SmartFinance: Quantitative Finance on a Smartphone

Quant finance professionals may soon add Swift to their repertoire of code languages.

Java, Swift (successor to Objective C), and C# are core languages for the development of mobile apps of the Android, iOS, and Windows operating systems, respectively. In my intellectual and networking travels through the worlds of financial risk and quantitative finance, I don't encounter "mobile app development" as a burning topic. Despite the seeming disconnect between demands for calculations and data and the simplicity of a smartphone, I do predict that mobile devices will surge into our work lives.

Ubiquity of Smartphones

We all see every day how our fellow humans interact with their phones. Frequency and data volume of communication by voice, text, and video has exploded. Whether the best term is "low barrier" or "instant gratification," people do not resist the urge to be productive (with news and data gathering or business interactions) or *unproductive* (with games or diverse trivia).

There's a psychology here that is both unstoppable and overwhelming. Essentially everything one may perform with desktop computers has migrated to laptops and now to mobile devices. Why should running models and managing data for financial institutions be different?

Seems like the wrong direction

Yet that picture is also counterintuitive. While creating or using tools of quantitative finance, we are most at home sitting at a spacious desk with multiple display screens. We have screens for input data and model output, screens for a user interface, and screens for source code. We query large databases and run CPU-crunching calculations. Our favorite stories revolve around the need for banks of servers in a remote and professionally managed

cloud. The prospect of reducing this activity to a smartphone seems both unlikely and insulting.

But don't underestimate the march of technology! Mobile devices have plenty of room to grow in terms of memory and computation. Even if just as a conduit for bringing data to the user from external sources and sending requests for calculations to remote servers, handheld tools are valuable.

Smartphone is the direction people want

The key advantage is the psychology. It's happened before. At the dawn of the personal computer (PC) almost forty years ago, there were many sceptics among the expert class. Why not use "dumb terminals," as we called them, connected to powerful mainframe computers? In this time-share mode, the user would access much more memory and have faster computation at lower cost than the "unreasonable alternative" of owning a PC with its own memory and CPU (central processing unit). Many experts laughed at themselves later while saying "people like the feeling of having their own computer."

Though it may be irrational on some level, let's take that as an element of human nature. The remoteness and (actual or perceived) lack of control and ownership of the device function are stark human disadvantages. Taking this statement, the challenge is to redesign and create methods for quantitative finance on mobile platforms.

Swift is a new language for quant finance

We noted earlier the roles of Java, Swift, and C# as mobile developer code. Many quants already work in Java and C# since these languages are not specific to mobile apps. But Swift is the emerging standard for iOS (Apple-based products). If my prediction of the emergence of *SmartFinance* proves accurate, then Swift will become a new productive language in our field.

To get this paradigm evolution moving, I conceived and built an iOS app to implement random number generation (RNG). RNG is the first

necessary element for Monte Carlo simulation. Not too surprisingly, it works! Swift has a workable RNG function and math library.

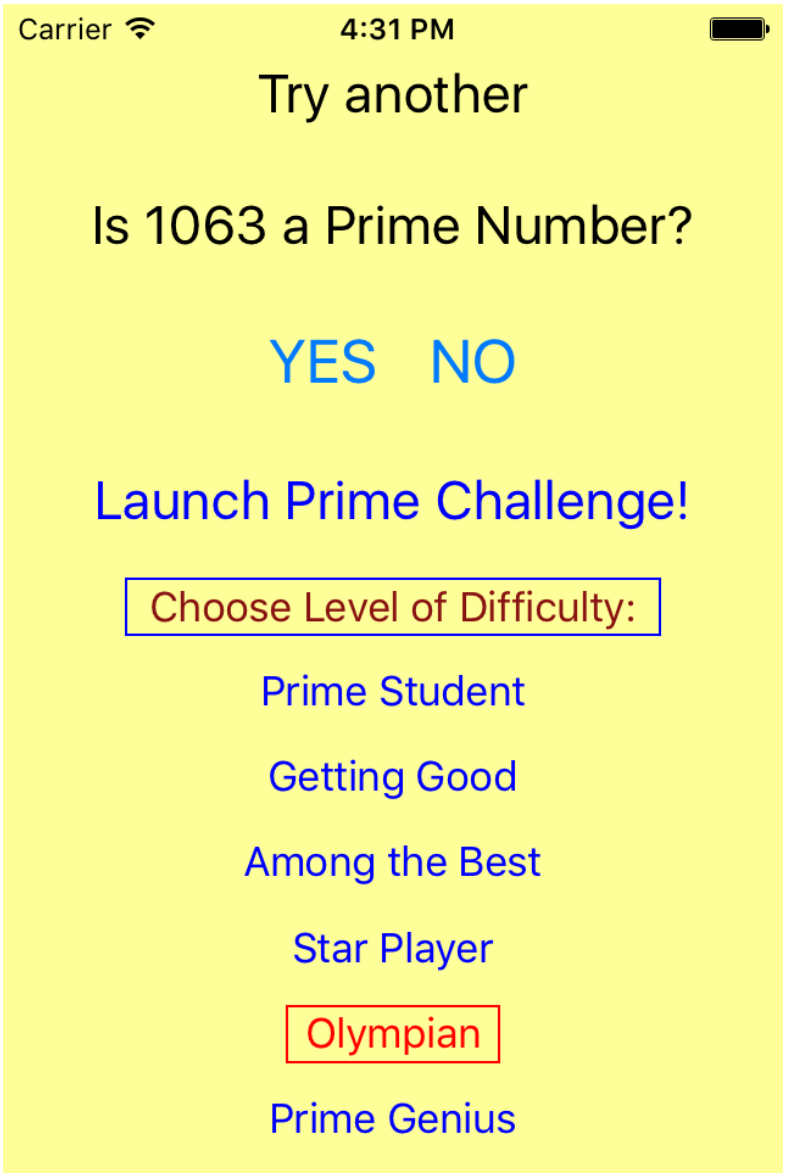

Figure: User Interface for iOS App with Rudimentary Monte Carlo simulation

The app itself, *PrimeGenius*, is an educational tool and game. We show above a screenshot of the iOS user interface. The code randomly generates prime and non-prime numbers with context-varying distribution functions. The difficulty level is adjustable to suit young elementary school students as well as (or beyond?) the most numerically intelligent readers of this article. Find *PrimeGenius* (one word) on the App Store.

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. Follow this link to get Joe's address to request addition to his E-mail List or to discuss your ideas for mobile apps!