Indian Institute of Management Calcutta

Working Paper Series

WPS No. 777
March 2016

Volatility curve generation using the Heston Model

Krishna Praturi
PGDM Student, IIM Calcutta, D. H. Road, Joka P.O., Kolkata 700 104 India

Binay Bhushan Chakrabarti
Professor, Indian Institute of Management Calcutta
D. H. Road, Joka, P.O. Kolkata 700 104

http://facultylive.iimcal.ac.in/workingpapers

Volatility curve generation using the Heston Model

Working Paper

by

a) Corresponding author: B.B.Chakrabarti

Prof. (Retd.) of Finance, IIM Calcutta, bbc@iimcal.ac.in

b) Krishna Praturi

PGDM student, IIM Calcutta, email krishnap2016@email.iimcal.ac.in

I. Abstract

The objective of the paper is to generate a volatility curve for the USDINR currency pair based on the Heston model. For any tenor, I will be using the ATM implied volatility, the 25 Delta Risk Reversal implied volatility, and the 25 Delta Butterfly implied volatility as inputs. I intend to use MATLAB as the programming tool. The aim is to compare the generated volatility curve with the observed market volatility curve and consequently examine the efficacy of the Heston model. The paper will be confined to the USDINR currency market and the stochastic Heston model for the purposes of volatility curve generation.

This working paper is structured as follows. We first introduce the FX variant of the Black Scholes model and typical quoting conventions in the FX options market. We then illustrate the Heston model and the methodology of the calibration process. We then summarize the results obtained from the calibration and analyze the success of the Heston model in replicating the actual implied volatility curve observed in the USDINR market. MATLAB was used for simulating the pricing and calibration routines demonstrated in this paper; the relevant code is presented in the appendix.

II. Introduction

Options play a crucial role in the financial markets. The earliest model to provide a framework to price options was the Black Scholes model. It makes several assumptions, the most crucial one being that the underlying asset follows geometric Brownian motion with constant volatility. However, it has been empirically established over the past few decades that implied volatilities for different strikes are almost never constant. This disparity is called the volatility skew or smile. Generally, it is observed that at-the-money options have lower implied volatilities than in-the-money or out-of-the-money options.

To account for this discrepancy, option pricing models began to model volatility as a stochastic process.

for a better calibration of the volatility surface, the 10Delta Risk Reversal and the 10Delta Butterfly. For the purposes of completion, I now define some regularly used terms in this paper.

Volatility Curve: It is a plot of the implied volatility as a function of either the strike price or the delta. In the FX market and therefore for the purposes of this working paper, we plot the implied volatility as a function of the delta.

Risk Reversal: It is the difference between the implied volatility of the call price and the implied volatility of the put price at a specified moneyness level. For example, the 25 Delta Risk Reversal would be the difference between the implied volatility of the 25 Delta call and the implied volatility of the 25 Delta put.

$$25RR = \sigma_{25C} - \sigma_{25P}$$

Butterfly: It is the difference between the average volatility of a call and put option at a specified moneyness level and the implied volatility of an ATM option.

$$25BF = \frac{\sigma_{25C} + \sigma_{25P}}{2} - \sigma_{ATM}$$

Given the ATM volatility and the risk reversal, butterfly for a specific delta, the implied volatility for a call/put option at that delta can be calculated:

$$\sigma_{25C} = \sigma_{25BF} + \frac{\sigma_{25RR}}{2} + \sigma_{ATM}$$

$$\sigma_{25P} = \sigma_{25BF} - \frac{\sigma_{25RR}}{2} + \sigma_{ATM}$$

The typically quoted volatilities in the interbank market are the ATM volatility, 25D and 10D Risk Reversal and the Butterfly. Using these, it is possible to construct the implied volatility surface using a SV model.

The Garman Kohlhagen Model:

This is the variant of the Black-Scholes model used in pricing FX options given by the following equations.

$$C = S e^{-r_f T} N(d_1) - K e^{-r_d T} N(d_2)$$

$$P = K e^{-r_d T} N(-d_2) - S e^{-r_f T} N(-d_1)$$

$$d_1 = \frac{\ln\left(\frac{S}{K}\right) + \left(r_d - r_f + \frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}}$$

2Ý:Táôçâìá U; L s t E s è ± 4 ¶ 4 AÄ?Ü ì BÝ:Táôçâìáî ; E î @ î

BÝ:Táôçâìáî ; L ‡ š k%ç:ìáî ; E &Ý:ìáî ;ôç E E î T

%ç L kN F N oî E ìE G à ê6 J:>5 F é ê î E @ F t Ž ‘ ‰ s F CÝA×õç s F CÝ GK

&Ý:ìáî ; L >Ý F é ê î E @ ê6 F s F A×õç s F CÝA×õç G

CÝ L >Ý F é ê î E @ >Ý F é ê î E @

@ L § ké ê î E >Ýo6 F ê6 kt Q î EF î 6o

>5 L â E ã F ê é

>6 L â E ã

The following are the constraints on the parameters:

F s O é O s

â P r

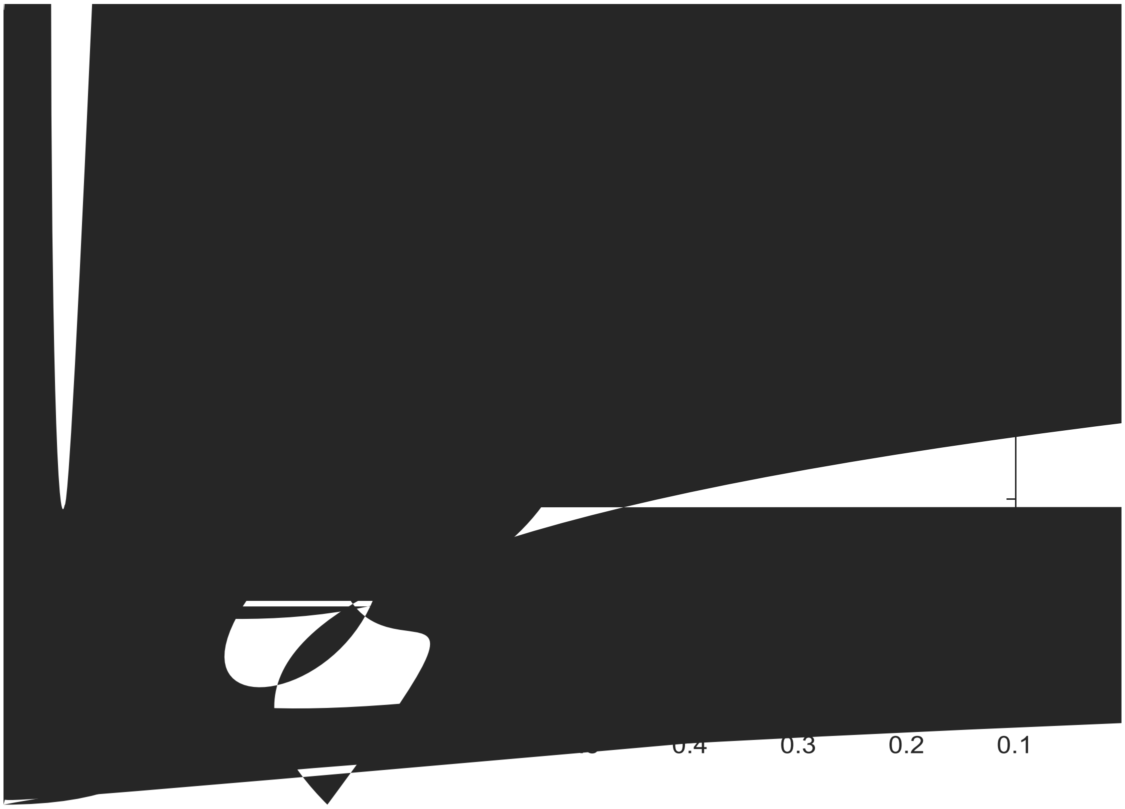
r O à O s

r O ê O s

t â à P ê6

We use MATLAB for the implementation of the Heston Model.

Model Volatility Curve

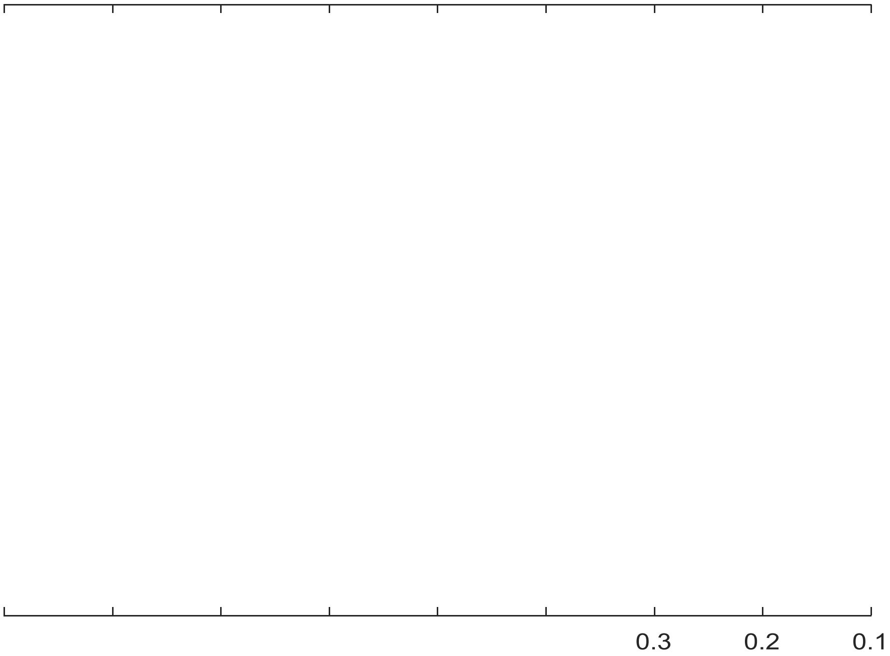


	Market Vols	Model Vols
10P	6.55	6.59
15P	6.44	6.47
20P	6.40	6.41
25P	6.38	6.38
30P	6.41	6.38
35P	6.45	6.41
40P	6.52	6.47
45P	6.59	6.56
ATM	6.69	6.67
45C	6.82	6.81
40C	6.98	6.98
35C	7.17	7.17
30C	7.40	7.40
25C	7.70	7.67
20C	7.99	7.99
15C	8.36	8.36
10C	8.82	8.81

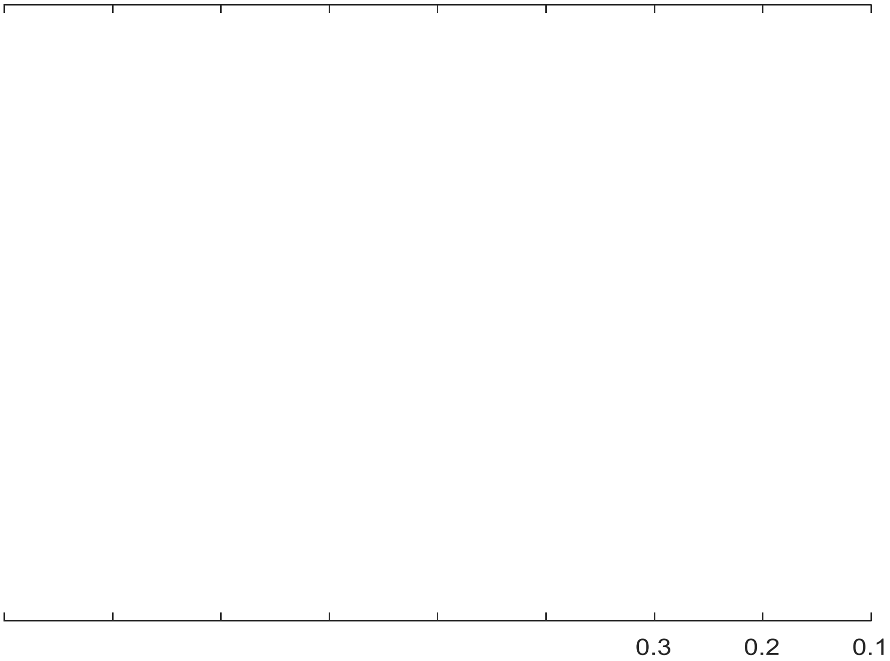
Mean Absolute % Error=0.279%

3M Maturity:

Observed Vol Curve:



Model Vol Curve:

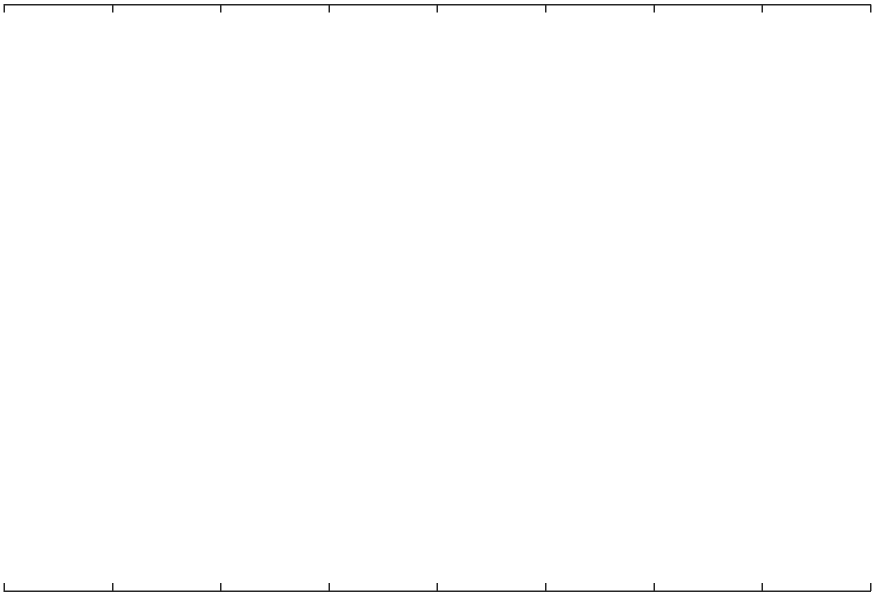


Model Vol Curve:

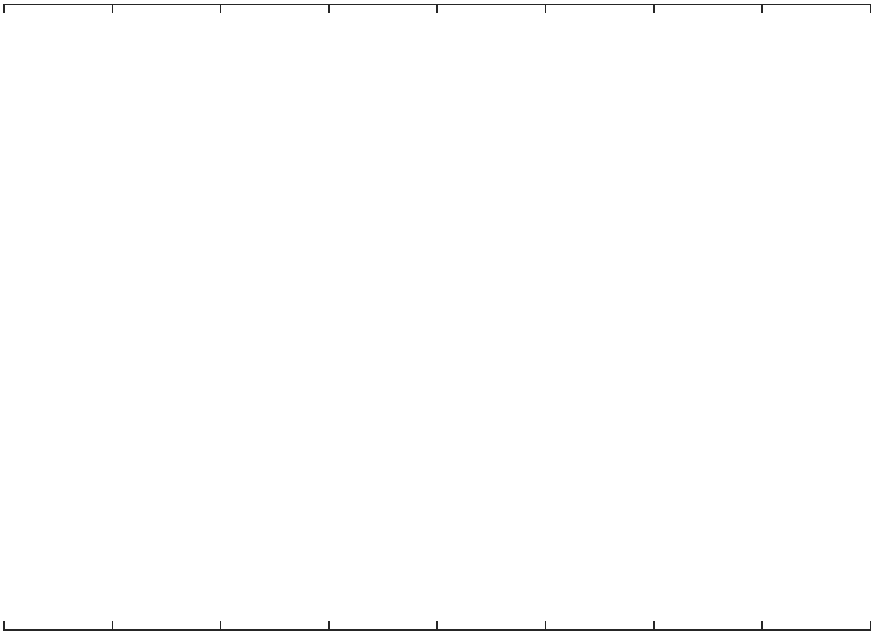
	Market Vols	Model Vols
0.9	7.06	6.99
0.85	6.99	6.95
0.8	6.97	6.97
0.75	7.00	7.04
0.7	7.15	7.14
0.65	7.30	

12M Maturity:

Observed Vol Curve:



Model Vol Curve:



	Market Vols	Model Vols
10P	7.68	7.70
15P	7.63	7.58
20P	7.63	7.59
25P	7.69	7.68
30P	7.88	7.82
35P	8.08	8.01
40P	8.30	8.24
45P	8.54	8.50
ATM	8.79	8.79
45C	9.06	9.12
40C	9.38	9.48
35C	9.77	9.89
30C	10.26	10.34
25C	10.85	10.85
20C	11.51	11.45
15C	12.29	12.17
10C	13.14	13.14

Mean Absolute Percentage Error=0.55%

VII. Summary

VIII. References

1) Janek Agnieszka, Kluge Tino, ŒtΩ}vZ (oU tÇ•šµ%oX FX smile in the Heston model. Statistical Tools for Finance and Insurance, 2011.

2) Crisostomo, Ricardo. "An Analysis of the Heston Stochastic Volatility Model: Implementation and Calibration using Matlab. Comisión Nacional del Mercado de Valores (CNMV) Working Paper, 2014.

3) Anders Persson. o] Œ š]}v }(&y K‰š]}v• v ‰Œ] •]v P Œ}[• dZ }Œ Œ] • Œš }‰š]}v•

University, 2013

4) Yu Tian. X ^dZ ,Ç Œ] ->š} oZ s•šq š]o]šÇ D} o Á]šZ ‰‰o] š]}v•]v WŒ]]

Doctoral Thesis at Monash University, Australia, 2013

ñ• zµ v zVal.Ping X a´Europe v }‰š]}v Á]šZ šZ , D š•}šv ŒE}[• od_Rochester š

Institute of Technology, 2013

IX. Appendix This is the MATLAB code used for implementing the Heston Model for the USDINR market

```
function buildsmile(atm,rr25,bf25,rr10,bf10,s,rd,rf,t)


    c10=atm+bf10+rr10/2;
    p10=atm+bf10-rr10/2;
    c25=atm+bf25+rr25/2;
    p25=atm+bf25-rr25/2;
atmk= s.*exp(-1.*norminv(0.5.*exp(rf*t),0,1).*atm.*sqrt(t)+(rd-
rf+0.5.*atm.^2).*t);
test= s.*exp((rd-rf+0.5.*atm.^2).*t);
    ck25= s.*exp(-1.*norminv(0.25.*exp(rf*t),0,1).*c25.*sqrt(t)+(rd-
rf+0.5.*c25.^2).*t);
    ck10= s.*exp(-1.*norminv(0.1.*exp(rf*t),0,1).*c10.*sqrt(t)+(rd-
rf+0.5.*c10.^2).*t);
    pk25= s.*exp(1.*norminv(0.25.*exp(rf*t),0,1).*p25.*sqrt(t)+(rd-
rf+0.5.*p25.^2).*t);
    pk10= s.*exp(1.*norminv(0.1.*exp(rf*t),0,1).*p10.*sqrt(t)+(rd-
rf+0.5.*p10.^2).*t);
atmgk=fxoptioncall(s,atmk,rd,rf,t,atm);
    cgk25=fxoptioncall(s,ck25,rd,rf,t,c25);
    cgk10=fxoptioncall(s,ck10,rd,rf,t,c10);
    pgk25=fxoptionput(s,pk25,rd,rf,t,p25);
    pgk10=fxoptionput(s,pk10,rd,rf,t,p10);
    heston0=[0.1 0.1 0.1 0.1 0.1];
ub=[0.9999 0.9999 0.9999 0.9999 0.9999];
lb=[0.0001 -0.9999 0.0001 0.0001 0.001];
nonlcon_heston=@(heston) nonlcon(heston);
opts = optimoptions('fmincon','Algorithm','sqp');

sse(heston0,s,rd,rf,t,atmk,ck25,ck10,pk25,pk10,atmgk,cgk25,cgk10,pgk25,pgk10)
problem = createOptimProblem('fmincon','objective', ...
    @(heston)
sse(heston,s,rd,rf,t,atmk,ck25,ck10,pk25,pk10,atmgk,cgk25,cgk10,pgk25,pgk10),
'x0',heston0,'lb',lb,'ub',ub, ...
'options',opts);
gs= GlobalSearch;
    [hest ans1]= run(gs,problem);
    ans1;
    p=1;vol=0;
%calculate smile for calls
for i=50:0.1:80
```

```
function h=hestonfxcall(s,v,k,rf,rd,t,rho,sigma,kappa,theta)
```