

Aydinli, Gökhan

Working Paper

Net based spreadsheets in quantitative finance

SFB 373 Discussion Paper, No. 2002,42

Provided in Cooperation with:

Collaborative Research Center 373: Quantification and Simulation of Economic Processes,
Humboldt University Berlin

Suggested Citation: Aydinli, Gökhan (2002) : Net based spreadsheets in quantitative finance, SFB 373 Discussion Paper, No. 2002,42, Humboldt University of Berlin, Interdisciplinary Research Project 373: Quantification and Simulation of Economic Processes, Berlin,
<https://nbn-resolving.de/urn:nbn:de:kobv:11-10051473>

This Version is available at:

<https://hdl.handle.net/10419/65341>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

Net Based Spreadsheets in Quantitative Finance

Gökhan Aydınlı

1 Introduction

Modern risk management requires accurate, fast and flexible computing environments. To meet this demand a vast number of software packages evolved over the last decade, accompanying a huge variety of programming languages, interfaces, configuration and output possibilities. One solution especially designed for large scale explorative data analysis is XploRe, a procedural programming environment, equipped with a modern client/server architecture (Härdle et al. (1999) and Härdle et al. (2000)).

As far as flexibility in the sense of openness and accuracy is concerned XploRe has a lot to offer a risk analyst may wish. On the contrary its matrix oriented programming language (Fickel, 2001) might be seen as a drawback in respect to other computational approaches. In terms of learning curve effects and total cost of ownership an alternative solution seems desirable.

This chapter will present and demonstrate the net based spreadsheet solution MD*ReX designed for modern statistical and econometric analysis. We concentrate on examples of Value-at-risk (VaR) with copulas and means of quantifying implied volatilities presented by Rank and Siegl (2002) and Fengler et al. (2002). All results will be shown in Microsoft Excel.

Recent research work suggests that the rationale for spreadsheet based statistical computing is manifold. Ours is to bring state-of-the-art quantitative methods to the fingertips of spreadsheet users. Throughout this chapter we will give a short introduction into our underlying technology, briefly explain the usage of the aforementioned spreadsheet solution and provide applications of this tool.

2 Client/Server based Statistical Computing

While the power of computing equipment increased exponentially, statistical algorithms yielded higher efficiency in terms of computing time and accuracy. Meanwhile the development of high capacity network architectures had another positive impact on this trend, especially the establishment of the world wide web. Consequently a vast number of researchers and programmers in computational statistics as well as institutions like commercial banks, insurers and corporations spent much effort to utilize this evolution for their field of research. An outcome has been the technological philosophy of *client/server* based statistical computing: meaning a decentralized combination of methods, users and providers of statistical knowledge.

Our understanding of client/server based statistical computing is such that there exists a formal relationship between user, provider and vendor of statistical methodology. An easy to grasp example is a telephone call. The caller (in our case the user demanding statistical methods and/or advice) calls (connects via TCP/IP enabled networks) someone (a high-performance server/vendor of statistical information and methods) who serves his call (the requested calculation is done/information is displayed in a HTML browser, etc.). This client/server

understanding is an approach to gain scalability of computational tasks, resource shifting of processing power and decentralization of methods.

There are numerous ways of implementing client/server based statistics, among others Common Gateway Interfaces (CGI), JavaScript, Java Applets and Plug-Ins are the most commonly used techniques. The technology behind the XploRe client/server architecture is thoroughly explained in Kleinow and Lehmann (2002). While that solution is applet based the spreadsheet client presented here has an Add-in character. The MD*ReX-Client is a software tool, nested within an spreadsheet application. In both cases the communication technique relies on the protocol stack MD*CRYPT. For the spreadsheet solution the Java based MD*CRYPT has to be modified. As Microsoft does not support any Java natively for its Office suite, MD*CRYPT has been implemented as a dynamic link library to utilize its interface for Office applications like Excel. The technical aspects and the design philosophy of the MD*ReX-Client are discussed in detail in Aydınli et al. (2002).

3 Why Spreadsheets?

Since their first appearance in the late 1970s spreadsheets gained a remarkable popularity in business as well as research and education. They are the most common software managers, researchers and traders utilize for data analysis, quantitative modelling and decision support.

Przasnyski and Seal (1996) find that most of the time series modeling done in the business world is accomplished using spreadsheets. Further research work suggests that those users have become so proficient with spreadsheets that they are reluctant to adopt other software solutions. Not even a higher suitability for specific applications is appealing then (Chan and Storey, 1996). An analysis of XploRe download profiles conducted in a data mining framework confirmed our perception of the statistical software market. A stunning majority of users are using Excel for statistical analysis (Sofyan and Werwatz, 2001).

A major difference between a spreadsheet application and statistical programming languages is the interaction model. This "*direct manipulation interaction model*" enables statistical computations e.g. by drag and drop (Neuwirth and Baier, 2001). In the cell based framework of spreadsheets the direct aspect of interaction means, that manipulations in one cell immediately effect the content of another cell. Of course this is only the case if the regarding cells are interconnected with appropriate cell functions and references. Especially in business applications the immediate visibility of numerical changes when cell values are modified, is an appreciated feature for decision making based on different scenarios.

Our approach is based on the philosophy to bring two ideals together: On the one hand an accurate and reliable statistical engine (which is represented by the XploRe Quantlet Server, *XQS*) and on the other a user friendly and intuitive Graphical User Interface like Excel. The numerical impreciseness of Excel has been discussed exhaustively in the literature. As a starting point the reader is referred to e.g. McCullough and Wilson (1999). The development of MD*ReX is guided by the principles of usability and flexibility. Hence we try to offer different modes of usage for varying user groups: a dialogue based and more "Windows" like appearance for the novice user of statistical software and

a "raw" mode where the spreadsheet application merely functions as a data import wizard and scratch-pad for the advanced user.

4 Using MD*ReX

We will demonstrate the usage of MD*ReX in the context of quantitative financial modeling. In order to use MD*ReX a working installation of Excel 9.x is required. Furthermore the installation routine will setup a Java runtime environment and if needed a Virtual Java Machine. For more information please refer to <http://md-rex.com/>.

After successfully installing the client it can be used in two ways: for an on-demand usage MD*ReX can be accessed via the Start → Programs shortcut in Windows or if a permanent usage in Excel is requested, the Add-in can be installed from the Extras → Add-in Manager dialogue in Excel. The `rex.xla` file is located under

`%Systemroot%\%Program Files%\MDTech\ReX\ReX.`

In the latter case the client is available every time Excel is started. Anyway the client can be accessed via the Excel menu bar (Figure 1) and exposes its full functionality after clicking on the ReX menu item.

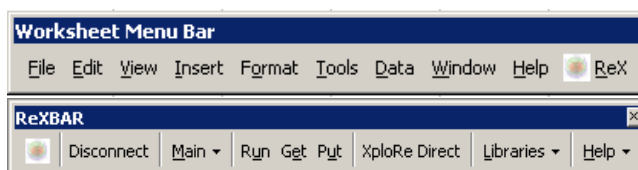


Figure 1: Excel and MD*ReX menus

In order to work with MD*ReX the user first has to connect herself to a running XploRe Quantlet Server. This can either be a local server, which by default is triggered if MD*ReX is started via the *Programs* shortcut, or any other XQS somewhere on the Internet. Evidently for the latter option a connection to the Internet is required. The *Connect* dialogue offers some pre-configured XQS'. After the connection has been successfully established the user can start right away to work with MD*ReX.

In contrast to XploRe, the user has the option to perform statistical analysis by using implemented dialogues e.g. the Time Series dialogue in Figure 3. Via this dialogue a researcher is able to conduct standard time series analysis techniques as well as e.g. more refined nonlinear approaches like ARCH tests based on neural networks. These interfaces encapsulate XploRe code while using the standard Excel GUI elements hence undue learning overhead is minimized. Alternatively one can directly write XploRe commands into the spreadsheet cells and then let these run either via the menu button or with the context menu, by right clicking the highlighted cell range (Figure 2). Furthermore it is now much easier to get data to the XploRe Quantlet Server. Simply marking an appropriate data range within Excel and clicking the *Put* button is enough to transfer any kind of numerical data to the server. We will show this in the

next section. A further virtue of using a spreadsheet application is the commonly built-in database connectivity. Excel for example allows for various data retrieval mechanisms via the Open Database Connectivity (ODBC) standard, which is supported by most of the database systems available nowadays.

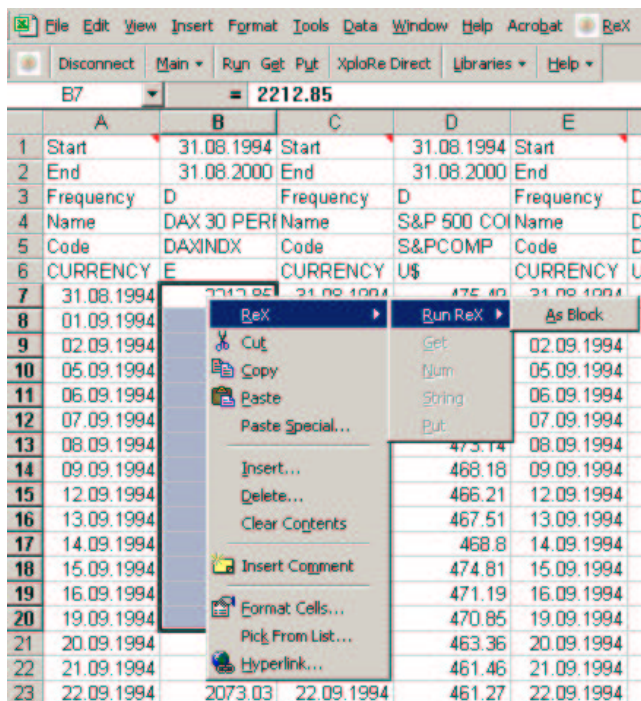


Figure 2: ReX Context Menu

5 Applications

In the following paragraph we want to show how MD*ReX might be used in order to analyze the VaR using copulas as described by Rank and Siegl (2002). Subsequently we will demonstrate the analysis of implied volatility shown by Fengler et al. (2002). All examples are taken out of this book and have been accordingly modified. The aim is to make the reader aware of the need of this modification and give an idea how this client may be used for other fields of statistical research as well.

We have willingly omitted the demonstration of dialogues and menu bars as it is pretty straightforward to develop these kind of interfaces on your own. Some knowledge of the macro language Visual Basic for Applications (VBA) integrated into Excel and an understanding of the XploRe Quantlets is sufficient to create custom dialogues and menus for this client. Thus no further knowledge of the XploRe Quantlet syntax is required. An example is the aforementioned Time Series dialogue, Figure 3.

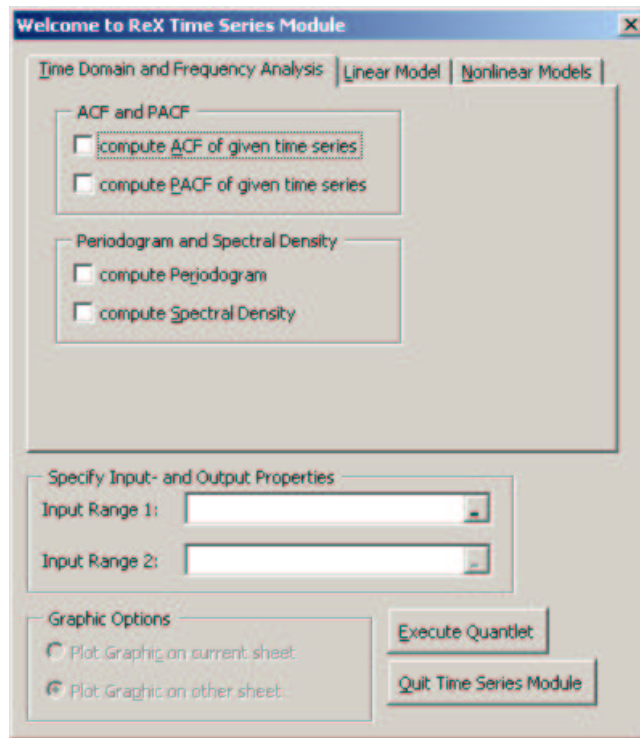


Figure 3: MD*ReX Time Series Dialogue

5.1 Value at Risk Calculations with Copulas

The quantification of the VaR of a portfolio of financial instruments has become a constituent part of risk management. Simplified the VaR is a quantile of the probability distribution of the value-loss of a portfolio, Rank and Siegl (2002). Aggregating individual risk positions is one major concern for risk analysts. The $\mu - \sigma$ approach of portfolio management measures risk in terms of the variance, implying a "Gaussian world" (Bouyé et al., 2001). Traditional VaR methods are hence based on the normality assumption for the distribution of financial returns. Though empirical evidence suggests high probability of extreme returns ("*Fat tails*") and more mass around the center of the distribution (*leptokurtosis*), violating the principles of the Gaussian world (Rachev, 2001).

In conjunction with the methodology of VaR these problems seem to be tractable with copulas. In a multivariate model setup a copula function is used to couple joint distributions to their marginal distributions. The copula approach has two major issues, substituting the dependency structure, i.e. the correlations and substituting the marginal distribution assumption, i.e. relaxation of the Gaussian distribution assumption. With MD*ReX the user is now enabled to conduct copula based VaR calculation with Excel, making use of Excel's powerful graphical capabilities and its intuitive interface.

The steps necessary are as follows:

1. Get the according Quantlets into Excel,

2. run them from there,
3. obtain the result,
4. create a plot of the result.

The first step is rather trivial: copy and paste the example Quantlet [XFGrexcopula1.xpl](#) from any text editor or browser into an Excel worksheet.

Next mark the range containing the Quantlet and apply the *Run* command. Then switch to any empty cell of the worksheet and click *Get* to receive the numerical output *rexcuv*. Generating a tree-dimensional Excel graph from this output one obtains an illustration as displayed in Figure 4. The according Quantlets are [XFGrexcopula1.xpl](#), [XFGrexcopula2.xpl](#), [XFGrexcopula3.xpl](#) and [XFGrexcopula4.xpl](#). They literally work the same way as the [XFGaccvar1.xpl](#) Quantlet.

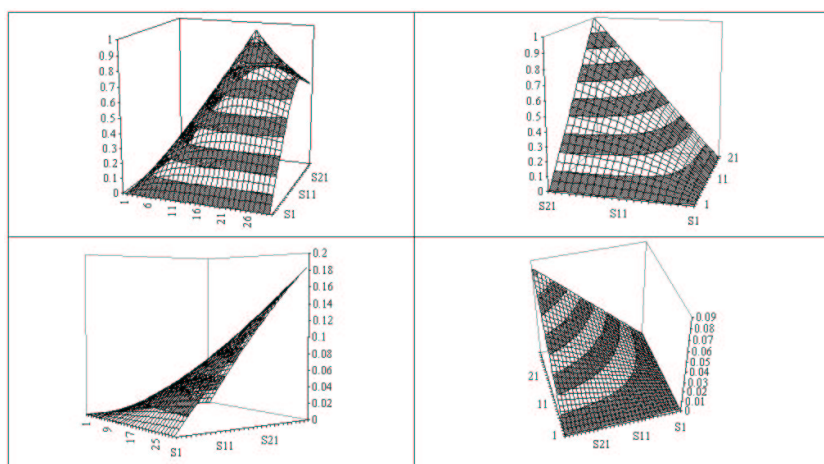


Figure 4: Copulas: $C_4(u, v)$ for $\theta = 2$ and $N = 30$ (upper left), $C_5(u, v)$ for $\theta = 3$ and $N = 21$ (upper right), $C_6(u, v)$ for $\theta = 4$ and $N = 30$ (lower left), $C_7(u, v)$ for $\theta = 5$ and $N = 30$ (lower right)

Of course the steps 1-4 could easily be wrapped into a VBA macro with suitable dialogues. This is exactly what we refer to as the change from the raw mode of MD*ReX into the "Windows" like embedded mode. Embedded here means that XploRe commands (quantlets) are integrated into the macro language of Excel.

The Monte Carlo simulations are obtained correspondingly and are depicted in Figure 5. The according Quantlet is [XFGrexmccopula.xpl](#). This Quantlet again is functioning analogous to [XFGaccvar2.xpl](#). The graphical output then is constructed along same lines: paste the corresponding results *z11* through *z22* in cell areas and let Excel draw a scatter-plot.

5.2 Implied Volatility Measures

A basic risk measure in finance is *volatility*, which can be applied to a single asset or a bunch of financial assets (i.e. a portfolio). Whereas the *historic*

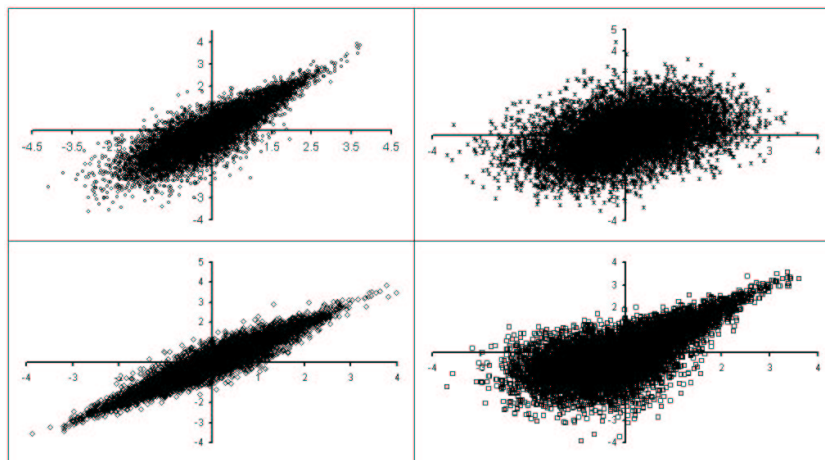


Figure 5: Monte Carlo Simulations for $N = 10000$ and $\sigma_1 = 1$, $\sigma_2 = 1$, $\theta = 3$

volatility simply measures past price movements the *implied volatility* represents a market perception of uncertainty. Implied volatility is a contemporaneous risk measure which is obtained by reversely solving an option pricing model as the Black-Scholes model for the volatility. The implied volatility can only be quantified if there are options traded which have the asset or assets as an underlying (for example a stock index). The examples here are again taken out of (Fengler et al., 2002). The underlying data are `VolaSurf02011997.xls`, `VolaSurf03011997.xls` and `volsurfddata2`. The data has been kindly provided by MD*BASE. `volsurfddata2` ships with any distribution of XploRe. In our case the reader has the choice of either importing the data into Excel via the data import utility or simply running the command `data=read("volsurfddata2.dat")`. For the other two data sets utilizing the *Put* button is the easiest way to transfer the data to an XQS. Any of these alternatives have the same effect, whereas the former is a good example of how the MD*ReX client exploits the various data retrieval methods of Excel.

The Quantlet `XFGReXiv.xpl` returns the data matrix for the implied volatility surfaces shown in Figure 6 through 8. Evidently the Quantlet has to be modified for the appropriate data set. In contrast to the above examples where Quantlets could be adopted without any further modification, in this case we need some redesign of the XploRe code. This is achieved with suitable reshape operations of the output matrices. The graphical output is then obtained by arranging the two output vectors `x2` and `y2` and the output matrix `z1`.

The advantage of measuring implied volatilities is obviously an expressive visualization. Especially the well known volatility smile and the corresponding time structure can be excellently illustrated in a movable cubic space. Furthermore this approach will enable real-time calculation of implied volatilities in future applications. Excel can be used as a data retrieval front end for real-time market data providers as Datastream or Bloomberg. It is imaginable then to analyze tick-data which are fed online into such a spreadsheet system to evaluate contemporaneous volatility surfaces.

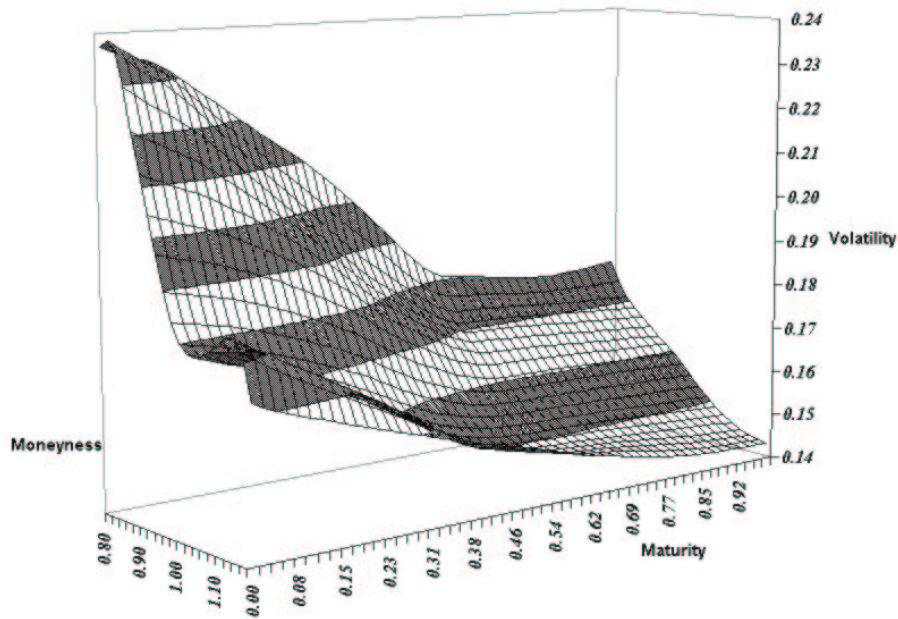


Figure 6: DAX30 Implied Volatility, 02.01.1997

Acknowledgement

The authors acknowledge support by the Deutsche Forschungsgemeinschaft via Sonderforschungsbereich 373 “Quantifikation und Simulation ökonomischer Prozesse” at Humboldt-Universität zu Berlin.

References

- Aydınlı, G., Härdle, W., Kleinow, T. and Sofyan, H.(2002). *ReX: Linking XploRe to Excel*, forthcoming Computational Statistics Special Issue, Springer Verlag, Heidelberg.
- Bouyé, E., Durrleman, V., Nikeghbali, A., Riboulet, G. and Roncalli, T.(2000). *Copulas for Finance, A Reading Guide and some Applications*, unpublished manuscript, Financial Econometrics Research Centre, City University Business School, London, 2000.
- Chan, Y.E. and Storey, V.C. (1996). *The use of spreadsheets in organizations: determinants and consequences*, Information & Management, Vol. 31, pp. 119-134.
- Fengler, M., Härdle, W. and Schmidt, P. (2002). The Analysis of Implied Volatilities, in W. Härdle, T. Kleinow and G. Stahl (eds), *Applied Quantitative Finance*, Springer Finance, Springer Verlag.

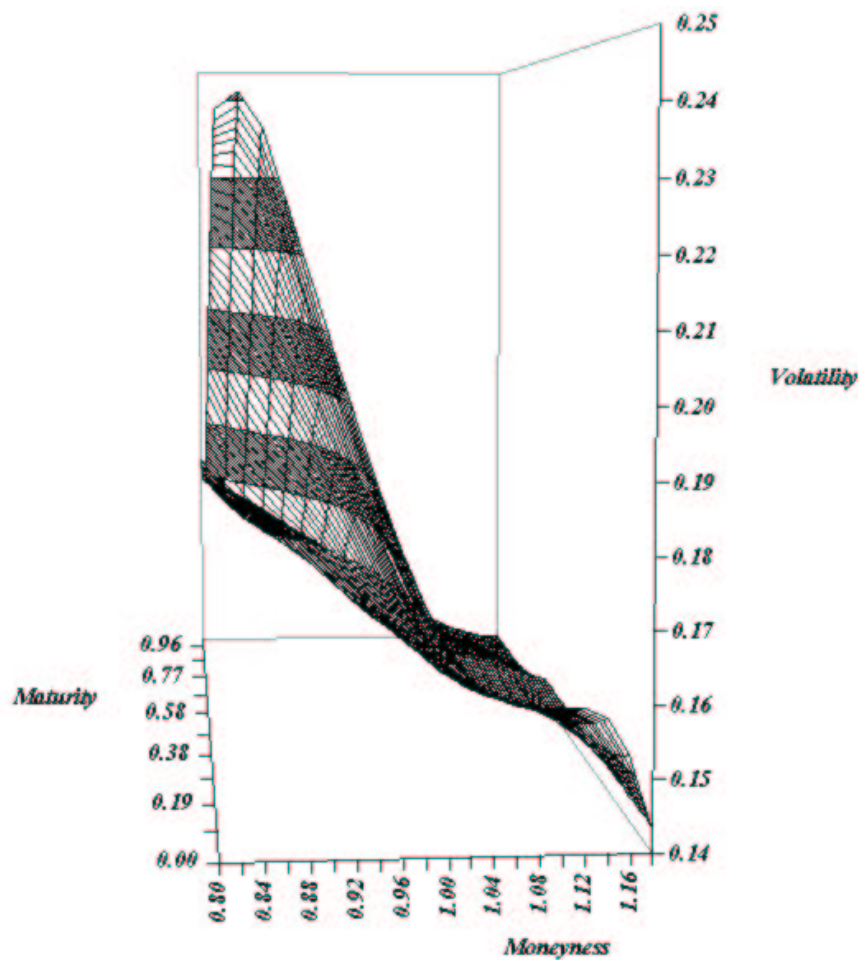


Figure 7: DAX30 Implied Volatility, 03.01.1997

- Fickel, N. (2001). *Book Review: XploRe - Learning Guide*, Allgemeines Statistisches Archiv, Vol. 85/1, p. 93.
- Härdle, W., Klinke, S. and Müller, M. (1999). *XploRe Learning Guide*, Springer Verlag, Heidelberg.
- Härdle, W., Hlavka, Z. and Klinke, S. (2000). *XploRe Application Guide*, Springer Verlag, Heidelberg.
- Kleinow, T. and Lehmann, H. (2002). *Client/Server based Statistical Computing*, forthcoming in Computational Statistics Special Issue, Springer Verlag, Heidelberg.
- McCullough, B.D. and Wilson, B. (1999). *On the Accuracy of Statistical Procedures in Microsoft Excel*, Computational Statistics & Data Analysis, Vol. 31, p. 27-37.

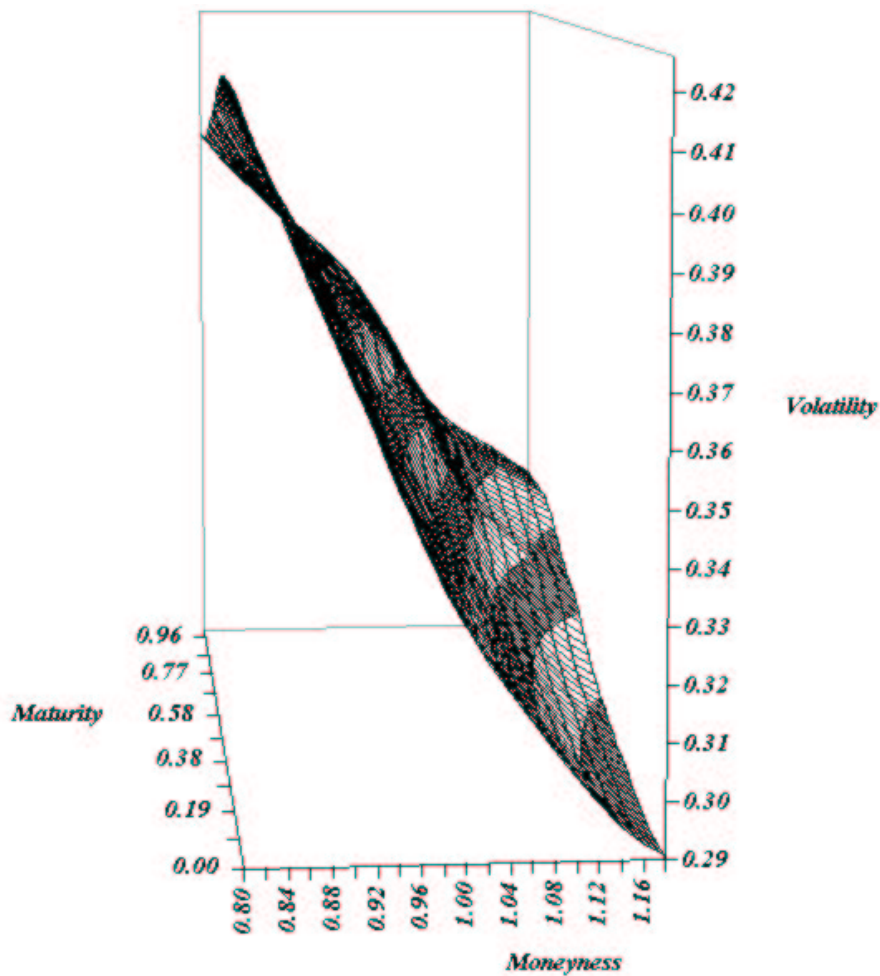


Figure 8: DAX30 Implied Volatility

- Neuwirth, E. and Baier, T. (2001). *Embedding R in Standard Software, and the other way round*, DSC 2001 Proceedings of the 2nd International Workshop on Distributed Statistical Computing.
- Przasnyski, L.L. and Seal, K.C. (1996). *Spreadsheets and OR/MS models: an end-user perspective*, Interfaces, Vol. 26, pp. 92-104.
- Rachev, S. (2001). *Company Overview*, Bravo Consulting, http://www.bravo-group.com/inside/bravo-consulting_company-overview.pdf.
- Ragsdale, C.T. and Plane, D.R. (2000). *On modeling time series data using spreadsheets*, The International Journal of Management Science, Vol. 28, pp. 215-221.

Rank, J. and Siegl, T. (2002). Applications of Copulas for the Calculation of Value-at-Risk. *in* W. Härdle, T. Kleinow and G. Stahl (eds), *Applied Quantitative Finance*, Springer Finance, Springer Verlag.

Sofyan, H. and Werwatz, A. (2001). *Analysing XploRe Download Profiles with Intelligent Miner*, Computational Statistics, Vol. 16, pp. 465-479.