Implied Volatility using Python's Pandas Library

Brian Spector

Thalesians Meetup London
15th January 2014

nag®
Experts in numerical algorithms and HPC services

Overview

- Motivation
- Python
- Pandas
- Implied Volatility
  - Timings in python
  - Different Volatility Curves
  - Fitting data points

Python

- Dynamically typed language
- Uses white spaces (as oppose to brackets) for control statements.
- Has grown in popularity:

Programming Language	2014	2009	2004	1999
Python Ranking	8	6	11	22

- http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html

Python

- Why use python?
  - Cheap
  - Easy to learn
  - Powerful

Python

- Why use python?
  - Cheap
  - Easy to learn
  - Powerful
- Why use python over R?
  - “I would rather do math in a programming language than programming in a math language.”

Python

- What python has:
  - Many built-in powerful packages
  - OO programming
    - Classes
    - Base + Derived Classes
  - Plotting
- What python does not have:
  - Operator Overloading
  - Multiple constructors
  - Speed
  - Pointers
  - ???

Numpy

- Has made numerical computing much easier in recent years.
- Numpy matrices / arrays
- Numpy.linalg
- Behind many of these functions are LAPACK + BLAS!

Scipy

- Special functions (scipy.special)
- Integration (scipy.integrate)
- Optimization (scipy.optimize)
- Interpolation (scipy.interpolate)
- Fourier Transforms (scipy.fftpack)
- Signal Processing (scipy.signal)
- Linear Algebra (scipy.linalg)
- Sparse Eigenvalue Problems with ARPACK
- Compressed Sparse Graph Routines scipy.sparse.csgraph
- Spatial data structures and algorithms (scipy.spatial)
- Statistics (scipy.stats)
- Multidimensional image processing (scipy.ndimage)

nag4py

- Built on top of NAG C Library + Documentation
- 1600 NAG functions easily accessible from python
- 25 examples programs to help users call NAG functions

```
from nag4py.c05 import c05ayc

from nag4py.util import NagError,Nag_Comm
```

Pandas

- Data Analysis Package
- Many nice built in functions
- Common tools:
  - Series / DataFrame
  - Reading + Writing CSVs
  - Indexing, missing data, reshaping
  - Common time series functionality

(Examples)

Implied Volatility

- Black Scholes Formula for pricing a call/put option is a function of 6 variables:
  - $C(S_0, K, T, \sigma, r, d) = S_0 N(d_1) - K e^{-rT} N(d_2)$
- Where
  - $d_{1,2} = \frac{1}{\sigma\sqrt{T}}\left[ln\left(\frac{S}{K}\right) + T\left(r \pm \frac{\sigma^2}{2}\right)\right]$
  - $N(x) =$ Standard Normal CDF

Implied Volatility

- We can observe the following in the market:
- $C(S_0, K, T, \sigma, r, d) = C$
- But what is σ?
- $\sigma_{imp} \rightarrow C_{BS}(S_0, K, T, \sigma_{imp}, r, d) = Market\ Price$

Implied Volatility

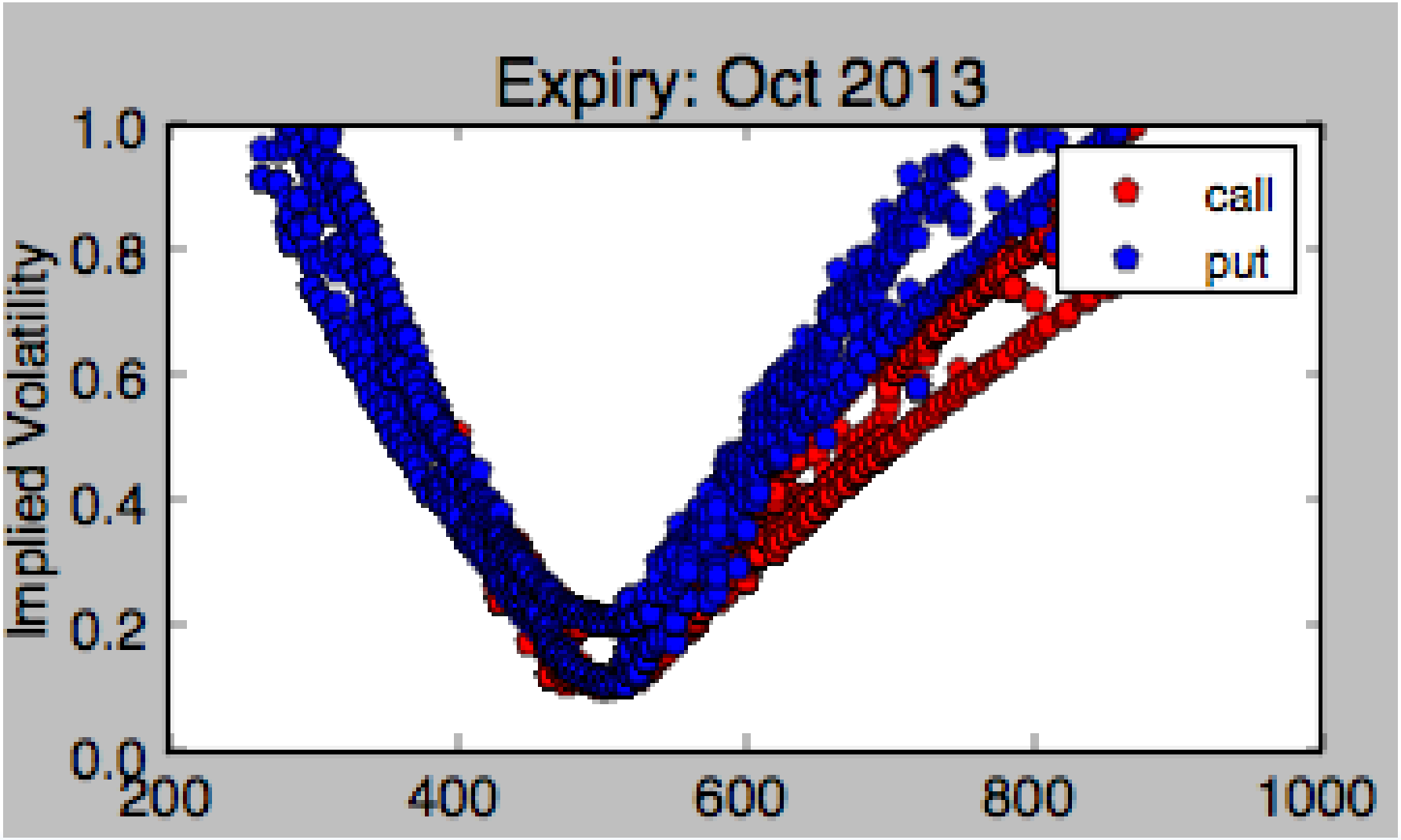
- We can observe the following in the market:
- $C(S_0, K, T, \sigma, r, d) = C$
- But what is σ?
- $\sigma_{imp} \rightarrow C_{BS}(S_0, K, T, \sigma_{imp}, r, d) = Market\ Price$
- Does σ_{imp} exist?

Implied Volatility

- We can observe the following in the market:
- $C(S_0, K, T, \sigma, r, d) = C$
- But what is σ?
- $\sigma_{imp} \rightarrow C_{BS}(S_0, K, T, \sigma_{imp}, r, d) = Market\ Price$
- Does σ_{imp} exist?
  - Yes

(Examples)

Implied Volatility – Different Curves?



Implied Volatility – Different Curves?

- **No hyphen or letter present** = Composite
  A = AMEX American Stock Exchange
  B = BOX Boston Stock Exchange - Options
  E = CBOE Chicago Board Options Exchange
  I = BATS
  J = NASDAQ OMX BX
  O = NASDAQ OMX
  P = NYSE Arca
  X = PHLX Philadelphia Stock Exchange
  Y = C2 Exchange
  4 = Miami Options Exchange
  8 = ISE International Securities Exchange

Implied Volatility

- Reasons for skews/smiles?
  - Risk Preferences
  - Fat tailed distributions

Implied Volatility Timings

Method	Timing
fsolve + python BSM	
fsolve + NAG BSM	
nag4py	
NAG C	

Implied Volatility Timings

Method	Timing
fsolve + python BSM	**~180 seconds**
fsolve + NAG BSM	
nag4py	
NAG C	

Implied Volatility Timings

Method	Timing
fsolve + python BSM	~180 seconds
fsolve + NAG BSM	**~15 seconds**
nag4py	
NAG C	

Implied Volatility Timings

Method	Timing
fsolve + python BSM	~180 seconds
fsolve + NAG BSM	~15 seconds
nag4py	**~10 seconds**
NAG C	

Implied Volatility Timings

Method	Timing
fsolve + python BSM	~180 seconds
fsolve + NAG BSM	~15 seconds
nag4py	~10 seconds
NAG C	**~.29 seconds**

Implied Volatility Timings

Method	Timing
fsolve + python BSM	~180 seconds
fsolve + NAG BSM	~15 seconds
nag4py	~10 seconds
NAG C	~.29 seconds

- **Derivatives?**
- **We have the derivative, vega**
  - $\frac{\partial C}{\partial \sigma} = S * T * N'(d_1)$

Fitting Data Points

- In our script we had k = l = 3…
  - What if we try different values?

Fitting Data Points

- In our script we had k = l = 3…
  - What if we try different values?
    - Poor results, can we do better?

Thank You

Questions?

- Further reading see:
- http://pandas.pydata.org/
- http://www.nag.co.uk/python.asp
- http://blog.nag.com/2013/10/implied-volatility-using-pythons-pandas.html