Quant Perspectives

Quantitative Finance on a Smartphone: New Apps and Unique Challenges

Quantitative finance will go mobile! How, you ask? Let's take a look at an example, and then explore some key differences that separate smartphone apps from conventional software.

By Joe Pimbley

The next step in the evolution of quant mobile apps is coming. Imagine, for example, an app that allows you to perform Black-Scholes computations on your phone.

In an earlier *SmartFinance* article, we described the novelty and utility of performing quantitative finance calculations on a smartphone. Even though handheld devices have evident disadvantages relative to large screens with desktop servers and other alternatives, the convenience and immediacy of smartphones are powerful.

This earlier article demonstrated a "prime number challenge" as the vehicle for *Monte Carlo* type calculation. (In addition to the article, see this short video that explains the freely available iOS app.) While *PrimeGenius* does test some important calculation methods, the logical leap forward is to create a more familiar financial algorithm.

Black-Scholes on a Smartphone

We've built in the *Swift* language a *BlackScholes* iOS app. We show below a screenshot of the user interface.

Figure: User Interface for iOS App for Black-Scholes Equity Option Calculation

The code merely performs the standard analytical call and put option calculations. (See, for example, the iconic publication of Fischer Black and Myron Scholes.)

The app retrieves the last traded price, dividend yield and issuer name for the user-provided ticker. Find *BlackScholes* (one word) by searching the App Store. (At time of writing, this is the only "free download" for Black-Scholes.) See also this short video demonstration of *BlackScholes*.

Joe Pimbley

Data Retrieval and User Inputs

Many readers of this column have long experience with Black-Scholes. Writing these formulas in *Swift* differs little from writing them on a spreadsheet or in familiar languages.

The greater learning curves for mobile app development involve bringing in necessary data and enabling user inputs. *BlackScholes* asks the user to specify a stock ticker to initiate retrieval of some necessary information.

Searching for a method to retrieve market data from a ticker, I found *BigBoard* – an elegant *Swift* library to assist iOS developers. *BigBoard* facilitates data gathering from a large public financial data website. This software handles much of the error processing and data quality assurance.

User Experience Dichotomy

A fascinating philosophical divide separates mobile apps and conventional professional software. The latter always entails an extended learning period.

Whether it's a *Bloomberg* terminal or *Intex Desktop* or the plethora of risk management software or even just *Microsoft Word*, conventional apps require days or weeks of "apprenticeship." The better applications within this class are sufficiently intuitive to let the user do something useful immediately (such as typing a few sentences into *Word*). But becoming an expert user requires time.

With mobile apps, however, the onus is on the developer/designer to engage the user *immediately*. In this world, the appearance of the app cannot give the subliminal message of "this will take time to figure out." Mobile users don't do learning curves that extend for days.

Why the difference? An easy answer may be that mobile users expect the apps to be games or functions that are immediately helpful; this is part of the psychology and enthusiasm of mobile use.

Another good answer is that professional software seeks deeply expert determinations of, say, cash flow projections of a complex CLO waterfall. If it takes days or weeks to learn CLO waterfalls, perhaps it must also take days or weeks to learn the app for the waterfalls.

Quant Finance Apps as Evolution

While we cannot refute these answers, neither do we accept them. Our sense is that quantitative finance *will* move to smartphones. It's challenging now to see how that will happen. In the long term, I believe it will demand great creativity and investment in improving the user experience.

In the short term, the most successful quant finance mobile apps should be adaptations of existing professional software. A great example is the *Bloomberg App*. App users learn the desktop professional software in their careers, and then have virtually immediate facility with the mobile version. Though the *Bloomberg App* is primarily a conduit for information and data, adding financial calculations is an obvious evolution.

Joe Pimbley (FRM) is a financial consultant in his role as Principal of Maxwell Consulting, LLC. His expertise includes enterprise risk management, structured products, derivatives, investment underwriting, training and quantitative modeling. Find Joe's archive of previous GARP columns here and his collection of short code projects here.