META DATABASE FOR DATASETS REGARDING STATISTICS EDUCATION

Peter Pipelers, Ellen Deschepper, Heidi Wouters, Olivier Thas and Jean-Pierre Ottoy
Department of Applied Mathematics, Biometrics and Process Control, Ghent University, Belgium
Peter.Pipelers@UGent.be

This project involves collecting didactic examples and datasets in the statistical field used for educational motives. Teachers often find it hard to come across good example datasets, accompanied with metadata of the collection procedure, the statistical methods used to analyze the data, which specific problems must be taken into account when conducting the analysis and specific background information regarding the dataset. The core of the project is the realization of a meta database, classifying the data and metadata with their sources (book, research and consultancy assignments), and the implementation of a user-friendly environment to generate structured search requests on the properties of the analyses on the (MySQL) database using a graphical user interface (Java Swing). The generic structure of the database allows to add analyses in conformity with the database design.

BACKGROUND

A general concern of teachers related to the statistical education field is finding good example datasets, which they can use for educational purposes. Despite of the fact that worldwide a lot of datasets are available, easy access or obtaining these datasets is not always a guarantee. Datasets are typically acquired by other science fields, such as biology, medial science, economics, sociology… The course notes provided by the teacher contain many example datasets for which the assembly required laborious efforts. As a consequence, lack of diversity of examples is a reoccurring phenomenon. Interchange of such example datasets among teachers of different fields of study is obviously a short-term didactic solution. It is a time-consuming process and the wide variety of the different fields of study raises difficulties in communication. A lot of syllabi tend consequently towards inadequate contextualization, which is didactically considered an indisputable disadvantage.

We therefore obtained for the implementation of an electronic consultative database where datasets are made available. The design of the database implies that extensive search assignments can be released to consult the database in a uniform matter, resulting in the provision of ready-to-use example datasets with all necessary information regarding the data and the analyses.

CONTENT OF THE DATABASE

A first step towards the design of such a database requires the question '*which information should be provided?*'. It is obvious that all necessary metadata should be considered. This will include detailed information on the manner the data was collected (experimental design), since this implies an irrefutable relevance to the research question for which the data is obtained for.

Besides the metadata, the database should clearly hold which statistical analysis can be conducted on the dataset. The teacher founds his search on these items when he searches a practical example for a certain analysis or study design.

Other important information fields are the specific problems that must be considered when the analyses are conducted and the specific background information regarding the dataset. All data encounters specific analysis and data problems, which may encourage certain examples. It is always informative to consider theoretical concepts, such as non-normality, outlying observations or confounding, utilized and contextualized in a practical dataset.

Since every field of study, every dataset and every analysis expects different important sources and problems, the design of the database should be well reflected. After all, we want our search statements to be performed on any of the above information criteria. The design of the database plays key role in this project and authorizes flexibility.

In C. Reading (Ed.), *Data and context in statistics education: Towards an evidence-based society. Proceedings of the Eighth International Conference on Teaching Statistics (ICOTS8, July, 2010), Ljubljana, Slovenia*. Voorburg, The Netherlands: International Statistical Institute. www.stat.auckland.ac.nz/~iase/publications.php [© 2010 ISI/IASE]

PHASES OF THE PROJECT

A general time structure of the project is attributed into several stages:

- Assembly of the datasets: Besides the classical sources (course notes, books, …) datasets related to internal and external research and consultancy activities are also considered, since they will add extra relevance to an example analysis and are direct applications of educational research.
- Assembly of the background information regarding the datasets: A well designed analysis and interpretation of the results requires all necessary information about the data to be available. The experimental design and research questions are considered as key aspects.
- Considering and conducting the possible analyses on the available datasets.
- Phrasing the conclusions and classifying the specific data and analysis related problems.
- Design of the database: Inevitably a repeating process.
- Implementation of a graphical user interface: The construction of the search statements must follow a generic user-friendly approach.

Determining the design of the database is naturally an iterative procedure spread over several phases. The design should reflect the qualities we encountered, but must be flexible when we aim for a large-scale application. The information included in the database, corresponding with a dataset, is the field of study, the conducted analyses and problems or properties related both the dataset and the analysis.

DETAILS OF THE INTERFACE IMPLEMENTATION

The use of databases is the preferred method in multi user applications. Several database management systems are available. To remain in the open source context, we chose to use MySQL (My Structured Query Language). The system uses the Structured Query Language (SQL), which is the most used language to create, to require and to modify data in a relational database management system. MySQL's generic character applies on several platforms, among them Linux and Windows.

The design of the database is obviously related to the structure of the search assignments, for which the databases practice will be most useful. Of course, the user knows nothing of the internal structure. The search procedure for the datasets of interest should however be accessible in a user-friendly way. A graphical user interface (GUI) which focuses on the internal composition of the search procedure, without the user knowing what is going on, is most desirable. The practical use of this interface provides the properties for which the user can search for.

The addition of checkboxes and radio buttons to a simple web form to supply the user with search choices, remained an early concept. It seems to be an outran idea as several disadvantages come to mind. Each selection will imply the search procedure to load a new page, anticipating no flexibility if a related search should be considered. Simple web forms often show security leaks as a result of the direct connection between the web forms and the database. Also, the word graphical in the GUI is not convenient when we let the layout depend on the browser.

When one thinks of the graphical character, independent from browser to browser, applets certainly come to mind. An applet is no less than an easy tool to provide platform independence. A well known property of an applet is that it can easily be embedded in a webpage. Such an application is often considered in an interactive context. Applets are executed on the platform of the client and go by the sandbox security mechanism. As a result, a database connection initiated from an applet is usually disallowed.

Bales (2002) described the implementation of Java applets that correspond with a database using a servlet, known as the applet-servlet architecture. These objects provide several advantages. Servlets generate responses based on received requests. So, we could leave the process of the search statements and the resulting output entirely to the servlet.

Apart from servlet objects, the choice of Java is not surprising. Opposite of the earlier Abstract Window Toolkit, the Java Swing library was developed to provide a more sophisticated set of GUI components. Every component in the Swing library is entirely written in the Java language and therefore guarantees that the output is the same for all browsers and platforms and adds some flexibility to the look and feel.

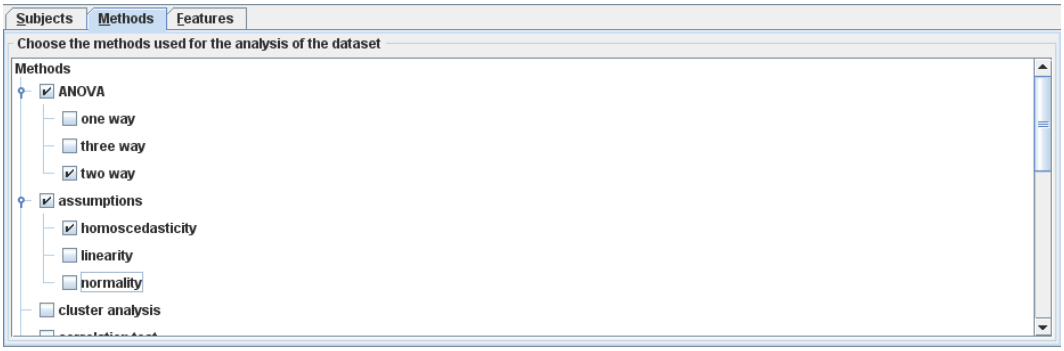
The execution of the application is uncomplicated: the applet communicates with the database through the servlet. Behind the scenes, the applet will stepwise construct SQL statements. The servlet processes the request with the statements and sends back the answer it receives from the database. The operation itself is very fast, since we lighten the functionality of the applet and let the hard work over to the servlet. Extra security is built in, since the servlet operates on the same server with the database management system. No direct connection from the client to the database is ever constructed. The only aspect the applet considers, is plotting the information it receives and sending information the user desires.

STRUCTURE OF THE APPLET

The design of the database is a guideline through the project, which stresses again the importance of reflection. It is necessary that the user bases his/her search strategy on the three aspects of interest: the field of study, the conducted analyses and problems or properties related both the dataset and the analysis. In the applet they are marked as *subjects*, *methods* and *features*. You can find them in the upper pane, known as the search pane. These are the aspects of most interest for the user. We make a distinction between analyses and sub-analyses, as can be seen by the screenshot below (ANOVA–Two Way). If a teacher wants to discuss a statistical topic in a certain field of study, he can simply fill in his/her selection by selecting the checkbox of choice.

The applet will immediately construct search statements, send them to the servlet and as a result, the applet shows the direct outcome of the search procedure on the database in the bottom pane, known as the result pane.

The user is then free to continue with his/her search procedure or to restart with a new search. He/She can do the same for the desired analyses and for the problems or the properties. Each change in selection implies a different result set, which is directly shown in the bottom part of the applet.



OUTPUT OF A SEARCH STATEMENT

The results of the search procedure are consistent. Each dataset corresponds of, if existing, a dataset file, a document describing the whole analysis (origin and experimental design of the dataset, explanation of present variables, …) and a command file of the used software for the analysis (R, SAS, …). All files are gathered in an archived file, provided in the indicated column. Information on who analyzed the data and uploaded the analysis is also handed.

Table (selection based on less strict search)

Dataset Name	Zipfile	Data file	PDF Document	Command File	Uploaded by user	Uploaded on
Balance	balance.zip	balance.txt	balance.pdf	balance.R	ppipeler	Mon Apr 14 10:08:20 ...
Calories	Calories.zip	calories.csv	calories.pdf	Calories.R	ppipeler	Mon Apr 14 10:08:20 ...
Cancer	cancer.zip	cancer.txt	cancer.pdf	cancer.R	ppipeler	Mon Apr 14 10:08:20 ...
Cereals	cereals.zip	cereals.txt	cereals.pdf	cereals.R	ppipeler	Mon Apr 14 10:08:20 ...
Chocolate	chocolate.zip	chocolate.txt	chocolate.pdf	chocolate.R	ppipeler	Mon Apr 14 10:08:20 ...
Gwaimasi	gwaimasi.zip	gwaimasi.txt	gwaimasi.pdf	gwaimasi.R	ppipeler	Wed Apr 16 17:57:28 ...
Heavy Metals	heavymetals.zip	heavymetals.txt	heavymetals.pdf	heavymetals.R	admin	Thu Apr 17 09:59:33 ...

Simply clicking on the matching buttons supplies the favoured information. The current implementation foresees no restrictions on the visibility of the data.

STRICT AND LESS STRICT SEARCH

Two search strategies are considered. The default strategy is the *less strict* search, applying a logical OR within each criteria, structured in the same tabbed page. If a certain dataset satisfies one selection criterion, the dataset will be included in the outcome. It can be easily seen that this could cause redundancy. Another provided search strategy, the *strict* search, offers an improved effect. The logical AND is applied both within and between the criteria. The resulting datasets correspond with all selections in contrast with the first strategy. One can alter between the search strategies through the Settings menu.

UPLOADING NEW DATASETS

Continuity is necessary to fully profit from the functionality of the database. Introducing a meta database for datasets in a graphical manner is one thing. Flexibility is another frequently neglected aspect. Our project anticipates this drawback by the implementation of an upload module. An authorized user can simply add a new dataset to the meta database and provide all necessary information that belongs to this dataset (field of study, conducted analyses and problems or properties related both the dataset and the analysis) by employing a similar selection display as the search module. The procedure guarantees that no violations towards the database design are made.

We are aware that a user might not agree with our intended structure. The facility of changing the design is provided. A user can simply add a node or sub node in one of the criterion lists, adding flexibility to the structure.

The database is however not immediately changed, as this could cause a violation in the isolation and consistency rule of a relational database, but the project developer is notified. Since all information in the GUI is provided by a database, flexibility in the structure is guaranteed.

CONCLUSION

We discussed the construction of a meta database, classifying data and the metadata with their sources, and the implementation of a user-friendly environment to generate structured search requests on the properties and specific problems of the analyses using a Java Swing applet. The resulting database supports a teacher with the preparation of explaining statistical concepts by means of a realistic dataset in a desired study field.

Flexibility is settled with the implementation of an upload module, to add datasets to the database in conformity with the database design.

DISCUSSION

We remark that the functionality of such a database is enhanced the more data is added. Interchanging datasets among teacher of different departments, faculties and universities by means of the upload module in the GUI, will add more power to the meta database. By improving both the design and magnitude actively, the meta database will prove its benefit. However, this requires discipline and initiatives of a whole lot of statistical tutors.

A facet we may not forget is that data is made public available through this database. This implies that dealing with copyrights should be treated carefully and demands a discussion. Can certain data be provided online or should we solve it otherwise? Will the public character of the database provide problems? For this reason, the GUI is yet only accessible for a small public, which obviously restricts its functionality.

REFERENCES

Bales, D. (2002). Database Access Using Lightweight Applets. http://onjava.com/pub/a/onjava/2002/02/20/applets.html

Loy, M., Eckstein, R., Wood, D., Elliot, J., & Cole, B. (2002). *Java Swing* (2nd edition). Sebastopol: O'Reilly.

Nolan, D., & Speed, T. P. (1999). Teaching Statistics Theory through Applications. *The American Statistician*, *53*.

Vaughan, T. S. (2003). Teaching Statistical Concepts With Student-Specific Datasets. *Journal of Statistics Education*, *11*(1).