

Introducing OpenOffice Calc plugin for JQuantlib

Praneet Tiwari, ISV Engineering, Sun Microsystems

April 3, 2009

Abstract

JQuantLib , founded by Richard Gomes in September 2007, is a quantitative finance library written in Java. The library is remarkable in many ways, one being the use of Java, something quite uncommon in the world of Quantitative Finance. This article introduces integration of JQuantlib with Open Office. The other approaches of making JQuantlib easy to use are listed and briefly analyzed. The functionalities available in version 0.1 of the Calc plugin are detailed next. Most of the functions available now are elementary. It is hoped that once the code is committed, the community will take the plugin forward to make the experience of JQuantlib users more useful and pleasurable.

Why a Calc plugin?

A library can be useful only when it is turned into an application. With an open source quantitative finance library, the challenges are further compounded. People want to evaluate a software quickly to find out what it can do for them. Not many have the time and patience to go through the source code. A verified integration of a library with other popular open source application goes a long way in assuring the end-users of the extensibility of a library.

Since JQuantlib is written in Java, there are numerous ways to provide cross-platform GUI. Some of the prominent ones being -

- Swing front end
- JavaFX front end
- Web based (may be a lot AJAX thrown in) front end
- Jython/Python front end
- OpenOffice Calc spreadsheet front end

A calc plugin was chosen over others because

- OpenOffice usage is on the rise. Since we have more and more people using OOo, it makes sense to capitalize on the opportunity.
- OpenOffice is open source. Nothing like combining two open source apps to get a more powerful one.
- Swing and Web based solutions are more suited for development and deployment in company that has its objective sorted out. Making a general purpose, one size fits all GUI for a library as extensive as JQuantlib will definitely make the presentation quite cluttered.
- Development and enhancement of the OOo plugin is less costlier when compared to other clients. Part of the reason is that with OOo calc plugin the front-end design phase is short-circuited. OOo does all the hard work and developers can focus on the main task at hand.

- A OOO calc plugin offers an excellent opportunity for a piecemeal, incremental and distributed development. Each function of the plugin can be developed independently of others, something which is invaluable in open source world.
- Last but not the least, with Calc there is no dependency on third-party chart libraries such as JFreeChart. This certainly makes the solution leaner and at the same does not restrict the visualization formats.

Version 0.1 - a first look

My intention with version 0.1 was to kickstart the process of JQuantlib plugin. Rather than trying to save the world, I have limited myself to the most elementary functions. As with all open source applications, the hope here is that community will accept and carry forward the work. As of now, the plugin has following functionalities -

1. BlackScholes European option pricing. Function name - JQEUROPEAN-BLACKSCHOLES()
2. Generation of prime numbers. Function name - JQGETPRIMENUMBERAT(index)
3. Generation of PoissonDistribution values.
4. Generation of NormalDistribution values.
5. Generation of NonCentralChiSquaredDistribution values.
6. Generation of InverseCumulativePoisson Distribution values.
7. Generation of InverseCumulativeNormal Distribution values.
8. Generation of Gamma distribution values.
9. Generation of CumulativePoissonDistribution values.
10. Generation of CumulativeNormalDistribution values.
11. Generation of BinomialDistributionValue values.
12. Generation of Factorials and log of factorials.

BlackScholes - the “Hello World!” of Quantitative Finance

No quant finance library or plugin can ever be considered complete without the implementation of the celebrated BlackScholes formula. So here is how it looks as of now -

BlackScholes.ods - OpenOffice.org Calc

File Edit View Insert Format Tools Data Window Help

Arial Black 10

This sheet shows the output of Jquantlib's implementation of BS formula

	A	B	C
1	This sheet shows the output of Jquantlib's implementation of BS formula		
2			
3			
4	Strike	40.00	
5	Underlying	38.00	
6	RiskFreeRate(p.a)	0.06	
7	Volatility(p.a)	0.20	
8	DividendYield(p.a)	0.00	
9	OptionType	call	
10	EvaluationDay	15	
11	EvaluationMonth	1	
12	EvaluatuionYear	2009	
13	MaturityDay	15	
14	MaturityMonth	2	
15	MaturityYear	2009	
16			
17	BlackScholesValue	0.28	
18			
19			
20	Strike	40.00	
21	Underlying	38.00	
22	RiskFreeRate(p.a)	0.06	
23	Volatility(p.a)	0.20	
24	DividendYield(p.a)	0.00	
25	OptionType	put	
26	EvaluationDay	15	
27	EvaluationMonth	1	
28	EvaluatuionYear	2009	
29	MaturityDay	15	
30	MaturityMonth	2	
31	MaturityYear	2009	
32			
33	BlackScholesValue	2.07	

BlackScholes BSInterestRateChanges BSInterestGraph BSUnderlyingChange BSUnderlyingGraph

Figure 1: BlackScholes Calculations

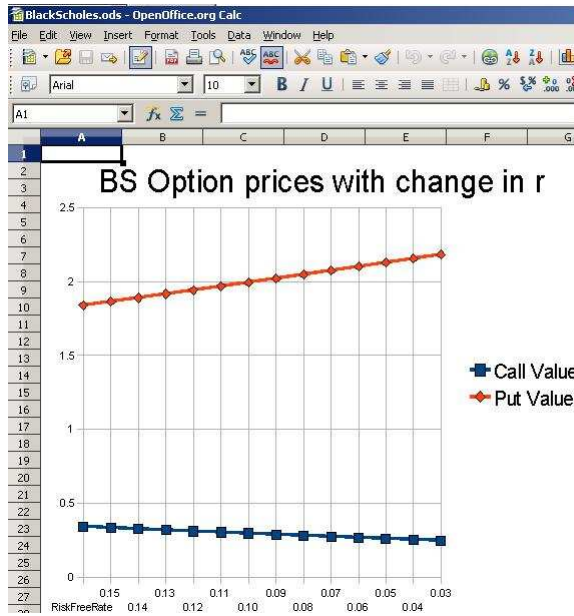


Figure 2: Variation of Prices with Riskfreerate

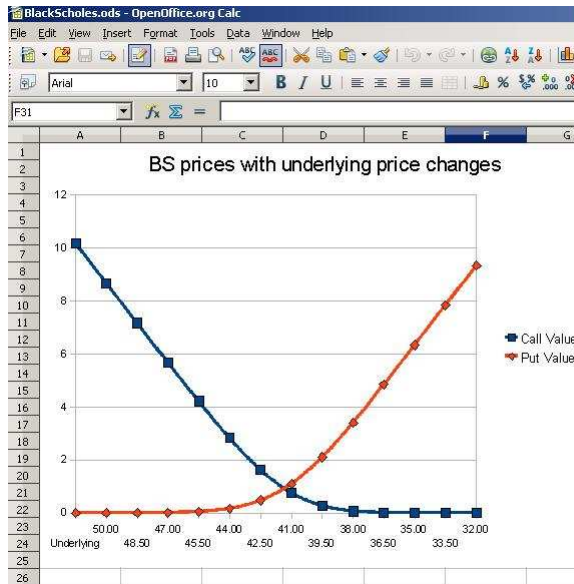


Figure 3: Variation of Prices with changes in Underlying

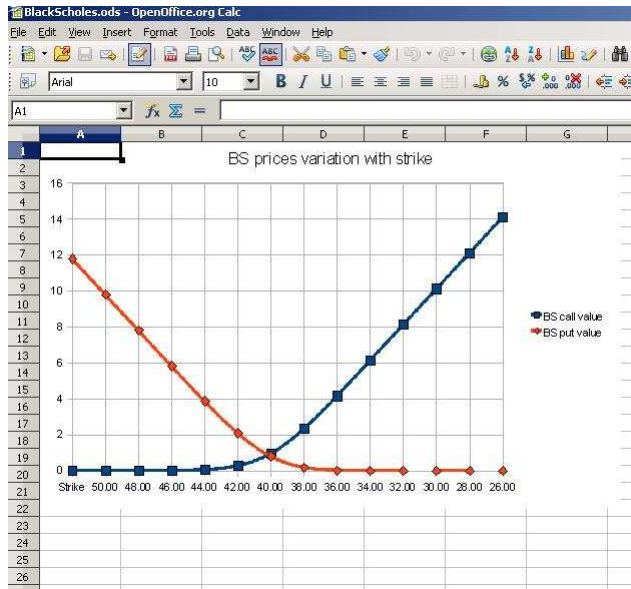


Figure 4: Variation of Prices with changes in Strike Price

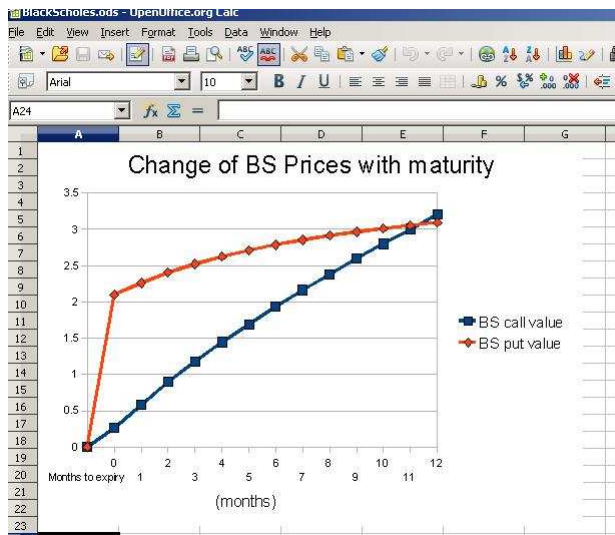


Figure 5: Variation of Prices with changes in Maturity

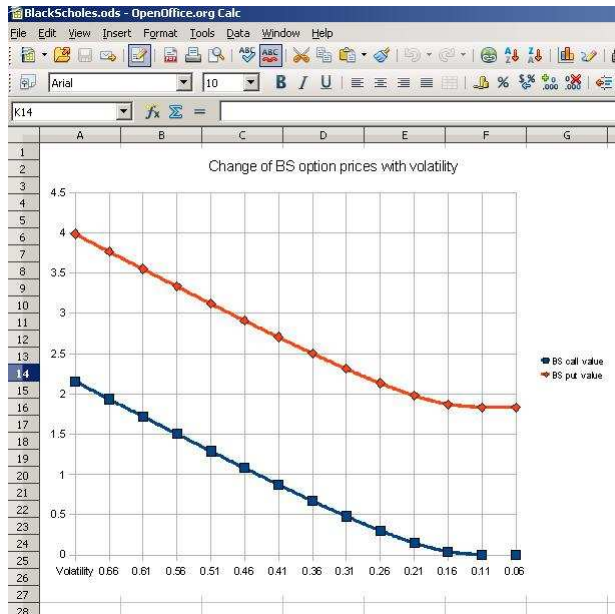


Figure 6: Variation of Prices with changes in Volatility

You can find the worksheets at

<http://blogs.sun.com/praneet/resource/BlackScholes.ods>

<http://blogs.sun.com/praneet/resource/BlackScholes.pdf>

<http://blogs.sun.com/praneet/resource/NumberOperations.ods>

Extending the plugin

This plugin can be extended in a number of ways. Some of the low hanging fruits are

1. Adding a menu item for JQuantlib
2. Adding American style option value calculation. JQuantlib already has this code.
3. Adding Black Formula calculation. JQuantlib already has this code.
4. Adding functions for Interest Rate Derivatives, the back-end code is about to be checked in.
5. Graphical depiction of binomial and trinomial trees.
6. Monte Carlo Simulation
7. Building the zero curve
8. Documentation. Getting a user manual that details all the functions and parameters that they take along with possible categorization.

Obviously, this list is not exhaustive and developers will use their creativity and imagination to contribute changes that are not mentioned here.