Aydinli, Gökhan; Härdle, Wolfgang; Kleinow, Torsten; Sofyan, Hizir

Working Paper

MD*ReX: Linking XploRe to standard spread-sheet applications

SFB 373 Discussion Paper, No. 2002,10

Provided in Cooperation with:
Collaborative Research Center 373: Quantification and Simulation of Economic Processes, Humboldt University Berlin

Suggested Citation: Aydinli, Gökhan; Härdle, Wolfgang; Kleinow, Torsten; Sofyan, Hizir (2002) : MD*ReX: Linking XploRe to standard spread-sheet applications, SFB 373 Discussion Paper, No. 2002,10, Humboldt University of Berlin, Interdisciplinary Research Project 373: Quantification and Simulation of Economic Processes, Berlin, http://nbn-resolving.de/urn:nbn:de:kobv:11-10048600

This Version is available at:
http://hdl.handle.net/10419/65364

MD*ReX: Linking XploRe to Standard Spreadsheet Applications

Gökhan Aydınlı, Wolfgang Härdle, Torsten Kleinow and Hizir Sofyan [1]

Summary

We will show a methodology of incorporating a profound statistical software environment into a standard spreadsheet application. Our approach is based upon a client/server computing philosophy, which will enable the user of our client side application to choose between various types of servers according to his needs of computing power.

1 Introduction

Spreadsheet applications such as Microsoft Excel have statistical utilities in their standard distribution but require sophisticated low level programming in handling advanced statistical procedures. Simple examples are heteroskedastic regression equations, cluster and panel data analysis or linear models with many parameters. These and many other statistical procedures are not provided with most standard spreadsheet applications although they are of crucial importance for teaching or research.

On the other hand, commonly used statistical packages do not offer the intuitive graphical user interface (GUI) of a spreadsheet application. It is

[1] Institut für Statistik und Ökonometrie, Wirtschaftswissenschaftliche Fakultät, Humboldt-Universität zu Berlin, Spandauer Str. 1, D-10178, Germany. Financial support from the *Deutsche Forschungsgemeinschaft*, SFB 373 is gratefully acknowledged.

especially this *direct manipulation interaction model* (Neuwirth & Baier 2001) of spreadsheets which is not comparable to enriched programming languages like XploRe.

Furthermore statistical packages often lack fully integrated support to other office applications or database solutions, but provide broad coverage of "state-of-the-art" statistical procedures. XploRe is such a statistical software developed for data analysis, research and teaching. It supports its user to handle an extensive set of pre-mastered procedures such as descriptive statistics, multivariate data analysis, time series analysis and financial applications.

However, statistical practice and the efficient use of modern statistical technology require both, powerful as well as flexible computing environments. One suggestion to the reduction of this problem is to link statistical software packages like XploRe to standard spreadsheet applications. Our aim hereby is to demonstrate how to embed an external statistical software into Excel taking advantage of *client/server* based computing. Obviously, this strategy offers two improvements: the user can access such statistical procedures which are not available in a standard distribution of Excel and the client application MD*ReX grants the spreadsheet user access to these statistical methods implemented in XploRe without the need to change his familiar environment. Conversely MD*ReX enables the user to present the results in a variety of graphical representations via the Excel graphical environment.

Rather than providing a set of Dynamic link libraries (*DLL*) that offer a basket of possibilities we pursue a more transparent strategy. We show how XploRe, as a statistical engine, can be used in Excel via the client application MD*ReX, which integrates various 'ready to use' statistical procedures into spreadsheet facilities. MD*ReX is designed to be able to exchange data between both software applications and takes advantage of a familiar spreadsheet interface. The basic idea of MD*ReX is to keep the XploRe high level programming language visible for faster development of new custom statistical methods and to provide menu based statistical libraries that are otherwise not available in MS Excel.

The remainder of the paper is organized as follows. In the next section, we present the potentials of MD*ReX. The technical implementation is discussed in section 3. The GUI is presented in section 4 and section 5 gives some illustrations.

2 Potentials of MD*ReX

XploRe is a well defined environment for both classical and modern statistical procedures in conjunction with sophisticated graphical capabilities. The package can be regarded as a device independent and interactive platform for statistical analysis, research and teaching. Its objective lies in the effective exploration and analysis of large scale data, as well as in the development of

new techniques like client/server based data analysis or multimedia enriched statistical teaching (Härdle, Klinke & Müller 1999). The statistical methods (*Quantlets*) are provided by various libraries (*Quantlibs*) and are fully accessible from networked environments like the WWW. Some of the included methods are generalized linear and partial linear models, kernel estimation, smoothing, spline smoothing, micro econometrics, generalized additive models, option pricing, stock simulation, nonlinear time series analysis, modern regression techniques, Kalman filtering, wavelets and neural networks.

A high level matrix oriented programming language enables the user to gradually adapt the computing environment to his needs by defining custom *Quantlets*. Variables can be collected in list structures making it possible to hold common information of a data set in a single object. All the features of a high-level language like recursion, local variables, loops and conditional execution are available. Dynamic link libraries (*DLL*) and remote procedure calls (*RPC*) to other applications enlarge the environment. An automatic HTML converter ensures the smooth integration of *Quantlets* into the Auto Pilot Support System (*APSS*).

The client/server version takes advantage of the Java technology and is especially designed for applications in the Inter-/Intranet. The XploRe Quantlet Server (*XQS*) delivers the full set of *Quantlibs*. This ensures that statistical technology can literally be accessed from almost any networked workstation or PC. The statistical engine for instance is used within the authoring and typesetting environment *e-book*, developed in cooperation with Springer-Verlag.

Excel is one of the most successful applications ever developed, backed by the overall acceptance of Microsoft's Windows operating system. As one of the first spreadsheets, Excel allowed the user to communicate with the application through a GUI and a mouse pointing device instead of a command line syntax, seeding today standards of pull-down menus, "drag'n'drop" functionality and mouse clicking.

As aforementioned the value of the spreadsheet lies in its flexibility. It allows one to interactively manipulate data and obtain corresponding graphical representations. In business applications Excel is a quasi standard for data manipulation and visualization as more and more business or more general statistical data sets are presented in spreadsheet format. Excel is also frequently used as a database front-end as it supports various data retrieval methods like the Open Database Connectivity (*ODBC*) standard.

However, Excel has only a limited number of statistical procedures. Moreover, Knüsel (1998) and McCullough & Wilson (1999) criticize an insufficient numerical reliability of built in statistical procedures. They analyze the spreadsheet's performance by examining its capability of estimation, random number generation and statistical distribution and suggest a cautious usage of Excel for statistical analysis.

MD*ReX is a convenient solution to Excel's lack of performance in con-

ducting modern statistical procedures. It offers 'ready to use' statistical procedures and takes advantage of the powerful spreadsheet and graphical facilities of Excel. It is therefore mainly useful as a data editor, reporting toolbox and comfortable interface. MD*ReX uses a three level client/server architecture supported by the XploRe Quantlet Server/XploRe Quantlet Client *(XQS/XQC)* technology sketched in the next section. The client allows local and remote connections to any *XQS* and offers an user friendly intuitive interface.

3 Technical Implementation

The ability of MD*ReX to access and execute statistical *Quantlets* is based on the XploRe client/server architecture described in-depth by (Kleinow & Thomas 2000), (Härdle, Kleinow & Tschernig 2001) and (Kleinow & Lehmann 2002) and illustrated in Figure 1. The architecture consists of three parts, the Quantlet Client (*QC*), the middleware program *MD*Serv* and the XploRe Quantlet Server (*XQS*). The data transmission between client and server is handled via the TCP/IP based protocol *MD*Crypt* (Feuerhake 2001). The *XQS* is the backend and provides the computing service for statistical analysis. The *XQS* may reside on the local host or on a remote *MD*Tech* center (like `http://www.i-xplore.de`).

The service provides a large number of numerical methods and statistical procedures as well as a high level programming language. The kernel of the *XQS* is identical to that of the standard desktop XploRe environment. Existing *Quantlets* may therefore be easily used in an *XQS/QC* setting. These *Quantlets* can be exclusively written in XploRe language or be a combination of native code written in any programming language (*DLL, SO*) and XploRe.

The middleware program *MD*Serv* avoids programming overhead at the *XQS* and handles all the TCP/IP communication details. *MD*Serv* is a software running on the *XQS* host that manages the *XQS/QC* communication, i.e. it reads data from the standard output stream of the *XQS* and transmits this data to the *QC* and vice versa. *MD*Serv* is implemented in Java to make it available on MS Windows platforms as well as on Unix machines.

In a browser or an applet based setting the user interface to the *XQS* is the *XploRe Quantlet Client*. In the MD*ReX environment MS Excel plays this role. The implementation of the protocol *MD*Crypt* is shifted to a *DLL* which is dynamically loaded during runtime. This avoids any TCP/IP specific programming for example in Visual Basic for Applications.

The user has the choice either to connect to a local instance of the middleware implementation or to a remote *MD*Tech* center, running various *XQS* on high performance SUN Solaris servers. The decision whether to work locally or remotely should be governed by the amount of data to be processed and the type of analysis to be conducted in the specific case.

Besides the MD*ReX environment the client/server architecture is also

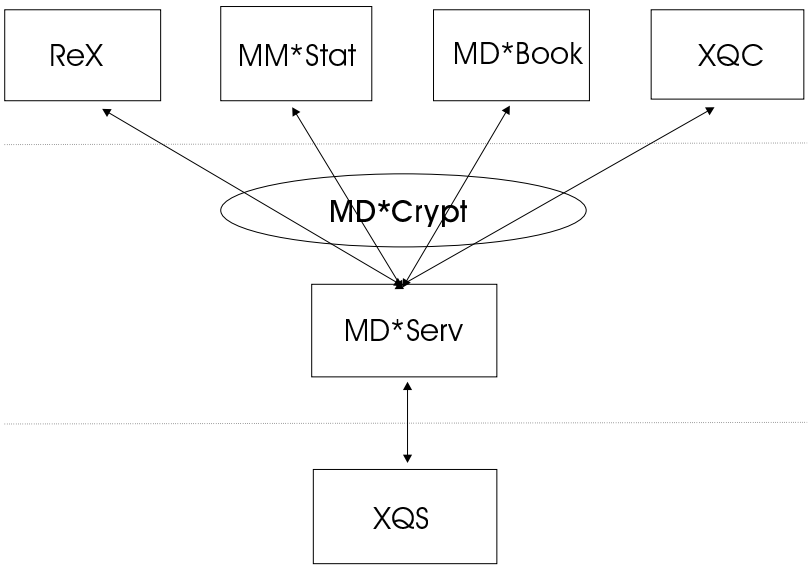

Figure 1: XploRe Client Server Architecture

applied in two other projects. One is the *XploRe Quantlet Client* that provides a GUI similar to that of the XploRe MS Windows edition for the XQS. This client is used as a development tool for new *Quantlets* and as a web based computing environment. It is applied in the context of cooperation with Springer-Verlag for electronic books (`http://www.md-stat.com`) to illustrate the contents by interactive examples. This technique is used in the *MM*Stat* project (`http://www.mm-stat.de`) for interactive teaching of statistics. In electronic papers it provides access to the introduced methods, e.g. the electronic version of (Härdle, Hlavka & Klinke 2000) at `http://www.xplore-stat.de/ebooks/e_xag.html` and of (Härdle, Kleinow & Tschernig 2001) at `http://ise.wiwi.hu-berlin.de/~rolf/webquant.pdf`. The second client is the *GraphFitI* project at `http://www.stat.uni-muenchen.de/welcome.html`. *GraphFitI* is a program for model selection in graphical chain models that uses the *XQS* as a statistical engine.

4 Graphical User Interface

MD*ReX is designed to provide an intuitive and easy to use interface for statistical analysis. Our technical implementation of the graphical user interface has been guided by the idea of an easy object exchange between client and server. Input and output parameters as well as variables may be identified using standard MS Excel facilities like cell-selection, copy and paste, etc.

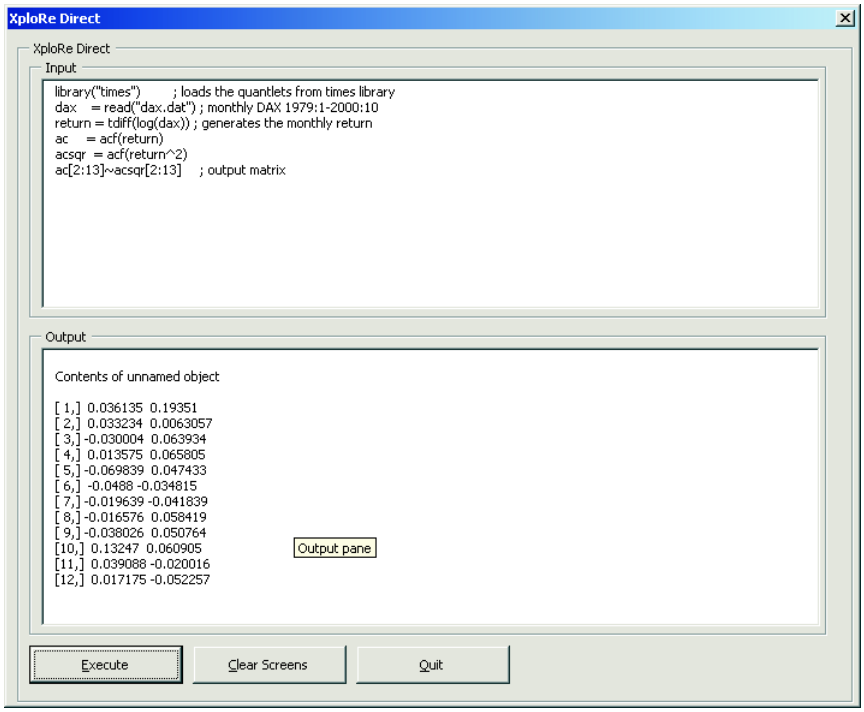

Figure 2: ReX Editor Window

Our strategy of a transparent environment is implemented through an editor (see Figure 2) which enables the user to directly write and execute XploRe *Quantlets*. After referencing the add-in in Excel via the add-in dialogue the client application is automatically loaded when the user starts Excel. Alternatively the add-in can be triggered from the Start → Programs shortcut in Windows 9x, NT, 2000 and ME.

Implemented as a typical Excel add-in file, the user is encountered with the standard Excel desktop while working with MD*ReX, i.e. the user will recognize a familiar environment with additional features (see Figure 3 and 4).

The MD*ReX client consists of menu items, context-menus and dialogues. In addition the GUI provides information about the current computing state such as the connection status, *XQS* IP-address, host name, used ports, etc.

Handling the client via the toolbar is rather straightforward as the buttons are self-explaining. The MD*ReX client is designed such that different user profiles are addressed. Pushing the "Connect" button makes Excel to an editor for the *XQS*. More refined procedures are consolidated under the "Libraries" button. Evidently the editor modus is intended for the user with

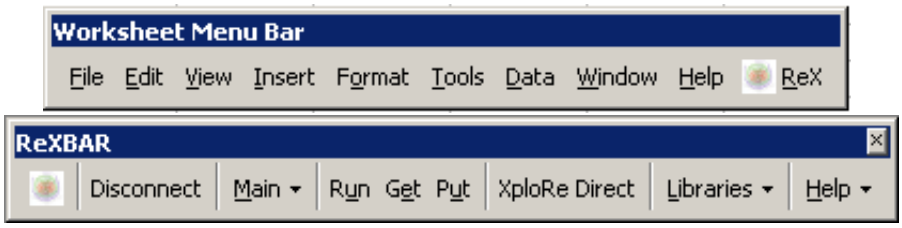

Figure 3: Excel's Menubar and MD*ReX Toolbar

some background in XploRe who wants to transfer numeric and string variables from his spreadsheet application to XploRe or wants to execute XploRe commands within the ranges of an Excel worksheet (see Figure 4).

The other emphasize lies on the "embedded" mode, where the interaction of the Excel programming environment and the *MD*Crypt* implementation for MD*ReX plays a crucial role.

The "embedded" interaction of both software packages becomes clearer as the XploRe compliant matrix language has been tightly though dynamically embedded within Excel's VBA environment to accomplish a frictionless communication between both and to provide the XploRe unacquainted user an easy access to its statistical methods.

MD*ReX can be downloaded at `http://www.md-rex.com`.

5 Illustration

We illustrate the GUI by two examples: Autoregressive conditional heteroskedasticity (ARCH) modelling for financial time series and cluster analysis.

5.1 ARCH

A well known stylized fact in financial data analysis is volatility clustering. ARCH models represent a dominant technology to model such volatility phenomena by relating current volatility in a linear fashion to past volatility (see Engle 1982, Bollerslev 1986).

More precisely the class of ARCH models is described by

$$Y_t = \sigma_t + \varepsilon_t,$$

$$\sigma_t^2 = f(\tilde{\sigma}_{t-p}^2, \tilde{Y}_{t-p}),$$

$$\tilde{\sigma}_{t-p}^2 = (\sigma_{t-1}^2, ..., \sigma_{t-p}^2)^T, \tilde{Y}_{t-p} = (Y_{t-1}^2, ..., Y_{t-p}^2)^T, \forall t, p \in \mathbb{N},$$

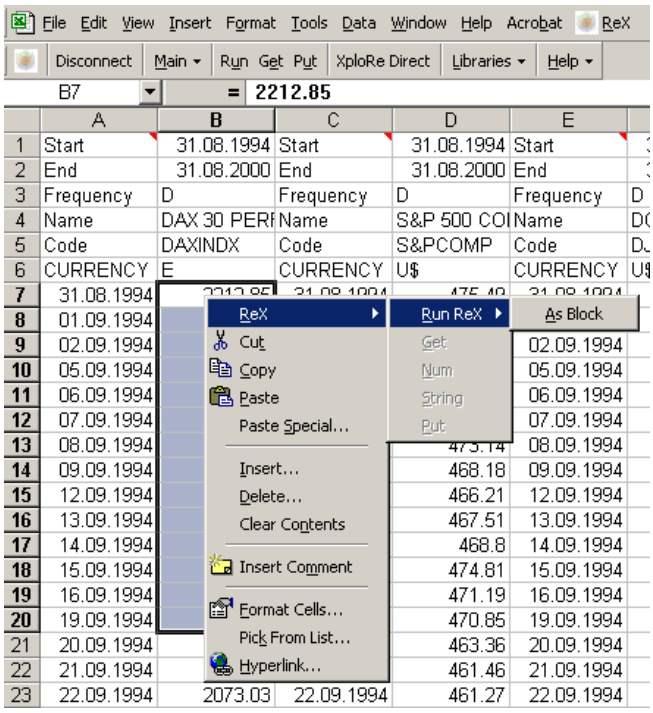

Figure 4: MD*ReX Context Menu

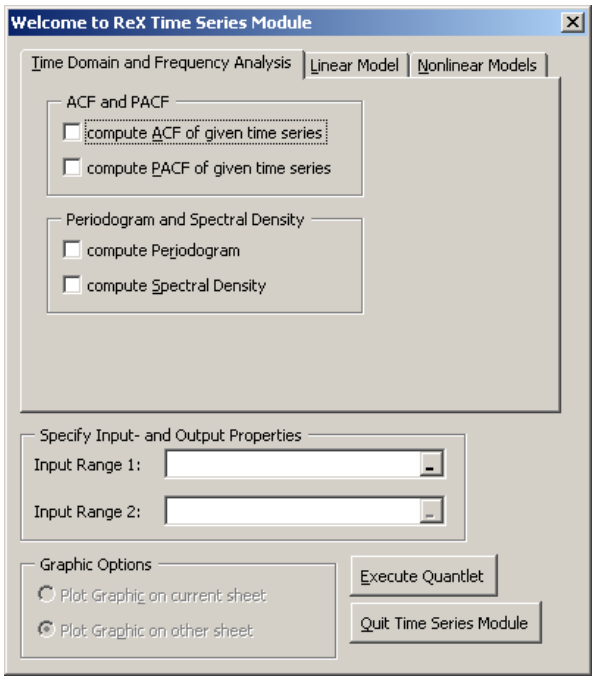

Figure 5: The MD*ReX Time Series Dialogue

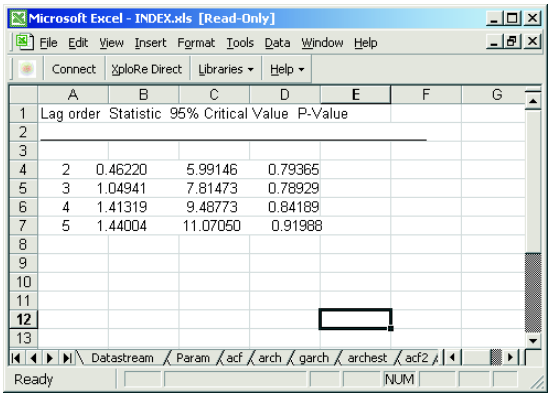

Figure 6: **MD*ReX in a GARCH analysis**

$f(\cdot)$ is a linear function of $\tilde{Y}_{t-p}$ for ARCH and a linear function of both $\tilde{\sigma}^2_{t-p}$ and $\tilde{Y}_{t-p}$ for generalized ARCH (GARCH). More refined functions are described e.g. in Gouriéroux (1997).

In the usual practice, the data are stored in a MS Excel sheet to which the ARCH *Quantlet* is applied. The model fit and estimated parameter values may be transferred to a different sheet. The situation is shown in Figure 5, where the ARCH *Quantlet* has been selected from the Time Series dialogue through the MD*ReX pull down menu.

The time series inputs are chosen from the spreadsheet, the procedure may be repeated with different parameter values and input variables as described above. The result of the estimated parameters and the model fit is shown in Figure 6.

5.2 Cluster Analysis

Clusters are groups of observations with similar situation. Cluster analysis is a set of techniques that identifies the structure of groups and classifies a set of observations into two or more mutually exclusive sets $(P_1, ..., P_k)$. Its aim is to construct groups in such a way that the profiles of objects in the same groups are relatively homogeneous whereas the profiles of objects in different groups are relatively heterogeneous. The main advantage of this technique is that homogeneous structures or clusters can be found directly from the data without imposing background knowledge on the involved variables. In general, clustering could be divided into two categories: Hierarchical Clustering and Non-hierarchical Clustering.

A hierarchical Clustering method gives at any stage in the procedure, a merger of clusters at a previous stage. It creates a tree-like structure of the clustering process. It is well known that the clusters of items formed at

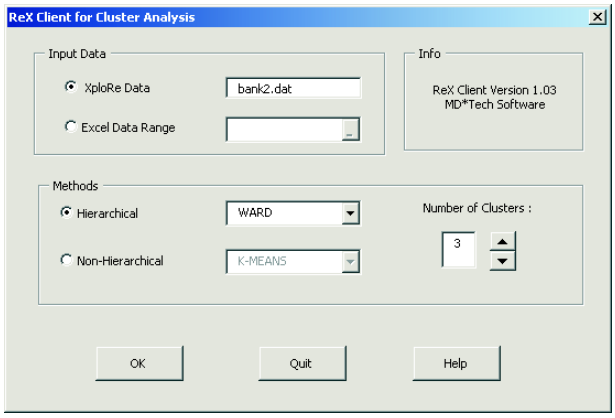

Figure 7: The MD*ReX Cluster Analysis Dialogue

any stage are non-overlapping or mutually exclusive. Examples of hierarchical techniques are single linkage, complete linkage, average linkage, median, Ward, etc. (Kaufmann & Rousseeuw 1990).

One of the most frequently used techniques of hierarchical clustering is the Ward method. Ward (1963) proposed a clustering procedure seeking to form the partitions $P_k, P_{k-1}, \ldots, P_1$ in a manner that minimizes the loss associated with each grouping and to quantify that loss in an interpretable form. Information loss is defined by Ward in terms of an error sum-of-squares (ESS) criterion. The ESS is defined as follows

$$ESS = \sum_{k=1}^{K} \sum_{x_i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2$$

with the cluster mean $\bar{x}_{kj} = \frac{1}{n_k} \sum_{x_i \in C_k} x_{ij}$, where x_{ij} denotes the value for the i-th individual in the j-th cluster, k is the total number of clusters at each stage, and n_j is the number of individuals in the j-th cluster.

Non hierarchical clustering possesses a monotonically increasing ranking of strengths as clusters themselves progressively become members of larger clusters. The values of the strength may be compared to arrive at a final solution to the clustering problem. On the other hand non hierarchical clustering methods do not posses tree-like structures and new clusters are formed in successive clustering either by merging or splitting clusters. Non-hierarchical techniques include K-means, adaptive K-means, K-medoids, fuzzy clustering, etc.

For the illustration, we use the `bank.dat` data set taken from Flury & Riedwydl (1988). This data set consists of 200 measurements on Swiss bank notes. One half of these bank notes are genuine, the other half are forged

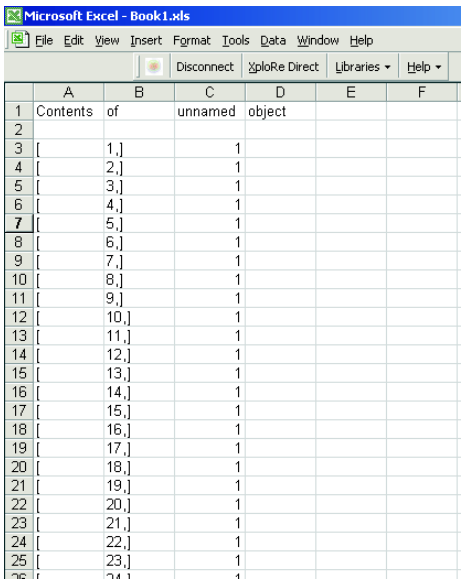

Figure 8: The Ward method with the Swiss Bank Data

bank notes. The data is already packed in XQS. We can also get the data from the MS Excel sheet as in the GARCH example above. From the dialogue menu we can select the method of cluster analysis we want to use. Before we execute, we must determine how many clusters we want to construct. The result of the cluster analysis using WARD method and three clusters is shown in Figure 8.

References

Bollerslev, T. (1986). *Generalized Autoregressive Conditional Heteroskedasticity*, Journal of Econometrics, 31:307-327.

Engle, R.F. (1982). *Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation*, Econometrica, 50:987-1008.

Feuerhake, J. (2001). *MD*CRYPT - The XQS/XQC protocol*, online available at http://md-crypt.com/.

Flury, B. & Riedwyl, H. (1988). *Multivariate Analysis, A Practical Approach*, Cambridge University Press.

Gouriéroux, C. (1997). *ARCH Models and Financial Applications*, Springer-Verlag, New York.

Härdle, W., Klinke, S. & Müller, M. (1999). *XploRe Learning Guide*, Springer Verlag, Heidelberg.

Härdle, W., Hlavka, Z. & Klinke, S. (2000). *XploRe Application Guide*, Springer Verlag, Heidelberg.

Härdle, W., Kleinow, T. & Tschernig, R. (2001). *Web Quantlets for Time Series Analysis*, The Annals of Mathematical Statistics, 53(1), 179-188.

Kaufman, L. & Rousseeuw, P.J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley, New York.

Kleinow, T., & Lehman, H. (2001). *Client/Server based Statistical Computing*, forthcoming in Computational Statistics Special Issue, Springer Verlag, Heidelberg .

Kleinow, T., & Thomas, M. (2000). *Computational Resources for Extremes*, in J. Franke, W. Hrdle and G. Stahl (eds), *Measuring Risk in Complex Stochastic Systems*, 147, Lecture Notes in Statistics, Springer-Verlag, New York.

Knüsel, L. (1998). *On the Accuracy of Statistical Distributions in Microsoft Excel*, The Statistical Software Newsletter in Computational Statistics & Data Analysis, 26, 375-379.

McCullough, B.D. & Wilson, B. (1999). *On the Accuracy of Statistical Procedures in Microsoft Excel*, Computational Statistics & Data Analysis, 31, 27-37.

Neuwirth, E. & Baier, T. (2001). *Embedding R in Standard Software, and the other way round*, Proceedings of the 2nd International Workshop on Distributed Statistical Computing, March 15-17, Vienna, in K. Hornik & F. Leisch (eds), ISSN 1609-395X.

Ward, J.H. (1963). *Hierarchical grouping to optimize an objective function*, Journal of the American Statistical Association, 58, 236-244.