Large Project 2011

This year's Large Project will be a joint project for all students participating in the course. In this project we shall write a "professional" package for evaluating contingent claims on the FX and Fixed Income markets. The term "professional" means that, contrary to the standard courseworks, this software has to be written for an end-user who uses financial instruments in its everyday operations and is not an expert in numerical analysis and computational finance. The project will be the continuation of the project from the last year. The project is a library of functions performing different tasks. A number of functions is already available. We shall only modify some of existing functions and write few new functions.

The main task of the project is to extend the main module (main programme) which manages all functions, supplies data to them, provides the communication between the functions and solves the problem of communication with an end-user interface. The version of this module written in web-octave is ready but it needs a suitable adjustment and extention.

The present version of the main module is prepared to operate with uploaded real market data and it requires creation of individual accounts for users (with users logins and passwords). On the other hand, we have a number of functions which use analytic formulas derived from the Black-Scholes model. These functions can be used with data entered from keyboard and produce immediate results. The idea is to split the present main module into two separate sub-modules: one – the present module – which uses individual users accounts and performs complicated calculations with market data provided by the user as separate files (this module should produce real market prices of traded instruments), and the second one which can be accessible on-line by everybody and calculate prices of simple instruments (that which possess analytic formulas for prices) after entering necessary data from keyboard.

In the course of the project the following tasks are essential:

1. Extension of the existing main module. Part of the present version of the module is written in "python-django". This part manages the input and processing of data (together with some verification procedure). On the other hand, not so many instruments calculations are implemented. Hence the main task is to construct a structure in which new instruments can be easily added, and then add those instruments for which relevant octave functions are already prepared.

2. An independent task is to write a communication module (also in "python-django") which can perform calculations for data entered from keyboard. This module can be partially similar to the existing main module, but it should have additional functionality of calling octave functions which perform calculations.

The above tasks which are oriented toward producing a working web-page should be complemented with some modifications of existing functions and writing a few specific functions.

1. Processing input data – term structure of implied volatilities. Available volatilities are implied volatilities for a very limited number of compound options. The existing function produces a volatility smile for the simple interpolation method called the Vanna-Volga method. This method works well for flat volatility smiles but is not accurate for "real smiles". We have to implement a procedure by which a volatility smile is recovered better from data (an appropriate algorithm is available in the literature). In addition we have to implement interpolation procedure between available dates. A simple linear interpolation is already implemented but more sophisticated interpolation is needed.

2. Processing input data – term structure of interest rates. Recovering the term structure of interest rates from the prices of traded instruments (LIBOR rates, FRA rates and swap rates) is complicated because it requires full use of the calendar. The existing functions are based on a term structure of discount factors. These discount factors are calculated from real market quotations. We need only proper interpolation of rates. A number of such interpolation methods is already implemented. We shall enrich it by some new methods.

3. Construction of functions calculating prices of different instruments from the Black-Scholes formula or simple parity considerations. We need functions for pricing structured products on FX market.

Documentation. It is absolutely essential to describe carefully the theoretical background of every function. But also the documentation of the computer code has to be provided. A particular attention should be put on the structure of the programme (block scheme), the format of input and output data, signaling choices, etc.

The project will consist of the following tasks:

1. Extend communication module for logged users.

2. Writting communication module for open access.

3. Processing input data — new interpolation methods for discount factors.

4. New interpolation method for construction of implied volatility surface.

5. Pricing instruments from the Black-Scholes formula, parity identity and time value of money considerations.

Literature comments

- Preparing discount factors table — knowledge for financial engineering, and numerical mathematics (spline), but you can also consult the course coordinators.
- Construction of implied volatility surface — general idea is in the book by Gatheral, but the paper by Bossens at al. is essential.
- Pricing instruments from the Black-Scholes formula — knowledge for financial engineering and from the book by Wystup. List of instruments will be provided by the course coordinators.

References

The is one fundamental reference about instruments on FX market:

1. Uwe Wystup – *FX Options and Structured Products*, J. Wiley 2007.

The book of Wystup contains many practical informations. It is in some form available from the course coordinators.

In addition there are plenty of books and papers devoted to specific problems.

1. Jim Gatheral – *The Volatility Surface*, J. Wiley 2006.

2. Justin London – *Modeling Derivatives in C++*, Wiley 2005. (Library)

3. Frédéric Bossens, Grégory Rayéey, Nikos S. Skantzos and Griselda Deelstra – Vanna-Volga methods applied to FX derivatives: from theory to market practice. (pdf)

4. Patrick S. Hagan and Graeme West – Interpolation methods for curve construction, *Applied Mathematical Finance*, **13** (2006), 89–129.

5. Andrew Lesniewski – Interest rates and FX models: The Forward Curve. (pdf)