USING WEB-BASED TOOLS FOR TEACHING STATISTICAL CONCEPTS AND EXPERIMENTATION SKILLS

P. Darius[(1)], E. Schrevens[(1)], H. van der Knaap[(2)], K. Portier[(3)], G. Massonnet[(1)], L. Lievens[(1)], L. Duchateau[(4)], O. Thas [(4)]
[(1)] K.U. Leuven, Belgium [(2)] Unilever Research, Vlaardingen, The Netherlands
[(3)] University of Florida, Gainesville, Florida, USA [(4)] Ghent University, Belgium

This paper describes a number of generally accessible web-based tools the authors developed over the last couple of years and used in several statistics courses, ranging from the introductory statistics course to a specialized design of experiments course. The first set of tools (VESTAC) illustrates statistical concepts, the second one (VIRTEX) consists of virtual experiments. We discuss some of our experiences with regard to the development and maintenance of these tools and their use in ex cathedra teaching sessions, in guided practice sessions and in student projects. Finally we discuss a third type of tools: distance experiments.

INTRODUCTION

Teachers of statistics courses are confronted with a multitude of didactic problems. In the introductory statistics course, the basic statistical concepts have to be introduced, and some students experience part of the material as very abstract and have difficulty mastering the key concepts. Follow-up courses need to prepare students not only to analyze the data, but also to collect data in an efficient and appropriate way. It has been argued that supplementing the traditional material with tools based on a visual approach and a more active form of learning could improve the effectiveness of the teaching (Anderson-Cook and Dorai-Rai, 2001; Cobb, 1992; Moore, 1997; Marasinghe et al., 1996).

There are currently many computer-based tools available for teaching statistics. Those among them that can be used over the internet are particularly appealing from a teacher's point of view. Such tools range from complete interactive textbooks over statistical computing packages to interactive applications covering one or more topics. They vary greatly in quality, scope and accessibility.

In this paper we describe two collections of tools we developed. They are freely accessible over the web, and address the teaching problems raised above. VESTAC (http://www.kuleuven.ac.be/ucs/java) (Darius et al, 2000, Darius et al, 2002) visualizes a number of statistical concepts and allows the user to experiment with them interactively. VIRTEX (http://www.kuleuven.ac.be/ucs/virtex) (see also Darius et al, 2003) provides software environments that mimic an experimental situation of interest, and invite the user to collect data to answer a research question. Selected items from both collections can be suitably included in introductory statistics courses and/or in a second regression/anova/experimental design course. This can take the form of an in-class demonstration, a guided exercise or a student project.

VESTAC

The VESTAC (Visualization of and Experimentation with STAtistical Concepts) collection currently consists of 32 applets. They cover selected topics from the following four areas: distributions and plots, tests and confidence intervals, regression and analysis of variance.

The distribution applets visualize the form and relevant tail areas of the usual univariate distributions and the bivariate normal. They also show how these change when the parameters are changed, allowing them to be used as a visual statistical table. The central limit theorem is illustrated by showing how the mean of samples from different distributions gradually approaches a normal distribution. Other applets let the user experiment with histograms, boxplots and QQ-plots, and visualize the concept of correlation.

The second group of applets deals with the concept of a confidence interval, illustrates one sample, two sample and some non-parametric tests and visualizes power and type I and type II errors.

The regression applets deal with the relation between population and sample, and show how the variation of the sample regression line is influenced by the choice of the x-values. It is also shown how fitting a curve according to the least squares criterion relates to finding the minimum of an SSerror surface, in the linear as well as in the non-linear case. The distributions of MSregression, MSerror and their quotient are built up through repeated sampling. The correlation between the estimators for slope and intercept is shown. Confidence bands and prediction intervals are illustrated, residual and normal probability plots, as well as the effect of influential points and outliers. The difference between regressing y on x and x on y is shown (in all these cases the data points can be dragged with the mouse and the effect immediately seen). Other applets illustrate the Box-Cox transformation and the effect of near-multicollinearity on the confidence intervals of the parameters. Finally, a couple of applets visualize permutation tests for small datasets.

The anova applets illustrate the relation between population and sample, and allow to see the effect of changes in the data on residual and normal probability plots. Another applet shows how the distributions of MStreatment, MSerror and their quotient are built up, when the usual null hypothesis is true, as well as in the case of a user specified alternative hypothesis being true. There is also an applet that shows the mechanism of the permutation test and visualizes the exact randomization distribution. Finally, an applet illustrates the danger of using t-tests for testing hypotheses generated by data snooping. It shows both the distribution of pairwise differences between sample group means under repeated sampling, and the distribution of the difference between the largest and the smallest sample group mean.

EXPERIMENTATION FEATURES IN VESTAC

Every applet gives a visual representation and allows interactive experimentation. Considerable effort has been made to give the applets, as much as possible, a common "look and feel".

Wherever applicable, the "Step", "Walk" and "Run" buttons implement repeated sampling. The "Step" button generates one new sample. The main results of the previous samples (if any) remain visible on the screen. The "Walk" button generates one sample after the other until a specified limit is reached, and does this at a slow rate, allowing the user to see how the final result is built up. The "Run" button generates new samples at the fastest possible rate. Fig. 1 (left) illustrates this for the applet visualizing the variability of the sample regression line. It shows the situation after hitting "Step" five times. Shown are: the population regression line (the dark line with intercept 20.0), and (in a different colour, but also dark in Fig 1) the current sample points and the fitted line. The gray lines are the lines fitted on the previous samples.

An applet originally appears in a separate window, with default parameter settings applied. This makes it possible to start the applet demonstration quickly (e.g. while teaching in a classroom). The visual elements (e.g. the points on a scatterplot) can, wherever applicable, be manipulated directly by dragging with the mouse. The most important parameters (e.g. the degrees of freedom of a distribution) can be adjusted directly on the screen. All the other parameters can be changed through a pop-up window. The applets not only execute simulations based on settings of population parameters, those for which it is relevant also allow import of an actual dataset. This makes it possible for the applets to use the same datasets as e.g. those in the textbook. However, due to security measures in Internet browsers, this is only possible with datasets stored on the web server.

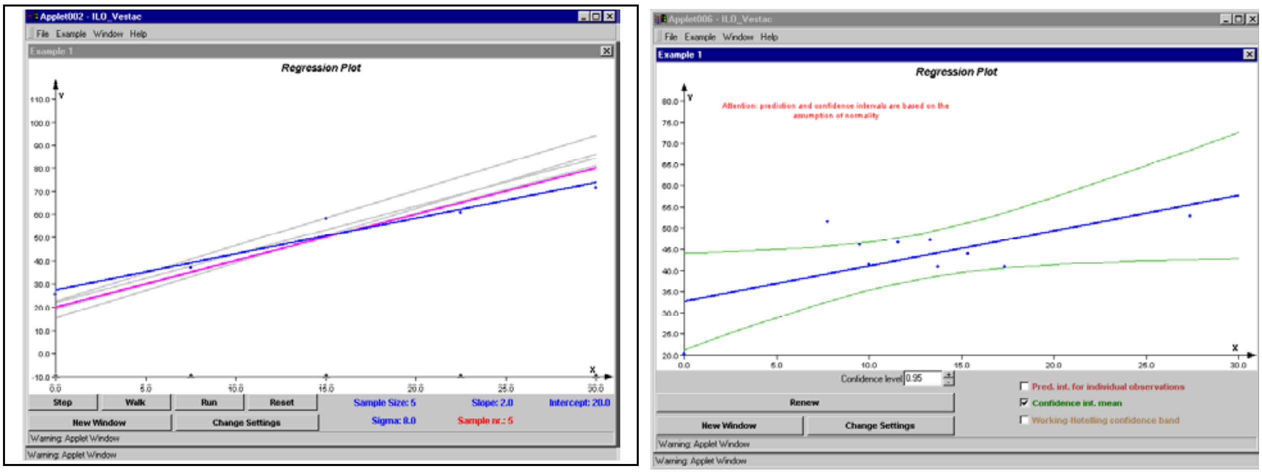

Fig 1.Variability of the sample regression line, as illustrated by repeated sampling from a population (left). Sensitivity of the fitted line and the associated prediction/confidence bands for the location of particular points (right).

Fig 1 (right) hints at the available possibilities for interactive experimentation. This applet shows a set of points (either obtained by random sampling from a specified population, or from an available dataset), with the fitted regression line and/or prediction/confidence bands. Each of the points can be dragged around with the mouse, and both line and bands are instantly adjusted to the new position.

The most interesting learning experience occurs when two or more parameter settings can be compared side by side. To make this possible, all the applets have the following features: upon hitting a “new window” button, a new window appears and the previous windows are resized, so that they are all visible on the screen. Each window now behaves as a separate applet: parameters can be changed on each and computations in each window occur concurrently.

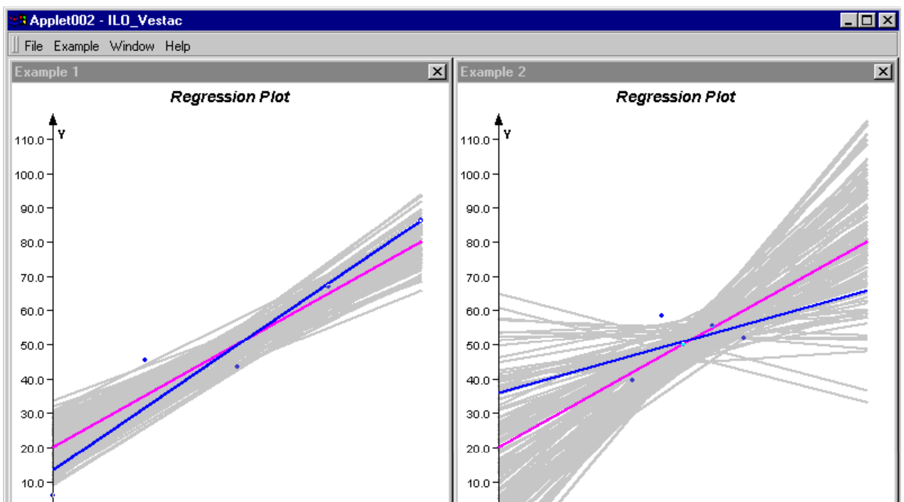

Fig 2. Comparing the variability of the fitted line, for samples from the same population but with different x-values (see detailed explanation in the text).

Fig 2 shows two windows with the same applet (the one from Fig 1 left). It illustrates how the x-values of the sample points influence the variability of the fitted line. In both windows, sampling is from the same population. But in each window points are sampled from different x-locations. The user can specify these by dragging the little triangles on the x-axis to the appropriate positions. As in Fig 1, the screen shows the population regression line, the last sample and its fitted regression line, and (in gray) all the lines fitted on the previous samples.

USING VESTAC IN TEACHING

The VESTAC applets are currently used in a variety of courses at the authors' universities, as well as in a number of other institutions. We did not conduct a formal study of student reactions. Informal contacts with colleagues who used the applets said that the applets made their courses livelier, that they made it easier to introduce abstract concepts and that the concepts introduced visually made a larger impact. It was remarked that the nature of the questions students asked changed after they used the applets: then their questions were less concerned with the concept itself, but with things like variability, departures from model assumptions and sample size issues.

In our courses we use the applets not only in class, but also during the PC exercises: a student is given a set of questions which he/she has to answer with the help of the applets. We located several web sites were the VESTAC applets were used in a similar way.

VIRTEX

The ability to design experiments in an appropriate and efficient way is an important skill, but students typically have little opportunity to get experience. Most textbooks introduce standard general-purpose designs, and then proceed with the analysis of data already collected. Some recent textbooks (e.g. Cobb 2002, Dean and Voss 1998) stress the importance of including projects in which the students actually have to prepare, perform and analyze a real experiment. Such projects provide an invaluable experience, but are very time and resource consuming, for the student as well as for the teacher. An alternative is to use the computer to "perform" the experiment (Anderson-Cook et al 2001).

Here we describe VIRTEX (VIRTual EXPerimentation), a collection of "virtual experiments". These are software environments, which mimic a real situation of interest, and invite the user to collect data to answer a research question. The data are generated by an underlying realistic stochastic model, invisible to the user. Once the data are collected, they can be transferred to a standard statistical package. The user can train his/her design skills by relating the quality of the statistical results obtained to the data collection strategy used.

SAMPLING APPLETS

The first applet introduces simple sampling problems. The user sees a window, showing part of a large (50x100) array of candies with different colors. The purpose is to get an idea of the percentage of candies that are colored red (fig 3 left).

When the user moves the mouse over a candy, its coordinates appear. With a left click, a candy is selected and the counters for the number of candies selected and the number of red candies are incremented.

Since there are so many candies, simply counting the number of reds is not an option. The user will have to use some form of sampling. Hence he/she will have to decide on a sample size and a sampling scheme (e.g. random or stratified sampling), but will discover that each of those requires careful planning before it becomes operational.

In the first version, patches are clearly visible where one color occurs more frequently. In the second version, the candies are wrapped in black and the color of a candy is only revealed after it has been included in the sample.

The second applet represents an environmental sampling problem. The user sees an aerial photograph of a section of a railway track (fig 3 right). It is explained that for many years a diesel train uses this place to take a halt. Hence diesel pollution of the site is suspected, and should be investigated.

The user is invited to take a set of soil samples (by clicking the mouse on the selected spot). Since there is a limited budget and analysis is expensive, at most 20 samples can be taken. The results of the analyses are shown in a separate window, from where they can be cut and pasted into e.g. a spreadsheet or a statistical package.

Again the user is confronted with a sampling problem, but now simple random sampling will not work. The user will have to set up a sampling strategy that takes the context (the layout of the site, the fact that part of it is covered by water and hence cannot be sampled,..) into account.

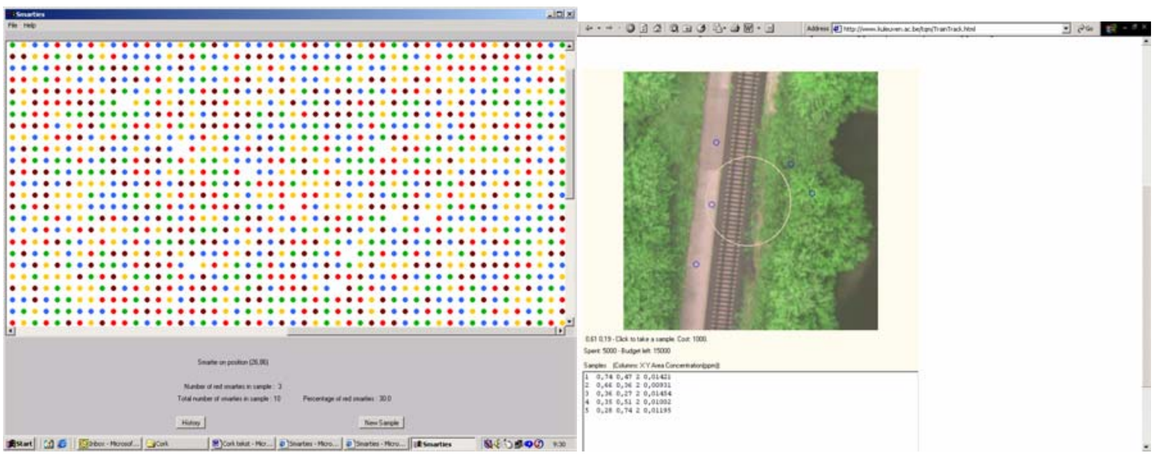
Fig 3. The candy applet (left) and the environmental sampling applet (right).

THE FACTORY APPLET

In the factory applet, the user has to experiment with a pilot plant in order to find optimized settings for the parameters of a production process. The experiment runs in “real” time: the user has 39 “weeks” (one “week” is about 3 minutes of real time) to complete the experiment. Each experimental run takes a number of "days" before the results are visible.

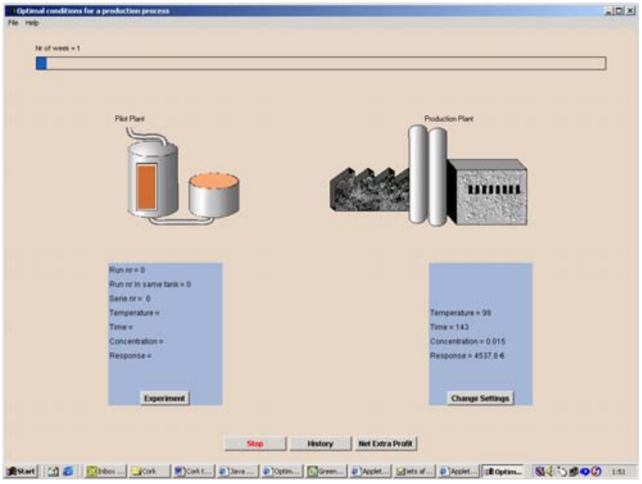

Fig 4. The factory applet.

The production plant appears at the right hand side. The current settings for the parameters temperature, time and concentration are shown, as well as the current mean yield per batch. The left hand side shows the pilot plant. This is a scaled down version of the production process with (hopefully) similar characteristics. The raw material for the pilot plant is stored in a tank, which can contain enough material for 10 trials. When the tank is empty (or upon request of the user) it will be refilled, but the new raw material may have slightly different characteristics.

The user can set up an experiment by opening a separate window and filling in the parameter settings for any number of runs (as well as instructions for handling the tank), or by using an external package to generate a design and pasting it in. After waiting an appropriate amount of "time", the results become available in the "History" window. This also shows the results of all previous experiments. Data from this window can be copied and pasted into an external program for statistical analysis.

The user now has to decide whether sufficiently promising new parameter settings have been found. If so, the production plant can be halted and the new parameter settings installed. The resulting new average yield becomes available after 6 "weeks".

The "Profit" window gives an overview of the current situation of costs (initial costs, cost per run of the pilot plant, costs for changing settings of the production plant,...) and benefits (increased production). At the end of the 39 weeks, the balance should be positive and as large as possible.

This applet allows the user to get some hands on experience with several important concepts. All types of designs discussed in Response Surface Methodology courses can be tried out. The tank-to-tank variability has to be dealt with. Moreover the user is confronted with at least two other problems: when to change the production plant settings, and how to choose between one large experiment or many small experiments.

THE GREENHOUSE APPLET

The greenhouse applet requires the user to set up an experiment with (tomato) plants in a greenhouse. To succeed, the user has to deal with the problem of selecting appropriate levels for a treatment variable, and with the many problems caused by diversity in raw material and experimental circumstances.

The purpose of the experiment is to find the optimal dose of a new fertilizer. At the start, a set of 144 young tomato plants is available (12 trays of 3x4 plants) (Fig. 5). The young plants are not all the same: the initial weight is shown on the screen. The user has to select plants for the experiment and place them on the greenhouse tablet in the middle. The tablet is bordered on the left and the right with central heating devices (the thick black vertical lines). The tablet is also lighted by four special light bulbs. The resulting pattern can be seen on the screen. Consequently the positions on the tablet are not identical, and the differences in light and heat can be expected to affect the growth.

The user also has to decide which plant gets which amount of fertilizer. Defining and selecting doses, and applying doses to plants is done with simple mouse operations. To account for the difference in locations and initial weights, “grouping” factors can be defined. This is done on the left side of the screen.

When the plants have been properly placed, the user should select the time period for the plants to grow with the buttons at the bottom. When the “Grow” button is hit, the growth of each plant is simulated on an hour-to-hour basis, and the final weight (along with all the other variables) is available in a window. The growth simulation uses an adapted version of TOMGRO, a well-known growth model for tomatoes (Jones et al. 1991), as well as standard climatic data.

This applet allows the user to get comparative experience with almost all classical designs: completely randomized, complete or incomplete block, Latin Square, etc. There is also ample opportunity to invent and use new setups, made to accommodate the specific features of the greenhouse situation.

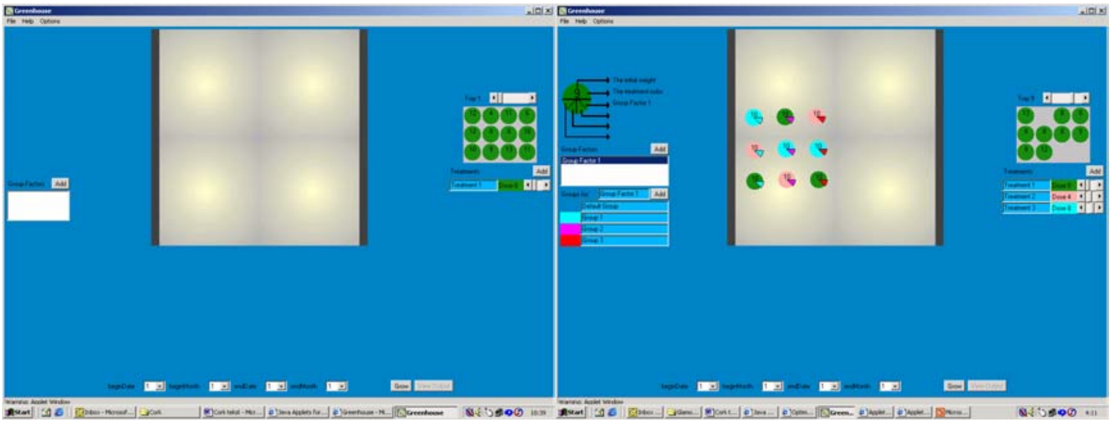

Fig 5. The greenhouse applet: initial view (left) and with a simple experimental setup (right)

THE VACCINE TRIAL

The effect of a vaccine for E. coli mastitis (udder infection) needs to be assessed by comparing the reduction in milk production in vaccinated and control dairy cows 48 hours after challenge with either a low or a high E. coli inoculum dose. The user has to select, from the available set of farms, which animals will participate and which challenge dose they will be given. Both factors inherent to the animal (e.g. parity) and environmental factors (linked to the farm) have an impact on the overall reduction and the vaccine effect.

The objective of this applet is to demonstrate two important aspects of experimental design: randomization is essential for valid conclusions, especially as random variability due to environmental factors is substantial. Second, efficient designs enable the investigator to explain part of the random variability and reduce error variability leading to more powerful testing. Randomized complete block designs, Latin Squares, split plots and blocked split plots can be generated in this experimental setup.

DEVELOPING WEB-BASED TOOLS: TECHNICAL ASPECTS

Currently, web-based tools can be developed using many different languages and software development aids. When it comes to applications for teaching, using applets written in the JAVA language has a number of apparent advantages.

The code for such an applet is stored on a web server. Any user with a computer connected to the internet can be assumed to have a web browser. These browsers have (if their options are set appropriately) the ability to download JAVA applets from anywhere on the net and execute them locally, without user intervention and without the need of any additional software being installed (this is, at least, the theory: see the remarks below).

To make an applet available, it only should be installed once on one server. Almost any number of students can access the applet at any time – the only load to the server is the downloading, execution takes place on the students' computers. The actual hardware/software configuration of the student should not matter. When a bug is detected, the applet on the server is replaced by a corrected version, and from then on the teacher can be sure that every student is always using the latest version of the applet.

JAVA applets also have their disadvantages. For security reasons, JAVA applets cannot read from or write to the user's disk. This implies that a user cannot directly use a statistical applet with his/her own data. However, this can be circumvented by explicit cutting/pasting from a window specifically generated by the applet. This is usually a satisfying solution for teaching applications.

Another implication is that the applet cannot make direct use of statistical software available on the user's computer. It could be run on the server (it is then called a servlet) and use a statistical package on the server, but this will increase the load on the server enormously, and involves licensing issues.

In practice, JAVA doesn't always live up to its "write once, execute everywhere" promise. To execute a JAVA applet, the browser uses a piece of software called the JAVA Virtual Machine. The most widely used PC operating system's browser has a Virtual Machine that cannot handle the newer versions of the JAVA language (in fact, at some point in the past, the browser had no JAVA Virtual Machine at all). It is technically possible to replace a Virtual Machine or to add one downloaded from another supplier. But then the JAVA capabilities of a user machine become dependent on its software configuration. Which JAVA capabilities future browsers will have will depend on the outcome of court trials and of economic pressure on software companies. But the continued general availability of internet browsers with the default ability to execute JAVA applets seems uncertain.

In VESTAC, all applets were written in JAVA 1.1, so they run on virtually any version of any browser. All statistical computations were coded in JAVA, so there is no need for a statistical package either on the server or the client side. With a reasonably fast internet connection, we found the download speed of the applets reasonable. The execution speed was found to be very high (taking the processor speed into account). To use an applet with a particular dataset (e.g. a textbook example), the maintainer of VESTAC must be asked to place that dataset on the server.

In VIRTEX, all applets are in JAVA 1.1, except the vaccine trial, which uses features of JAVA 2. This implies that for running this applet, the SUN Virtual Machine needs to be present. All the applets use JAVA to generate the data. Where relevant the resultant dataset is shown in a window. The user is supposed to cut it and paste it into a regular file or directly into a statistical package available on his/her computer. The greenhouse applet has two additional file-related features. It uses climatic data, which are stored in a separate file on the server. It also has a facility to capture the current state of the experimental setup (which can be quite complex) and save it into a window as a coded text file. The user can cut and paste the text into a regular file. The next time the applet is activated, the user can import the setup from the file and continue working with it.

DEVELOPING WEB-BASED TOOLS: TEACHING ASPECTS

While designing our web-based tools for use in statistics courses, a number of (sometimes difficult) choices had to be made again and again. This section gives an overview of some of these issues, and discusses how they have been dealt with in VESTAC and VIRTEX.

A first problem is: for which type of use is the applet primarily meant? Is it for in-class demonstration or for use by the student as a personal exercise or project? The VESTAC applets were primarily designed for in-class demonstration, and as a consequence have some typical properties: the first screen that shows up already allows meaningful experimentation. This requires a careful setting of the default values, but avoids the need to run quickly to several menus while the students see one window after another appear and disappear. Also the size of the most important widgets, the character and line sizes and the colors have been chosen to maximize visibility in a large lecture hall. The VIRTEX applets were designed with more personal use in mind.

Somewhat to our surprise, we got many requests from teachers for stand-alone versions of VESTAC: it seems that still many lecture rooms lack convenient web access. There is a facility to download VESTAC as a zip-file and install it on a PC where it can be used without the need for web access. Obviously, with this approach there is no longer any guarantee that the version used is the most recent one.

Another problem is the choice of topics. Despite what is seen in some distance courses, not every figure requires interactive experimentation. In VESTAC, we have been very careful to select only those topics for which it was thought that an interactive visual applet could have didactic properties exceeding those attainable with more traditional didactic tools.

There is also the choice between specific and general. It is tempting to design an applet in such a way that it can do almost “anything” about a given topic. This leads to huge applets that take longer time to download and are more difficult to understand and use because there are so many options. Maintenance is also more difficult. In VESTAC we tried to make lean applets that illustrate one point each.

Another issue relates to the models used. In VIRTEX, we tried to make the underlying models as realistic as possible. Sometimes the real value of a parameter is not the best one from a teaching point of view. In the greenhouse applet for example, the influence of the light on the final response is not very large. Hence the student ignoring this factor is hardly “punished”. It is then tempting to increase the importance of this factor in the model. The new model will be less realistic, but perhaps a better teaching tool.

A related issue is the best way to deal with randomness. In virtually all applets random sampling or random errors are used. Again this makes the demonstration more realistic but has some drawbacks. Random aspects cannot be repeated, and it is precisely this repetition (with some other aspect changed) that can give valuable insight. In the greenhouse applet 144 plants are provided with initial weights chosen at random. Comparing two experimental setups is then very difficult, because the starting material is different every time. We plan to have, in a future version, a randomization feature that can be turned on/off.

Finally it should be noted that however broad an applet is designed, it always reflects the designer's view on the subject matter. Hence it will not appeal to every teacher. On the other hand we experienced that some teachers use VESTAC applets in ways unanticipated by the developers. One teacher used the nonlinear regression applet to explore the shape of a likelihood surface, another teacher used the t-distribution applet to study the Cauchy distribution through a t-distribution with 1 df.

DISTANCE EXPERIMENTS

A distance experiment is also a web based tool that allows the student to "execute" an experiment, but now the experiment is not simulated like in VIRTEX, but real. An experimental setup is placed at some physical location. It must be fully controllable by means of a computer – taking observations should not require human intervention. Then, it can also be controlled by another computer connected to the first over the web. The user can, through a web page, "turn the knobs" of the experimental apparatus, start up an experiment and actually see what is going on in real time through one or more video images. Numerical results are also shown.

The implementation of a distance experiment involves much more than setting up a connection: a lot of hardware and interfaces must be made to operate reliably, software must be written to restrict control to one user at a time and the user's actions must be closely monitored to prevent unsafe situations. Obviously, this approach can only be used for certain types of experiments.

Up to now, distance experiments have primarily been used to allow broader access to expensive or highly specialized devices (Dominguez et al, 2001). Applications for teaching are mainly in the context of robotics and control theory (Curto et al, 2001). However, we think there is a future for distance experiments within several courses in the university curriculum, including statistical and/or experimental design courses, because they have a number of very interesting features. They make it feasible to include (almost) real experimentation even in courses with larger numbers of students, when there are no sufficient lab facilities available during working hours. They allow students to gather data that they can experience as real and as their own. Experimentation is now possible 24 hours a day, and the students have access to the facilities over a longer time span. This implies that the student doesn't have to follow a strict cookbook procedure, but can explore several approaches and learn from his/her errors.

A prototype distance experiment has been set up recently (Groenewege 2003). It consists of a set of fertilized chicken eggs. A robot can select an egg indicated by the software, apply a weak mechanical impulse, and record the spectrum of the acoustic response signal. By selecting particular eggs at particular times, a student can follow the evolution of the vibration characteristics of each egg, and try to extract from the data information about the embryonic development, the occurrence of cracks and predict hatching time. Further distance experiments are planned for joint use in the courses of statistical data analysis and properties of biological materials.

Though the niche for this type of applications is still unclear, they have the potential to enrich statistics courses with very personal and realistic experimentation and analysis experiences.

ACKNOWLEDGEMENTS

Development of the VESTAC library was financed through grants from the Flemish government (Ministry of Education), the K.U. Leuven Education Council, and

Ghent University. Bert Raeymaekers, Stefan Michiels, Wim Moreau, Andrej Solomin, Brian van de Noortgate and Ruqi Wang contributed substantially to the programming of the library. VIRTEX was financed through grants from the Flemish government (Ministry of Education) and K.U. Leuven. Steve Dufresne and Bart Jacobs contributed to the programming.

REFERENCES

Anderson-Cook, C.M. and Dorai-Rai, S. (2001). An Active Learning In-Class Demonstration of Good Experimental Design, *Journal of Statistics Education*, 9.

Cobb, G. (1992). Teaching Statistics. In Heeding the Call for Change: *Suggestions for Curricular Action*, L. A. Steen (Ed.), Washington DC. Math. Association of America.

Cobb,G. (2002). Introduction to Design and Analysis of Experiments. Key College Publishing, Emeryville California.

Curto, B., Moreno, V., Vincente, J.A., Moreno, A.M. and Garciá, F.J. (2001). "A Java based tool to develop a remote laboratory for process control teaching". IFAC workshop on Internet Based Control Education, Madrid, Spain.

Darius, P.L., J-P. Ottoy, A. Solomin, O. Thas, B. Raeymaekers, S. Michiels. (2000). "A Collection of Applets for Visualizing Statistical Concepts" In Proceedings in Computational Statistics 2000, Bethlehem, J.G. and P.G.M. van der Heijden (eds). Physica Verlag.

Darius, P.L., J-P. Ottoy, O. Thas, S. Michiels, B. Raeymaekers,. (2002). "Applets for Experimenting with Statistical Concepts" In: ICOTS 6 Proceedings.

Darius, P.L., H.C.M. van der Knaap, E. Schrevens, K.M. Portier, G. Massonnet, L. Lievens, L. Duchateau (2003). "Virtual Experiments and their Role in Teaching Design and Analysis of Experiments". In: ISI 2003 Proceedings.

Dean, A. and D. Voss. 1998. Design and Analysis of Experiments. Springer, New York.

Dominguez, M., Reguerra, P., Javier González, J., Marcos, D. and Felipe Blázquez, L. (2001). "Connection pilot plant to the internet". IFAC workshop on Internet Based Control Education, Madrid, Spain.

Jones, J.W., E. Dayan, L.H. Allen, H. Van Keulen, and H. Challa. 1991. "A Dynamic Tomato Growth and Yield Model (TOMGRO)." Transactions of the ASAE 34, No. 2, 663-672.

Groenewege, P. (2003). "Development of a web-based interface for the control of an experiment" (in Dutch). Unpublished Ms Sc Thesis, K. U. Leuven.

Marasinghe, M.G., Meeker, W.Q., Cook, D. and Shin, T.-S. (1996). Using Graphics and Simulation to Teach Statistical Concepts, *The American Statistician*, 50, 342-351.

Moore, D.S. (1997). New Pedagogy and New Content: The Case of Statistics, *International Statistical Review*, 65, 123-165.