advantage of the Goodle GMS fusion code architecture. The design of a particular exercise is provided ad hoc by the instructor and requires basic Matlab knowledge. The system has been employed with satisfactory results in several university courses. To demonstrate the automatic evaluation process, three exercises are presented in detail. The first exercise involves a linear regression analysis of data and the calculation of the quality parameters of an instrumental analysis method. The second and third exercises address two different comparison tests, a comparison test of the mean and a t-paired test.

Keywords: evaluation; numerical exercises; analytical chemistry.

INTRODUCTION

In the European higher education system,¹ instructors are encouraged to provide personalized attention to their students. However, instructors of university courses often spend a considerable amount of time and effort designing and grading numerical exercises for their students, rather than using this valuable time to discuss or reinforce theoretical content with them. In this context, information and communication technologies (ICTs) have had an increasing impact on education over the past few decades, and collaborative web tools of the so-called Web 2.0 variety, such as Moodle or Blackboard, are now common in educational institutions at all levels,²⁻⁴ including chemistry courses.⁵ In general, learning management systems are very effective in distributing material, collecting student assignments, and offering the opportunity to exchange knowledge between different members of the educational community through the use of forums, blogs, and similar tools.

Automatic evaluation is a challenging field that has been addressed by the academic community in multiple ways. Possibly, multiple-choice-questions (MCQs) are the most popular approach, implemented in most e-learning platforms.⁵⁻⁷ The use of MCQs in higher education is widespread^{8,9} and its convenience is currently under discussion.¹⁰ From our perspective, more sophisticated tools specific for evaluating elaborate projects from students are highly desirable. Little effort has been made to automatically grade complex

assignments, although some interesting examples in diverse fields are found in the literature.¹¹⁻¹³

However, in technical colleges, a great improvement would be made if the instructor were given the ability to automatically assign and grade more complex assignments, such as simulations of laboratory experiments, to complement their current educational curriculum. Several web-based systems using a computer algebra system (CAS) have been developed with Internet submission systems to provide a sophisticated mathematical mechanism with which to grade a student's work.¹⁴⁻¹⁷ In general, these systems are aimed at designing online courses based on a sequence of short, concise questions that may not be appropriate for general science courses and, in particular, chemistry courses. In our framework, an appropriate system is achieved by employing a novel code fusion-based evaluation platform named Goodle Grading Management System (GMS).¹⁸⁻²¹ Goodle GMS is a web-based innovative education tool designed for automating the collection, evaluation, and error detection of practical exercises assigned to students that cover a wide class of engineering and science courses. By employing a student/instructor-code execution architecture, the conceptual limits of multiple-choice tests have been significantly overcome. The design of a particular exercise must be provided ad hoc by the instructor and requires basic Matlab knowledge. The proposed system is also able to individually parameterize the exercises for each student and allows an instructor to implement innovative self-learning techniques to enable a student to continuously measure his knowledge base along the entire course length.

*e-mail: arsenio@unex.es

In a previous work,²¹ we applied Goodle GMS to grade experimental exercises in which student results depended on their laboratory work. In the present work, we extend those results and present a novel methodology for designing analytical chemistry instrumental analysis exercises that can be evaluated and personalized for each student automatically; that is, each student is given a different set of experimental data generated randomly while satisfying a set of constraints, rather than using data obtained from actual laboratory work. This allows the instructor to provide students with a set of practical problems to complement their laboratory work along with corresponding feedback from the system's evaluation process. To this end, we include an additional step in the process to generate the experimental data using a Matlab script.

The proposed methodology has been used to grade numerical exercises personalized for over 200 students in courses of the Environmental Analytical Chemistry subject during the first course of the Master of Science in Environmental Pollution and of the Advanced Analytical Chemistry course of the Chemistry Grade (Analytical Chemistry Department, University of Extremadura). In addition, three different exercises are presented in detail, a linear regression analysis and quality parameter calculations of an instrumental analysis method in addition to two separate comparison tests.

EXPERIMENTAL PART

Materials and methods

Goodle GMS

In this paper, we present an automatic web-based exercise evaluation and data generation methodology to propose and correct numerical analytical chemistry exercises. The main idea is that the instructor, following the code fusion approach of Goodle GMS,^{21,22} designs the exercise wording, evaluation code, and data generation code that will allow for the automation of the process. Developing the Matlab code to automatically evaluate a student's results for a particular problem requires an initial effort; however, one of the advantages of the proposed approach is the possibility of using the same evaluation code in different courses.

Goodle GMS is a web application that is executed on a dedicated server located at the Engineering College of the University of Seville. Both students and instructors access the server using a standard web browser. While the use of MCQs is probably the most widespread automatic evaluation technology, Goodle GMS is a completely different evaluation method, which is inspired by the core architecture illustrated in Figure 1. The key idea is that students submit the solutions to practical exercises by following a certain programming language syntax, including Matlab among other possibilities.

The student's code is executed in the server along with code designed by the instructor to evaluate the student's work. This framework is denoted as black-box evaluation. The evaluation code is generated by the instructor, which must satisfy some design constraints, in particular, the order in which the student and instructor code segments are executed/compiled and the way in which the grade and possible comments are to be generated so that the server can retrieve them and save them in the database. These design constraints leave freedom to the instructor to design complex exercises and their corresponding evaluation codes using all the possibilities that the chosen programming language provides, for example C code libraries or Matlab toolboxes.

As an example of Goodle GMS architecture, consider a simple mathematical problem in which the students must evaluate the solution to the equation $ax + b = 0$, where the parameters a and b are personalized for each student depending on the first and second digits

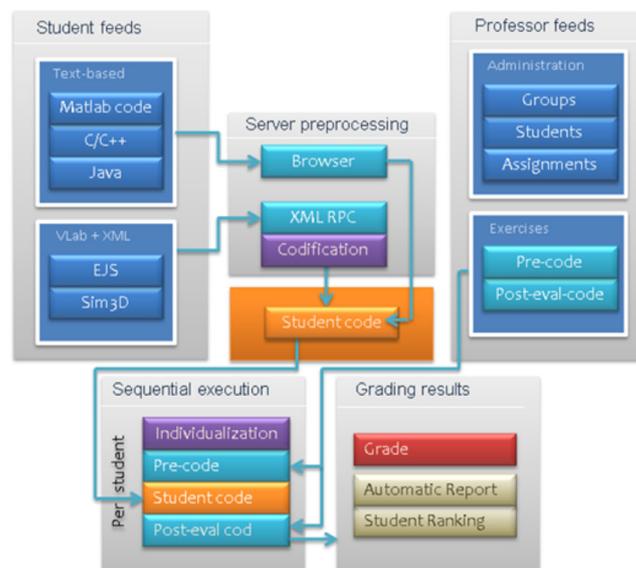


Figure 1. A schematic describing the workflow for both types of users and their interactions with the system

of their identification (ID) number, respectively. The student responds, for example, in accordance with the following single text line.

```
x = 34;
```

The instructor must then develop a Matlab code that is able to determine the parameters a and b from the ID number (which is provided automatically by Goodle GMS), solve the problem, and assess student responses in a workspace variable named `grade`. Assuming that a variable ID generated from a student's ID number is an integer vector, the instructor code would be expressed in the following format.

```
% ID is generated by Goodle GMS
a = ID(1);
b = ID(2);
x_sol = -b/2
if (x-x_sol) < 0.01 % Error margin
    grade = 1;
else
    grade = 0;
end
% Goodle GMS stores automatically the value of grade
```

Goodle GMS first executes the student's solution, generating a variable x in the workspace. It then generates in the workspace a variable `ID` using the ID number of the particular student being assessed, and the instructor code is executed. Finally, the value of the workspace variable `grade` is stored in the database for future reference.

In addition to numeric values, Goodle GMS also stores strings in the database. These strings can be generated by the instructor code to include comments and feedback in accordance with student results. In the examples presented in this paper, these strings provide information concerning the evaluation criteria and point out errors in student answers.

Goodle GMS is based mostly on open-source technologies. The infrastructure employed in the Engineering College at the University of Seville consists of an HP Proliant DL-140 rack server with 2 GB RAM and a 160 GB hard drive, and the Windows 2003 Server operating System, web server Apache 2.2.6, PHP 5, and MySQL database Server 5.0 are all installed on the same machine. The server also runs a copy of

Matlab 7.3, and the compiler Dev C++ GPL License C/C++ integrated development environment (IDE) and command-line compiler.

Using the proposed approach, it is possible to personalize the numerical parameters of the exercises as a function of the digits of a personal unique ID or enrolment number and to assess complex logical processes that bring together conclusions from different sets of data. Under these conditions, each student is given a unique problem to solve and they will be unable to share their solutions. This architecture requires an exchange between the database, containing the student ID number, and the evaluation code. In order to submit results, the student must log in using a security password field, which allows for the secure handling of the data. A detailed description of Goodle GMS can be found in Muñoz de la Peña *et al.*²⁰

This personalization approach has been applied with success in different courses in which problems can be personalized by modifying the numerical values of a finite set of parameters (such as the physical properties of an object or the initial conditions of a differential equation). In previous works, the personalization was obtained through generic exercise wording in which the student applied a numerical formula based on the student's ID number (for example, the mass of the object is equal to ten times the second digit of the ID number plus one). Each student applied the formula and solved their personalized exercise. Although this method can be used for multiple problems, it is not suitable to generate data for exercises that require experimental data obtained from measurements such as analytical chemistry exercises. In the following section, we propose to add an additional step in the workflow to allow for more complex exercise personalization.

Exercise wording personalized for each student

In the proposed modification, each student continues to receive from the instructor a generic exercise wording. The wording describes the particular analytical chemistry objective problem and the method used to solve it; however, the wording does not include the experimental data that will be used. Instead, the students receive from the instructor a program (a Matlab script) that each student must run at home using their individual ID number. This program randomly generates the set of data that the student must use in their personalized exercise. In particular, we have proposed to provide a Matlab script coded into a protected function file (–p file), which uses hidden code. This allows an easy integration with the code fusion method of Goodle GMS. In particular, the Matlab script generates an –html file that includes the student's data by means of a specific algorithm that takes into account the class of problem considered and imposes a particular structure; for example, in a linear regression problem, the samples are obtained from a linear function plus a stochastic normal distribution. It is important to remark that, for automatic evaluation of a student's solution in Goodle GMS, the server must be able to generate again the exact same random data set for each student. Therefore, while the data must be generated randomly to simulate the results of a given laboratory exercise, the algorithm must provide the same data for the same ID number at each iteration and must be repeatable in the Goodle GMS server. To this end, the data generation code is based on Matlab's random function using a fixed seed obtained for each student algebraically from their ID numbers. This guarantees repeatability and deterministic results for a given student. This method is the main contribution of this paper and is summarized in Figure 2. Note that, in this figure, the workflow has been simplified to take into account only the main elements in the proposed application (for example C or Java submissions are not considered). The main difference is that the data is not obtained from a mathematical formula, but from a program that must currently be executed by the students in a platform external to Goodle GMS. This workflow allows for more complex and sophisticated exercise personalization.

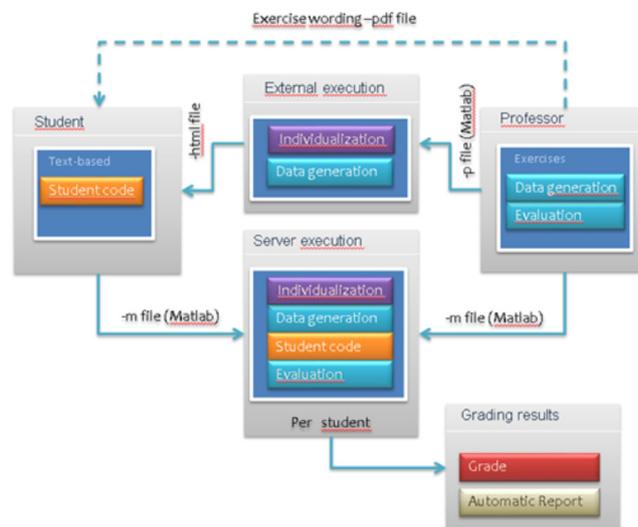


Figure 2. A schematic describing the proposed data generation and exercise evaluation workflow

Analytical chemistry exercises considered

Three different examples of exercise wording are described below to illustrate how the system can be employed to generate data and to provide automatic evaluation of exercises based on complex data or experimental results. In particular, these examples involve a linear regression analysis and calculation of the quality parameters of an analytical method, and two separate comparison tests.

- Statement of Exercise 1. To determine the concentration of molybdenum in a sample, an external standard calibration using a linear regression was performed using five different concentration levels of a standard solution and molecular luminescence as the instrumental signal. All measurements were performed in triplicate. A data generation script provides the numerical information. Students must determine the concentration of the unknown sample and the quality parameters of the analytical method, which include linearity, analytical sensitivity, and limit of detection.
- Statement of Exercise 2. Two methods were employed to analyze the total iron content of a siliceous rock. A data generation script provides the numerical information derived from both methods. Students must determine if there are significant differences between both methods at a confidence level of 95%, by performing a comparison test of the analysis results of both methods for the same sample.
- Statement of Exercise 3. A new method to determine antimony in urban atmospheric samples has been developed based on flame atomic absorption spectroscopy. To validate the performance of the new methodology, results are compared with those obtained by an official spectrophotometric method. A data generation script provides the numerical information derived from both methods. Students are asked to determine whether there are significant differences or not among both methodologies at a confidence level of 95% using a t-paired test of the results of the analysis of the two methods from different samples.

Theoretical considerations associated with each exercise

Exercise 1

The use of an unweighted linear regression for the establishment of a linear calibration curve involves the acceptance of four assumptions as follows.²³

- a) The Y-axis represents the instrumental response and the X-axis the concentration.

- b) Errors occur only in the instrumental signal measurements.
- c) The errors are normally distributed.
- d) The absolute errors of the instrumental signal measurements are constant and independent of the concentration of the standard solutions (homocedasticity).

We obtain an analytical signal for each standard. The next step is to fit these results to the following linear equation:

$$y = mx + b \quad (1)$$

where y is the analytical signal, x is the concentration, and b and m are the intercept and slope of the regression line, respectively. This relation is obtained using the statistical model of a linear regression (least squares). The regression standard deviation and the standard deviation of the slope and intercept can be also determined. The slope and intercept values are calculated according to

$$m = \frac{\sum_{i=1}^n [(x_i - \bar{x})(y_i - \bar{y})]}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2)$$

$$b = \bar{y} - m\bar{x} \quad (3)$$

where y_i represents the value of the instrumental signal obtained for a concentration x_i of standard analyte for a total of n data points used to calculate the regression line, and \bar{x} and \bar{y} are the mean values of the set of concentrations and signals, respectively, used in the calibration.

The standard deviation $s_{y/x}$ of the regression of y over x is calculated as follows:

$$s_{y/x} = \sqrt{\frac{1}{n-2} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4)$$

where \hat{y}_i is the value of the signal calculated from the regression equation, corresponding to a concentration x_i , and $s_{y/x}$ is a measure of the goodness of the fit of the experimental data to a line.

In the prediction of a problem sample using the external calibration method, the general expression to calculate the standard deviation of the prediction, in concentration units, is given by

$$s_x = \frac{s_{y/x}}{m} \sqrt{\left(\frac{1}{M} + \frac{1}{n}\right) + \frac{(y - \bar{y})^2}{m^2 \sum_{i=1}^n (x_i - \bar{x})^2}} \quad (5)$$

where M is the number of repetitions of the analysis of the problem sample, and y is the arithmetic mean of the signal values obtained from the M analysis.

A large number of statistical analytical parameters needed to express analytical results are commonly calculated by hand, for instance, figures of merit. To express the linearity of a regression analysis, the square of the correlation coefficient between x and y , r^2 , which is called the coefficient of determination, is frequently used:

$$r^2 = \frac{\sum (\hat{y}_i - \bar{y})^2}{\sum (y_i - \bar{y})^2} \quad (6)$$

The coefficient of determination expresses the proportion of the total variation that is explainable in terms of the regression line. If r^2

= 1, all observations perfectly fit a straight line, and, consequently, the total variation in y can be explained in terms of the regression line alone. If, on the other hand, $r^2 = 0$, there is no correlation at all between x and y . However, a regression line that runs parallel to the x -axis cannot explain any variation in y because it reflects the condition where y is a constant for all x .²⁴ Nevertheless, values of r^2 close to 1 do not necessarily imply a good straight-line relationship of the analytical data.²⁵ In the literature, an alternative expression of linearity involving the relation between the standard deviation of the slope s_m and the slope m , that is, the relative standard deviation of the slope expressed in %, has been proposed:²³

$$\text{linearity}(\%) = \left(1 - \frac{s_m}{m}\right) \times 100 \quad (7)$$

There are various definitions of the sensitivity of an analytical method. The International Union of Pure and Applied Chemistry (IUPAC) recommends a relationship between the instrumental signal and the analyte concentration. This can be measured from the regression analysis because the relationship between the standard error of the regression and the slope, expressed in concentration units, establishes the minimum concentration difference that is statistically discernible at any point on the calibration line. This parameter is known as the analytical sensitivity (or analytical resolution), which is given as follows:²³

$$\gamma^{-1} = \frac{s_{y/x}}{m} \quad (8)$$

The concept of the limit of detection (LOD) has been, and currently is, one of the most controversial in analytical chemistry due to the multiple definitions and calculation methods proposed for its determination. In this manuscript, we have introduced the concept established by the IUPAC, and two alternative methods are proposed, which overcome the weakness of the definition established in the original IUPAC definition. In general terms, the LOD of a method may be described as the concentration that yields an instrumental signal significantly different from a blank or background signal.²⁵ In the 1970s, the IUPAC²⁶ stated that the LOD, expressed as the concentration x_L , is derived from the smallest measure y_L , where $y_L = \bar{y}_{bL} + kS_{bL}$, \bar{y}_{bL} is the mean of the blank responses, S_{bL} is the standard deviation of the blank responses, and k is a constant. The detection limit x_L is obtained as $x_L = kS_{bL}/m$. The IUPAC strongly recommends the use of $k = 3$.

The IUPAC procedure for calculating the LOD has been discussed by many authors, and the different conceptions of the LOD have been reviewed by Boque and Heyden.²⁷ Long and Winefordner, in a critical work, proposed a modification of the IUPAC definition, based on the error propagation theory.²⁸ The error propagation theory takes into account errors in the values of m and b :

$$x_L = 3 \frac{\sqrt{\left(S_{bL}^2 + S_b^2 + \left(\frac{b}{m}\right)^2 S_m^2\right)}}{m} \quad (9)$$

Clayton *et al.*²⁹ stated in their work that traditional methods for determining detection limits are based on achieving protection against false-positive conclusions. A preferred approach is to define detection limits so that protection against both false positives and false negatives is assured. Clayton proposed the use of a parameter $\Delta(\alpha, \beta)$, rather than 3. This factor depends on α , β , and on the degrees of freedom ($n - 2$), where n is the total number of pairs of data used to build the

calibration line and α and β are related to the probabilities of false-positive and false-negative errors, respectively. A good approximation is when $\alpha = \beta = 0.05$. The LOD under these conditions is given as

$$x_L = \Delta(\alpha, \beta) \frac{S_{y_0}}{m} \quad (10)$$

where S_{y_0} is obtained by extrapolating the value of the standard deviation to a concentration of zero, using the signal standard deviation at the point y_0 :

$$S_{y_0} = S_{y/x} \sqrt{\frac{1}{M} + \frac{1}{n} + \frac{(-\bar{x})^2}{\sum_{i=1}^n (x_i - \bar{x})^2}} \quad (11)$$

Exercise 2

When reference materials for the evaluation of a method do not exist and the method must be validated by comparing its results with those obtained from a standard method, or when the results provided by two different laboratories are to be compared, it is necessary to perform a comparison between the means of the two result sets ($-X_1$ and $-X_2$).^{25,30} Such a comparison is performed using comparison tests, which are statistical tests used to compare the accuracy between two data sets by means of the probability that the null hypothesis (H_0) is not rejected. The null hypothesis states that the differences between the standard deviations and the means are only due to random errors. The test is carried out in two steps. First, the variances of the two methods are compared through the Fischer test or F-test. Then, depending on the results of this first test, the means are compared using two different procedures.

To carry out the F-test, the parameter F_{cal} is calculated according to the relationship

$$F_{cal} = \frac{S_1^2}{S_2^2} \quad (12)$$

where S_1^2 and S_2^2 are the variances of the two sets of results. Note that the larger variance of the two methods must be in the numerator because $F_{cal} > 1$. The value is compared with the tabulated value F_{tab} obtained from the statistical F-distribution for the significance level chosen (α) and the degrees of freedom ($n_1 - 1$) and ($n_2 - 1$), where n_1 and n_2 are the number of measurements performed using methods 1 and 2, respectively.

If $F_{cal} < F_{tab}$, there are no significant differences between the variances of the two sets of results. In this case, the next step consists of performing a t-test. If n_1 or n_2 are lower than 30, considering a Gaussian density function and that S_1^2 and S_2^2 are not significantly different, the parameter t_{cal} is calculated according to the following:

$$t_{cal} = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{S^2 \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}} \quad (13)$$

using S^2 , the so-called common variance, which is calculated as

$$S^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2} \quad (14)$$

where t_{cal} has $(n_1 + n_2 - 2)$ degrees of freedom. Finally, the obtained t_{cal} is compared with $t_{tab}(\alpha, (n_1 + n_2 - 2))$. If $t_{cal} < t_{tab}$, the null hypothesis is accepted because there are no significant differences between the two sets of results.

However, if $F_{cal} > F_{tab}$, it is not possible to calculate the common variance S^2 . In this case, the test used for comparison between the two sets of results is the t-test of Cochran in which t_{cal} is

$$t_{cal} = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{(S_1^2/n_1) + (S_2^2/n_2)}} \quad (15)$$

The degrees of freedom d.f. of t_{tab} are calculated according to

$$\text{d.f.} = \frac{\left(\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2} \right)^2}{\frac{(S_1^2/n_1)^2}{n_1 + 1} + \frac{(S_2^2/n_2)^2}{n_2 + 1}} \quad (16)$$

Then, if $t_{cal} < t_{tab}$, the null hypothesis is accepted and it can be concluded that there are no significant differences between the two sets of results for the chosen confidence level.

Exercise 3

In this exercise, students have to compare two methods from a study of samples containing different amounts of the analyte. The test for a comparison of the means is not adequate in this case because any variation due to the method will be affected by the differences between the compared samples.^{25,30} The null hypothesis established under these conditions states that significant differences between the concentrations given by the two methods do not exist. The hypothesis is not true if the mean of the differences is significantly different from zero. The t_{cal} equation used to compare the mean with a known value is

$$t_{cal} = \frac{(\bar{X} - \mu)\sqrt{n}}{S} \quad (17)$$

Where μ is the true value. Here, according to the null hypothesis, $\mu = 0$, and the above equation is expressed in the form

$$t_{cal} = \frac{(\bar{X}_d)\sqrt{n}}{S_d} \quad (18)$$

where $-\bar{X}_d$ and S_d represent the mean and standard deviation of the differences, respectively, and, for t_{cal} , the degrees of freedom are $n - 1$. Finally, if $t_{cal} < t_{tab}$ the null hypothesis is accepted and, therefore, the two method are not significantly different.

Evaluation code design

The results submitted by the students in text form can be executed as Matlab code to generate a set of variables in the workspace. Two types of Matlab code files are presented in this work. Encrypted Matlab files, also known as $-p$ files, are used to code the script needed to generate student data while blocking the possibility of altering the code. Matlab open files, also known as $-m$ files, are used to store the script needed to grade student results. The server executes this code and then the evaluation code designed by the instructor for the exercise. The evaluation code performs the calculation using a student's raw data to establish the correct answers. A different grading code must be used for each type of exercise. Note that the grading criteria can be easily modified by the instructor by modifying the evaluation code. The evaluation code not only provides a grade, but can also provide students with a detailed report of their results. The evaluation code generates a string named *comments* using Matlab string functions *strcat* and *sprintf*. These functions allow generation of strings with

information that depends on workspace variables and the merging of strings to obtain a single concatenated string. In the code, a new line is added after evaluation of a result to keep track of the feedback. The string stored in the database must be formatted using html tags. In particular, in the example provided, only the newline html tag is used. The comments are displayed in the web server (thus, the necessity of using html code). The evaluation code used for each one of the three exercises discussed is available as discussed in the Supporting Information.

RESULTS AND DISCUSSION

Exercise 1

The first part of the work consists of generating the working data using an automatic system in the Matlab environment. The concentrations and data signals of the employed standards, as well as the analytical samples are personalized for each student based in their ID number. The measurements for each student are obtained from a linear regression using a random slope and bias, and adding a 5% measurement error to both the standard and the problem measurements. The standard concentrations were fixed from 0.2 to 3.2. The problem concentration was chosen randomly between 0.5 and 3. This process is carried out using the Matlab script, `exercise1_data.p`, which can be downloaded by the students from the web page of the subject (Moodle). The code of this script can be obtained as described in the Supporting Information. This script must be run and provides an `-html` file with the necessary data that works as a function of the ID number of the student. As an example, for an ID number 80071067, the script creates a file `80071067.html` containing the following data.

```
EXERCISE DATA
ID number 80071067

The three measures for the standard 1 (x = 0.2) are:
y1 = 4.25154, y2 = 3.93557, y3 = 4.30114

The three measures for the standard 2 (x = 0.4) are:
y1 = 4.2702, y2 = 4.21205, y3 = 4.25277

The three measures for the standard 3 (x = 0.8) are:
y1 = 4.73237, y2 = 4.96964, y3 = 4.78168

The three measures for the standard 4 (x = 1.6) are:
y1 = 5.46605, y2 = 5.5696, y3 = 5.4491

The three measures for the standard 5 (x = 3.2) are:
y1 = 7.45906, y2 = 7.40427, y3 = 7.12714

The three measures of the sample are:
y1 = 4.65451, y2 = 4.91009, y3 = 4.89413
```

The proposed problem is solved according to these data. The student must use the ACOC software,^{31–33} to obtain the calibration curve parameters and the analytical figures of merit, as well as the calculated concentration with its corresponding uncertainty. The results are uploaded to Goodle GMS. To this end, each student must register on the server with his/her identification number and a password.

The students upload the linearity (`lin`), the analytical sensitivity (`res`), the LOD according to the procedure developed by Long and Winefordner (LODWL),²⁸ and the Clayton (LODC) criteria,²⁹ and, finally, the calculated concentration of the molybdenum in the problem

sample (x_p), as well as the associated uncertainty (e_p). The uncertainty is given by $t \cdot s_x$ for a t-Student value corresponding to $n - 2$ degrees of freedom and a significance level of 95%. The students must give the answer with a single uncertain digit. The results are submitted in text form with the following structure.

```
Quality_parameters = [lin res LODWL LODC];
Concentration = [xp ep];
```

To facilitate uploading of the answers by the students, the server provides a template with an adequate format. An example of the Goodle GMS submission is shown in Figure 3.



Figure 3. An example Goodle GMS results submission page

The grade of the student is calculated by comparing their solution to the correct solution obtained using Matlab. The maximum grade is 10. If the real concentration is inside the range of values estimated by the student, the student receives 2.5 points. If the associated uncertainty is correct, the student receives 2.5 points. For each correct quality parameter, the student receives 1.25 points.

Exercise 2

In this exercise the students compare the results provided by two different methods according to the theoretical indications described above. As in the previous exercise, each student works with individualized data according to their ID number. Using the same ID example, the student will obtain, after running the generator Matlab script, `exercise2_data.p`, an `-html` file containing numerical information. In this case, the data generation was designed in such a way that the solutions of the resulting exercises were distributed into the four different possible cases reflecting whether or not the mean or the standard deviation showed significant differences. Each student had a different set of data distributed between maximum and minimum values, and was randomly assigned one of the four cases. Depending on the case, the two sets of data were generated using different errors and mean values. The particular values for each case were tuned by extensive simulation until the distribution was acceptable.

As an example, for an ID number 80071067, the script creates a file `80071067.html` comprising the following data.

```
NUMERICAL DATA FOR THE EXERCISE
ID 80071067
Method A: Atomic absorption spectroscopy
Fe2O3 (percentage)
36.8961
```

```

36.6263
36.1783
34.7433
35.8801
Method B: Molecular spectroscopy
Fe2O3 (percentage)
36.1608
36.0275
36.1827
35.7914
35.8459
36.1211
36.2558
35.8745
35.8983

```

With this information, students must apply the appropriate statistical test and submit the results to Google GMS. In particular, they must submit the calculated and tabulated F-values (F_{cal} and F_{tab}) and t-values (t_{cal} and t_{tab}), and the results of the comparison. In this case, the student must indicate the presence of significant differences with a number 1 and the absence with a 0. Thus, the results are submitted in text form with a structure given by the following.

```

test_parameters = [Fcal Ftab Tcal Ttab];
results = res;

```

For instance, for the ID case listed above the results will be as follows.

```

test_parameters = [24.2764 5.0530 0.1249 4.3030];
results = 0;

```

If the results and the parameters are correct the grade is 10, else the grade is 0.

Exercise 3

The aim of this exercise is to validate a newly established analytical method by comparing the results provided by the new method (Method A) with those obtained by a reference method (Method B). To solve this exercise, students must apply the equations described above, keeping in mind that, in this case, they are to compare two methods by means of the study of samples containing different amounts of the analyte. As in the previous exercise, each student works with individualized data according to their ID number following the same procedure as mentioned above. In this case, the data conforms to only two possible cases depending on whether or not there are significant differences. The parameter design was also verified by extensive simulation, as was the case for Exercise 2.

Using the same ID example, the student will obtain, after running the generator Matlab script, exercise3_data.p, an -html file containing numerical information. As an example, for an ID number 80071067, the script creates a file 80071067.html comprising the following data.

```

NUMERICAL DATA FOR THE EXERCISE
ID 80071067

Method A
New method
Antimony (mg/m3)
86.22

```

```

40.48
37.15
32.49
66.75
20.47
66.96
44.35
54.63

Method B
Standard official method
Antimony (mg/m3)
83.62
45.05
39.77
37.11
72.21
22.84
68.59
47.45
52.51

```

With this information students must apply the appropriate statistical test and submit the results to Google GMS. Students submit the calculated (t_{cal}) and tabulated (t_{tab}) t-values, and the results of the comparison (res) indicating whether or not there are differences between methods A and B. As in the previous example, the student must indicate the presence of significant differences with a number 1 and the absence with a 0. Thus, the results are submitted in text form with the following structure.

```

test_parameters = [Tcal Ttab];
results = res;

```

For instance, for the ID case listed above, the results will be given by the following.

```

test_parameters = [2.2954 2.3060];
results = 0;

```

If the results and the parameters are correct, the grade is 10, else the grade is 0.

Post-task analysis

Student opinions were sought concerning the suitability of the application and their degree of satisfaction by means of a post-task survey. The post-task survey was administered to 40 students from a course of analytical chemistry, in December 2012. A total of five questions were included in the survey wherein the students were asked to agree or disagree with a particular statement. The survey questions were as follows:

Q1. Using a webpage to submit the results of practical exercises is helpful;

Q2. Using a server to automatically grade the numerical exercises of a course is helpful;

Q3. Google GMS is easy to use;

Q4. Automatic evaluation may improve the learning process because it can provide a continuous assessment;

Q5. In general, I was satisfied with the course.

According to the results summarized in Figure 4, where NA refers to "not answered", SD is "strongly disagree", D is "disagree", A is "agree", and SA is "strongly agree", a high degree of satisfaction

can be noted regarding the prototype and the simplicity of its use. Over 80% of the students are in favor of the use of a webpage to submit their exercise results. This is not surprising, considering that young people are the main users of the internet applications. Note that 100% of the students found the use of the application easy. With respect to the use of a server to automatically grade their work, around 70% were in favor; therefore, while one third of the students disagreed, the use of this new technology was widely accepted. Student acceptance of the numerical grades provided by the server is one of the most important issues with Goodle GMS. It should be stressed that the exercises were used to complement a series of laboratory exercises and exams, not to supplant them. The use of Goodle GMS as a tool to provide continuous assessment was also very highly valued, with over 80% in favor. Finally, all the students were satisfied with the course in general, which implies that they accepted the use of Goodle GMS as part of the evaluation procedure.

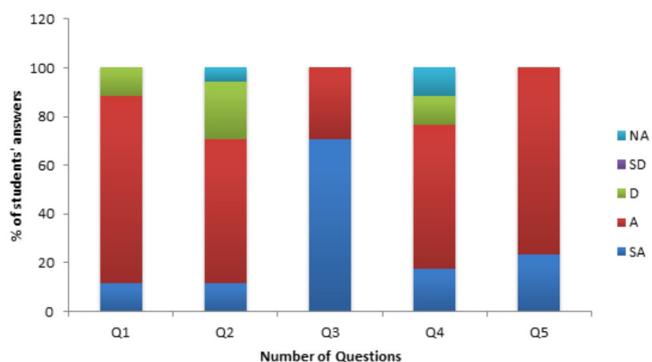


Figure 4. Student satisfaction evaluated through the post-task survey, where the results are expressed as % of student answers according to the following scale: SA: strongly agree; A: agree; D: disagree; SD: strongly disagree; NA: not answered

CONCLUSION

The methodology presented in this manuscript allows instructors to design personalized, advanced analytical chemistry exercises, which students can carry out in an autonomous way, using the Goodle GMS web-based system. The main contribution of this work is to present a methodology that allows for complex personalization, which is particularly well suited to generating data sets that simulate the laboratory work of analytical chemistry numerical exercises.

In addition, the system has been designed in such a way that the students obtain an evaluation of their work directly after the submission owing to the automatic correction script that is implemented in Goodle GMS. This allows students to reinforce those concepts for which they might have had more difficulties while solving the exercises. Another advantage is that, because the exercises are personalized, students cannot share their solutions, and, therefore, the instructor has good surety of the authenticity of the submitted results. It is also important to highlight that the instructor has complete control over the grading process because it is possible to modify the code according to the instructor's personal objectives owing to the fact that it can be programmed in Matlab. The application has been tested with different groups of students at the University of Extremadura during three different years, with satisfactory results.

Future work includes the modification of Goodle GMS to account for a preprocessing step in which the server automatically generates a personalized wording for each student, thereby avoiding the use of an external execution step in the workflow.

SUPPLEMENTARY MATERIAL

All Matlab files required to generate the numerical data for the exercises and to grade student responses are provided as supplementary material (free access at the Química Nova website (<http://quimicanova.sbq.org.br>)). Should the reader be interested in testing and using the application by freely creating auto-evaluated exercises (or using the code presented in this work), and assigning them to their students, it is possible to do so on a fully functional application running on a publicly available server. Please contact the authors by email to gain password access.

ACKNOWLEDGMENTS

The financial support of Servicio de Orientación y Formación Docente del Vicerrectorado de Calidad e Infraestructuras de la Universidad de Extremadura (Convocatoria de Ayudas para la mejora de la calidad docente Cursos 2012-2013 y 2013-2014, Projects A2_2013_90 and B_2014_27), Ministerio de Economía y Competitividad of Spain (Project CTQ2011-25388), and Gobierno de Extremadura (Project GR-10033, Research Group FQM003), both co-financed by European FEDER funds, is acknowledged.

REFERENCES

- van der Wende, M. C.; *Higher Education in Europe* **2000**, 25, 305.
- Poindexter, S. E.; Heck, B. S.; *IEEE Control Systems Magazine* **1999**, 19, 83.
- Kapur, S.; Stillman, G.; *Innovations in Education & Training International* **1997**, 34, 316.
- Heck, B. S.; Poindexter, S. E.; Garcia, R.; *Proceedings of the 2000 American Control Conference, Chicago, Illinois, USA, 2000*.
- Vician, C.; Charlesworth, P.; *J. Chem. Educ.* **2003**, 80, 1333.
- Penn, J. H.; Nedeff, V. M.; Gozdzik, G.; *J. Chem. Educ.* **2000**, 77, 227.
- Stewart, B.; Kirk, R.; LaBrecque, D.; Amar, F. G.; Bruce, M. R. M.; *J. Chem. Educ.* **2006**, 83, 494.
- Petridis, V.; Kazarlis, S.; Kaburlasos, V. G.; *IEEE Trans. Educ.* **2003**, 46, 102.
- Tartaglia, A.; Tresso, E.; *IEEE Trans. Educ.* **2002**, 45, 268.
- Gentil, S.; Exel, M.; *Proceedings of 2nd IFAC Conference on IBCE, Grenoble, France, 2004*.
- HuaQiang, J.; Liang, Z.; WangQiong, Y.; *International Conference on Computational Intelligence and Software Engineering, Wuhan, China, 2009*.
- Gerosa, M.; Narayanan, S.; *IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, Nevada, USA, 2008*.
- Chiou, C.-K.; Hwang, G.-J.; Tseng, J. C. R.; *Computers & Education* **2009**, 53, 261.
- Klai, S.; Kolokolnikov, T.; van den Bergh, N.; *International Workshop on Advanced Learning Technologies, Palmerston North, New Zealand, 2000*.
- Sangwin, C.; *Teaching Mathematics and its Applications* **2004**, 23, 1.
- Mavrikis, M.; Maciocia, A.; *Workshop of Advanced Technologies for Mathematics Education in 11th International Conference on Artificial Intelligence in Education, Sydney, Australia, 2003*, <http://www.lkl.ac.uk/manolis/pubs/pdfs/aied-wallis-03.pdf>.
- Olivier, G. W. J.; Herson, K.; Sosabowski, M. H.; *J. Chem. Educ.* **2001**, 78, 1699.
- López-Martínez, M.; Gómez-Estern, F.; Muñoz de la Peña, D.; *International Journal for Knowledge, Science and Technology* **2010**, 1, 1.
- Gómez-Estern, F.; Muñoz de la Peña, D.; *8th IFAC Symposium on Advances in Control Education, Kumamoto, Japan, 2009*.
- Muñoz de la Peña, D.; Gómez-Estern, F.; Dormido, S.; *Computers & Education* **2012**, 59, 535.

21. Muñoz de la Peña, A.; González-Gómez, D.; Muñoz de la Peña, D.; Gómez-Estern, F.; Sánchez Sequedo, M.; *J. Chem. Educ.* **2013**, *90*, 308.
22. Gómez-Estern, F.; López-Martínez, M.; Muñoz de la Peña, D.; *Revista Iberoamericana de Automática e Informática Industrial - RIAI* **2010**, *7*, 111.
23. Cuadros-Rodríguez, L.; García Campaña, A. M.; Jiménez Linares, C.; Román Ceba, M.; *Anal. Lett.* **1993**, *26*, 1243.
24. Massart, D. L.; Vandeginste, B. G. M.; Buydens, L. M. C.; De Jong, S.; Lewi, P. J.; Smeyers-Verbeke, J.; *Handbook of Chemometrics and Qualimetrics: Part A*, Elsevier: Amsterdam, 1998.
25. Miller, N. J.; Miller, J. C.; *Statistics and Chemometrics for Analytical Chemistry*, 6th ed.; Pearson: Essex, 2009.
26. MacDougall, D.; Crummett, W. B.; *Anal. Chem.* **1980**, *52*, 2242.
27. Boque, R.; Heyden, Y. V.; *LC-GC Eur.* **2009**, *22*, 82.
28. Long, G. L.; Winefordner, J. D.; *Anal. Chem.* **1983**, *55*, 712A.
29. Clayton, C. A.; Hines, J. W.; Elkins, P. D.; *Anal. Chem.* **1987**, *59*, 2506.
30. Ramis-Ramos, G.; García Alvarez-Coque, M. C.; *Quimiometría, Síntesis*: Madrid, 2001.
31. Espinosa-Mansilla, A.; Muñoz de la Peña, A.; González-Gómez, D.; *Chem. Educ.* **2005**, *10*, 337.
32. Galeano Díaz, T.; Muñoz de la Peña, A.; Espinosa-Mansilla, A.; Durán Martín Merás, I.; Acedo Valenzuela, M. I.; Cañada Cañada, F.; González-Gómez, D.; *ACOC v2.0, Herramienta estadística para Química Analítica*, Servicio de Publicaciones Universidad de Extremadura: Cáceres, 2006.
33. González-Gómez, D.; Gallego Picó, A.; Muñoz de la Peña, A.; Jeong, J. S.; *Chem. Educ.* **2013**, *18*, 136.