18th Winter school on Mathematical Finance

Lunteren, January 21-23, 2019

Rough volatility

Lecture 3: Forecasting

Jim Gatheral
Department of Mathematics

Outline of Lecture 3

- Forecasting realized variance

- Predicting the forward variance curve

- The fair value of VIX futures

Rough volatility and forecasting

- What is are rough volatility models good for?
    - One obvious application is to volatility forecasting.

- If follows that we can estimate the fair value of the variance swap curve.
    - With high frequency data as input, rough volatility models generate option fair value estimates.

Forcasting fBm

- In the RFSV model (1), $\log \sigma_t \approx \nu\, W_t^H + C$ for some constant C.

- [Nuzman and Poor][6]</sup> show that $W_{t+\Delta}^H$ is conditionally Gaussian with conditional expectation

$$\mathbb{E}[W_{t+\Delta}^H | \mathcal{F}_t] = \frac{\cos(H\pi)}{\pi} \Delta^{H+1/2} \int_{-\infty}^t \frac{W_s^H}{(t - s + \Delta)(t - s)^{H+1/2}} ds$$

and conditional variance

$$\mathrm{Var}[W_{t+\Delta}^H | \mathcal{F}_t] = \tilde{c}\, \Delta^{2H}.$$

where

$$\tilde{c} = \frac{\Gamma(3/2 - H)}{\Gamma(H + 1/2)\, \Gamma(2 - 2H)}.$$

A heuristic explanation of the formula

- The forecast formula comes from regressing $W_{t+\Delta}^H$ against the W_s^H with $s

- Let

$$\beta(u, \Delta) = \frac{\cos(H\pi)}{\pi} \Delta^{H+1/2} \frac{1}{(u + \Delta)\, u^{H+1/2}}.$$

Then, for $t, \Delta > 0$ and $0

- In particular,

$$\int_0^\infty \beta(u, \Delta)\, du = 1.$$

- With $\beta(u, \Delta)$ thus defined and for $s

$$\mathbb{E}\left[W_s^H \left(W_{t+\Delta}^H - \int_{-\infty}^t \beta(t - u, \Delta)\, W_u^H\, du \right) \right] = 0.$$

  - In other words, the $\beta(t - u, \Delta)$ are the normal regression coefficients.

The forecast formula

Using that W^H is a Gaussian random variable, we get that

Variance forecast formula

(3)

$$\mathbb{E}^{\mathbb{P}}\left[v_{t+\Delta}\,|\,\mathcal{F}_t\right] = \exp\left\{\mathbb{E}^{\mathbb{P}}\left[\log(v_{t+\Delta})\,|\,\mathcal{F}_t\right] + 2\,\tilde{c}\,\nu^2\Delta^{2H}\right\}$$

---

where

$$\mathbb{E}^{\mathbb{P}}\left[\log v_{t+\Delta}\,|\,\mathcal{F}_t\right]$$
$$= \frac{\cos(H\pi)}{\pi}\Delta^{H+1/2}\int_{-\infty}^{t}\frac{\log v_s}{(t-s+\Delta)(t-s)^{H+1/2}}ds.$$

Discretization of the forecast formula

In [Gatheral, Jaisson, Rosenbaum][4]</sup>, we discretize the integral by taking mid-points as in

$$\mathbb{E}^{\mathbb{P}}\left[\log v_{t+\Delta}\,|\,\mathcal{F}_t\right] \approx \frac{1}{A}\sum_{j=0}^{L}\frac{\log v_{t-j}}{\left(j+\frac{1}{2}+\Delta\right)\left(j+\frac{1}{2}\right)^{H+1/2}}.$$

where L is the maximum number of lags and the normalizing constant A is given by

$$A = \sum_{j=0}^{\infty}\frac{1}{\left(j+\frac{1}{2}+\Delta\right)\left(j+\frac{1}{2}\right)^{H+1/2}}.$$

Inspired by [Bennedsen, Lunde and Pakkanen][2]</sup>, we approximate the first term in the sum more accurately as follows.

$$\mathbb{E}^{\mathbb{P}}\left[\log v_{t+\Delta}\,|\,\mathcal{F}_t\right] \approx \frac{1}{A}\left\{\frac{\log v_t}{(s^{\star}+\Delta)(s^{\star})^{H+1/2}} + \sum_{j=1}^{L}\frac{\log v_{t-j}}{\left(j+\frac{1}{2}+\Delta\right)\left(j+\frac{1}{2}\right)^{H+1/2}}\right\}$$

where $s^{\star}$ is chosen such that

$$\frac{1}{\gamma} = \int_0^1\frac{ds}{s^{H+\frac{1}{2}}} = \frac{1}{s^{\star H+\frac{1}{2}}} = \frac{1}{s^{\star 1-\gamma}}$$

where $\gamma = \frac{1}{2} - H$. Thus

$$s^{\star} = \gamma^{\frac{1}{1-\gamma}}.$$

Load some useful libraries

```
In [1]:  library(quantmod)
         library(repr)
         library(stinepack)
```

```
Loading required package: xts
Loading required package: zoo

Attaching package: 'zoo'

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric

Loading required package: TTR
Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
In [2]:  options(repr.plot.width=10,repr.plot.height=7)
```

Load realized kernel estimates from Lecture 1

```
In [3]:  setwd("./LRV")

         load("OxfordRV.rData")
```

```
In [4]:  h.spx.regression <- 0.15
         nu.spx.regression <- 0.30
```

```
In [5]:  tail(spx.rk)
```

```
                      [,1]
2019-01-03 2.168852e-04
2019-01-04 1.305852e-04
2019-01-07 8.449293e-05
2019-01-08 8.199344e-05
2019-01-09 7.417078e-05
2019-01-10 6.374206e-05
```

Implement variance forecast in R

```
In [6]:  # Find all of the dates
         dateIndex <- substr(as.character(index(spx.rk)),1,10) # Create index of
          dates

         cTilde <- function(h){gamma(3/2-h)/(gamma(h+1/2)*gamma(2-2*h))} # Factor
          because we are computing conditional on \cF_t

         # XTS compatible version of forecast
         rv.forecast.XTS <- function(rvdata,h,date,nLags,delta,nu){
             gam <- 1/2-h
             j <- (1:nLags)-1
             cf <- 1/((j+1/2)^(h+1/2)*(j+1/2+delta)) # Lowest number should apply
          to latest date
             s.star <- gam^(1/(1-gam))
             cf[1] <- 1/(s.star^(h+1/2)*(s.star+delta))
             datepos <- which(dateIndex==date)
             ldata <- log(as.numeric(rvdata[datepos-j])) # Note that this object
          is ordered from earlier to later
             pick <- which(!is.na(ldata))
             norm <- sum(cf[pick])
             fcst <- cf[pick]%*%ldata[rev(pick)]/norm # Most recent dates get the
          highest weight
             return(exp(fcst+2*nu^2*cTilde(h)*delta^(2*h)))

             }
```

SPX actual vs forecast variance

- In order to forecast using (3), we need estimates of H and ν.
    - We use our estimates of H and ν from the regressions rather than from the ACF estimator.
    - The choice does not seem to make much difference.

```
In [7]:  var.forecast.spx <- function(h,nu)function(del){
             n <- length(spx.rk)
             nLags <- 200

             range <- nLags:(n-del)
             rv.predict <- sapply(dateIndex[range],function(d){rv.forecast.XTS(rv
         data=spx.rk,h,d,nLags=nLags,delta=del,nu)})
             rv.actual <- spx.rk[range+del]
             return(list(rv.predict=rv.predict,rv.actual=rv.actual))
             }
```

- From experiment, we found that around 200 lags works best.

Scatter plot of delta days ahead predictions

```
In [8]: del <- 1
        vf <- var.forecast.spx(h=h.spx.regression,nu=nu.spx.regression)(del)
        rv.predict <- vf$rv.predict
        rv.actual <- vf$rv.actual
        vol.predict <- sqrt(as.numeric(rv.predict))
        vol.actual <- sqrt(as.numeric(rv.actual))
        vol.actual <-  sqrt(as.numeric(rv.actual))
```

```
In [9]: c(mean(vol.actual-vol.predict),sd(vol.actual-vol.predict))
```

-0.000342823993285706   0.00270576631284872

```
In [10]: plot(vol.predict,vol.actual,col="blue",pch=20,cex=.7, ylab="Actual vol."
         , xlab="Predicted vol.")
         abline(coef=c(0,1),col="red")
```
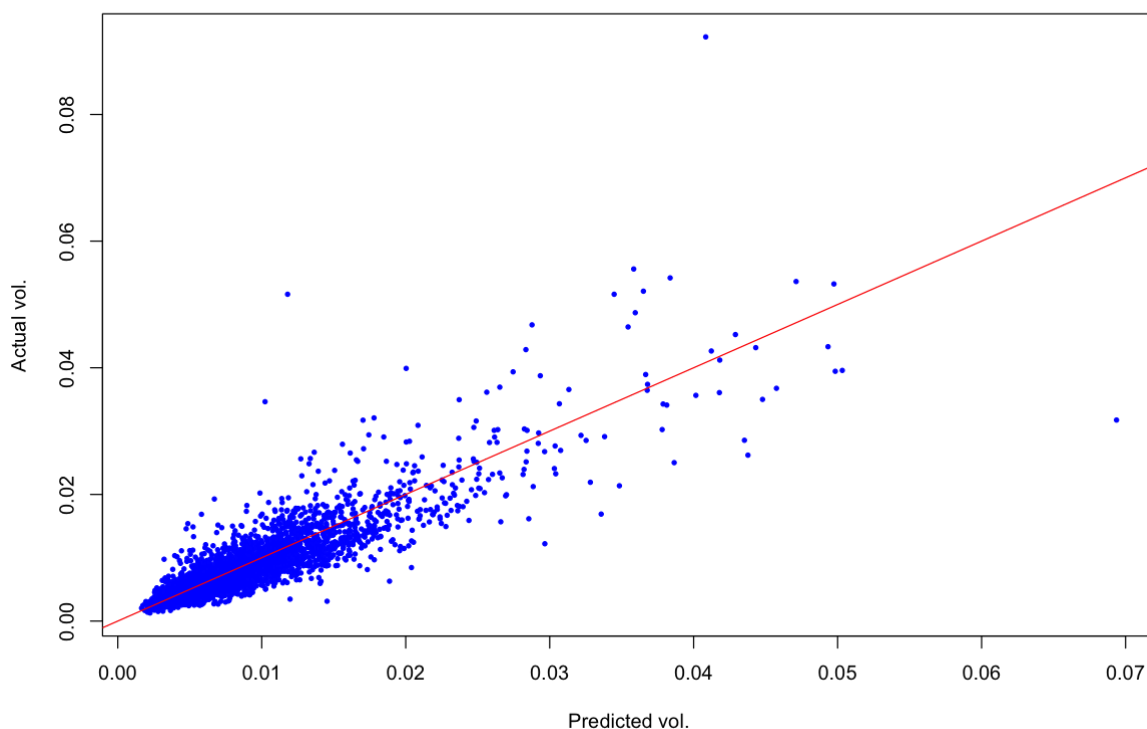

Figure 1: Actual vols vs predicted vols.

Which point is the outlier?

```
In [11]: rv.actual[which(as.numeric(vol.actual)>.09)]
```

```
                       [,1]
2008-10-10 0.008509655
```

```
In [12]:  rv.predict[which(as.numeric(vol.predict)>.06)]
```

2008-10-10: 0.00481234404469367

Superimpose actual and predicted vols

```
In [13]:  plot(vol.actual, col="blue",type="l")
          lines(vol.predict, col="red",type="l")
```
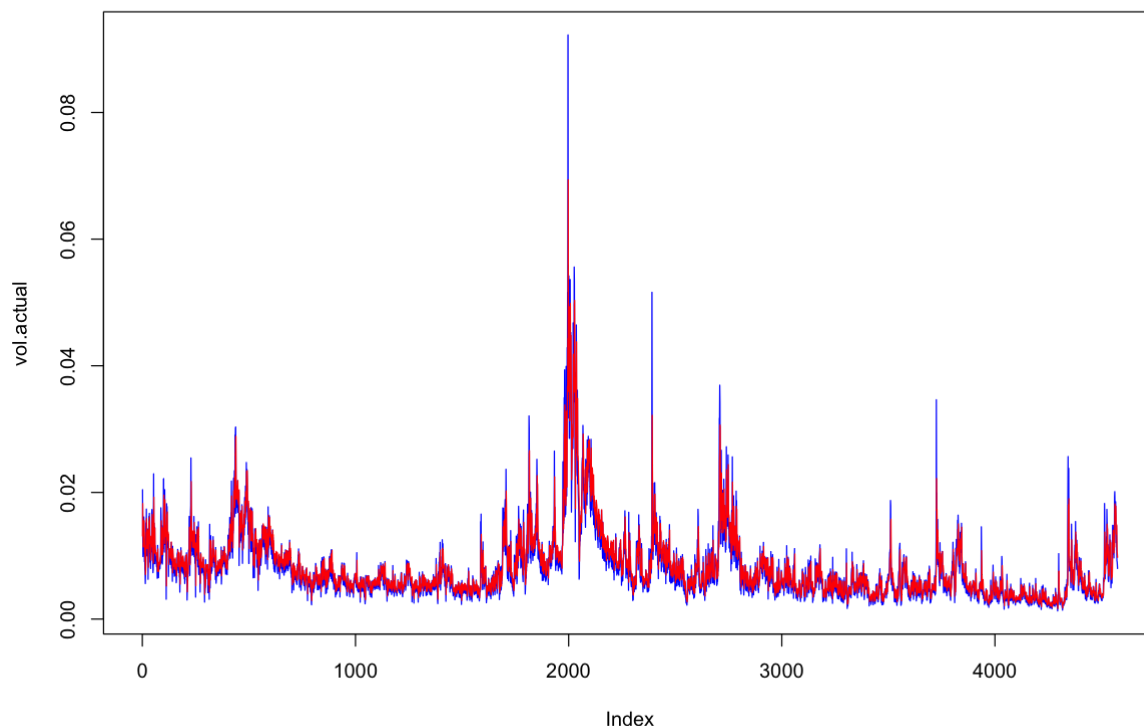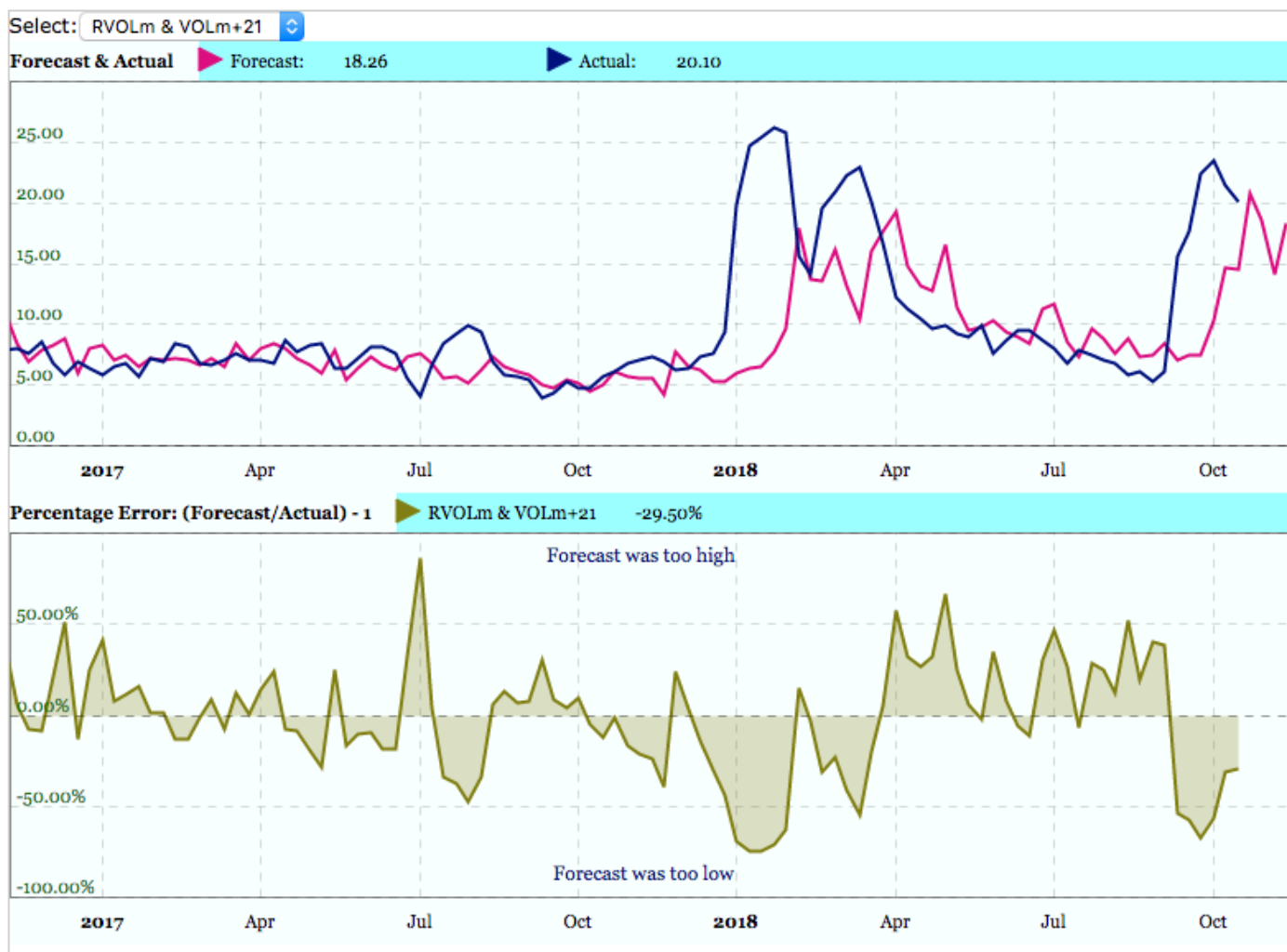

Figure 2: Actual volatilities in blue; predicted vols in red. Note that volatilities are in daily terms.

VolX

- The commercial company VolX (http://volx.us (http://volx.us)) has developed a number of RealVol Instruments and RealVol Indices based on realized volatility as defined by the RealVol Formulas.
    - Their business model is to license these indices to exchanges and information providers.

- They publish daily forecasts of RV using HARK (which is HAR-RV with Kalman filtering, and RVOL, an implementation of the Rough Volatility forecast.

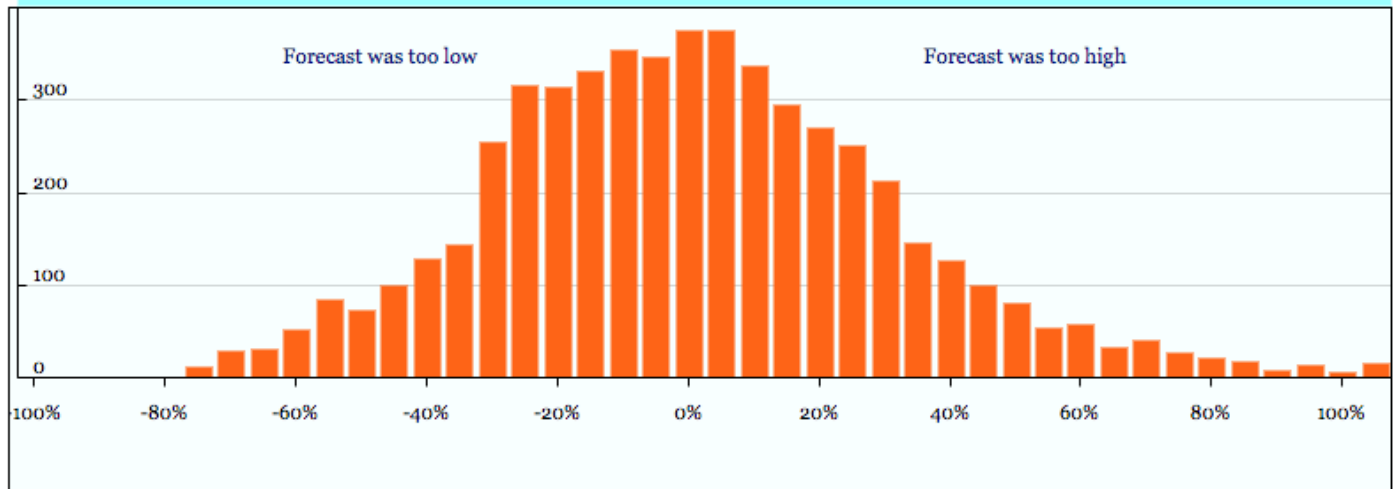- You can compare forecast versus actual volatility for the two estimators here: http://www.volx.us/volatilitycharts.shtml?2&SPY&PRED (http://www.volx.us/volatilitycharts.shtml?2&SPY&PRED).

VolX screenshots
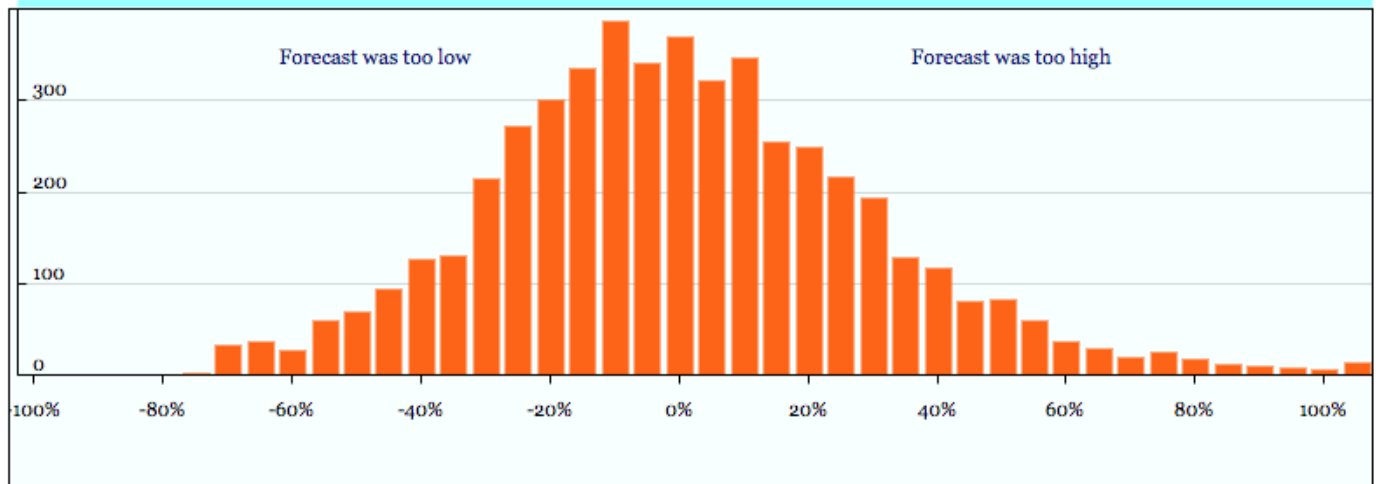
Conditional and unconditional variances

- The HAR and rough volatility forecasts are both impressive.
  - Much superior to alternatives such as GARCH.

- However, HAR is a regression and rough volatility is a proper model.

- One practical consequence is that we can put error bars on our volatility forecasts.

So how good is the forecast?

Specifically, by how much is the variance of the future variance reduced by taking into account the whole history of the fBm?

- In practice of course, we only consider some finite history, 200 points say.
- We know this again from [Nuzman and Poor][10]</sup> who showed that the ratio of the conditional to the unconditional variance of the $\log v_t$ is

$$\tilde{c} = \frac{\Gamma(3/2 - H)}{\Gamma(H + 1/2)\,\Gamma(2 - 2H)}.$$

- We can compute this ratio empirically and compare with the model prediction.

Unconditional and conditional variance vs lag Δ

First we compute the time series of prediction errors.

```
In [14]:  log.vol.err <- function(del){
              vf <- var.forecast.spx(h=h.spx.regression,nu=nu.spx.regression)(del)
              rv.predict <- vf$rv.predict
              rv.actual <- vf$rv.actual
              vol.predict <- sqrt(as.numeric(rv.predict))
              vol.actual <-  sqrt(as.numeric(rv.actual))
              err <- log(vol.actual)-log(vol.predict)
              return(err)
              }
```

```
In [15]:  var.log.err <- function(del){
              var(log.vol.err(del))
              }
```

```
In [16]:  var.log.err(10)
```

0.132808422717691

The following code takes too long to run. You can run it by uncommenting the code.

```
In [17]:  del <- 1:100
          #var.log.err.del <- sapply(del,var.log.err)
```

```
In [18]:  #save(var.log.err.del ,file="./LRV/varerr201811.rData")
          load(file="varerr201811.rData")
```

Plot of conditional and unconditional variance

- The unconditional variance of differences in log-vol is given by

$$m(2, \Delta) = \langle (\log \sigma_{t+\Delta} - \log \sigma_t)^2 \rangle.$$

- The conditional variance is given by $\verb|var.log.err|(\Delta)$.

```
In [19]:  sig <- sqrt(as.numeric(spx.rk))

          mq.del.Raw <- function(q,lag){mean(abs(diff(log(sig),lag=lag))^q)}
          mq.del <- function(x,q){sapply(x,function(x){mq.del.Raw(q,x)})}
```

```
In [20]:  plot(del,mq.del(del,2),pch=20,cex=1,ylab=expression(Variance),
               xlab=expression(Delta),col="blue",ylim=c(0,.35),
               main= "Unconditional and conditional variance")
          curve(nu.spx.regression^2*x^(2*h.spx.regression),from=0,to=100,add=T,col
          ="red",lwd=2,n=1000)
          points(del,var.log.err.del,col="green4",pch=20)
          curve(cTilde(h.spx.regression)* nu.spx.regression^2*x^(2*h.spx.regressio
          n),from=0,to=100,
               add=T,col="orange",lwd=2,n=1000)
```

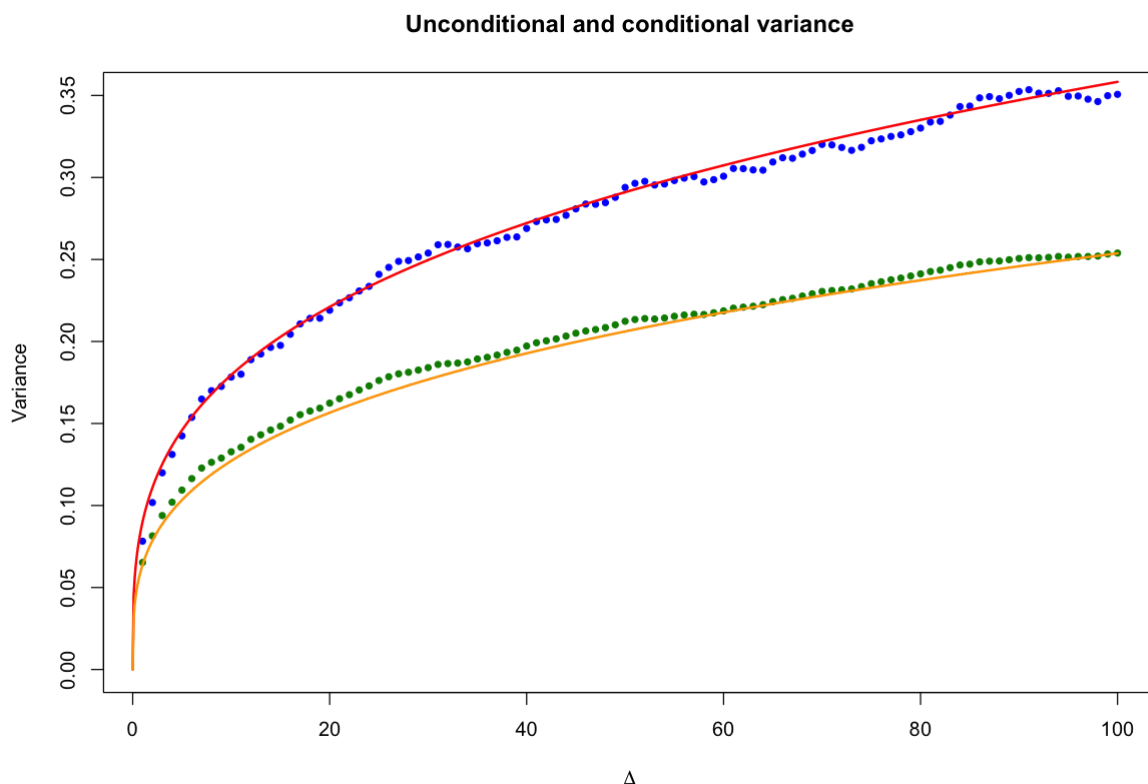**Unconditional and conditional variance**

Figure 3: Actual unconditional variance in blue, rough volatility prediction in red; Actual conditional variance in green, rough volatility prediction in orange.

Amazing agreement between data and model

- We observe that the ratio of conditional to unconditional variance is more or less *exactly* as predicted by the model!

Forecasting the variance swap curve

Finally, we forecast the whole variance swap curve using the variance forecasting formula (3).

```
In [21]: xi.raw <- function(date,nu,h,delta){ # delta is in days
             xi <- rv.forecast.XTS(rvdata=spx.rk,h=h,date=date,nLags=200,delta=de
         lta,nu=nu)
             return(xi)
         }

         # Forward variance curve (again the array tt should be in units of year
         s)
         xi.forecast <- Vectorize(xi.raw, vectorize.args = "delta")
```

```
In [22]: xi.forecast(date="2010-05-10",nu=nu.spx.regression,h=h.spx.regression,de
         lta=c(9,10))
```

0.000273054381799061 0.000266195103982083

```
In [23]: varSwapBDays <- function(date,nDays,nu,h,onFactor){

           tvec <- 1:nDays
           xicurve <- xi.forecast(date,nu,h,tvec)
           int.xicurve <- cumsum(xicurve)
           varcurve <- cumsum(xicurve)/tvec * onFactor
           res <- data.frame(tvec,varcurve)
           names(res) <- c("BDays","varswap")
           return(res)
         }
```

Plot of variance swap forecasts by day

```
In [24]: plot(varSwapBDays("2016-01-04",3*252,nu=nu.spx.regression,h=h.spx.regres
         sion,onFactor=1.4),
             pch=20,col="blue",xlab="Trading days", ylab="Variance swap rate fore
         cast")
```
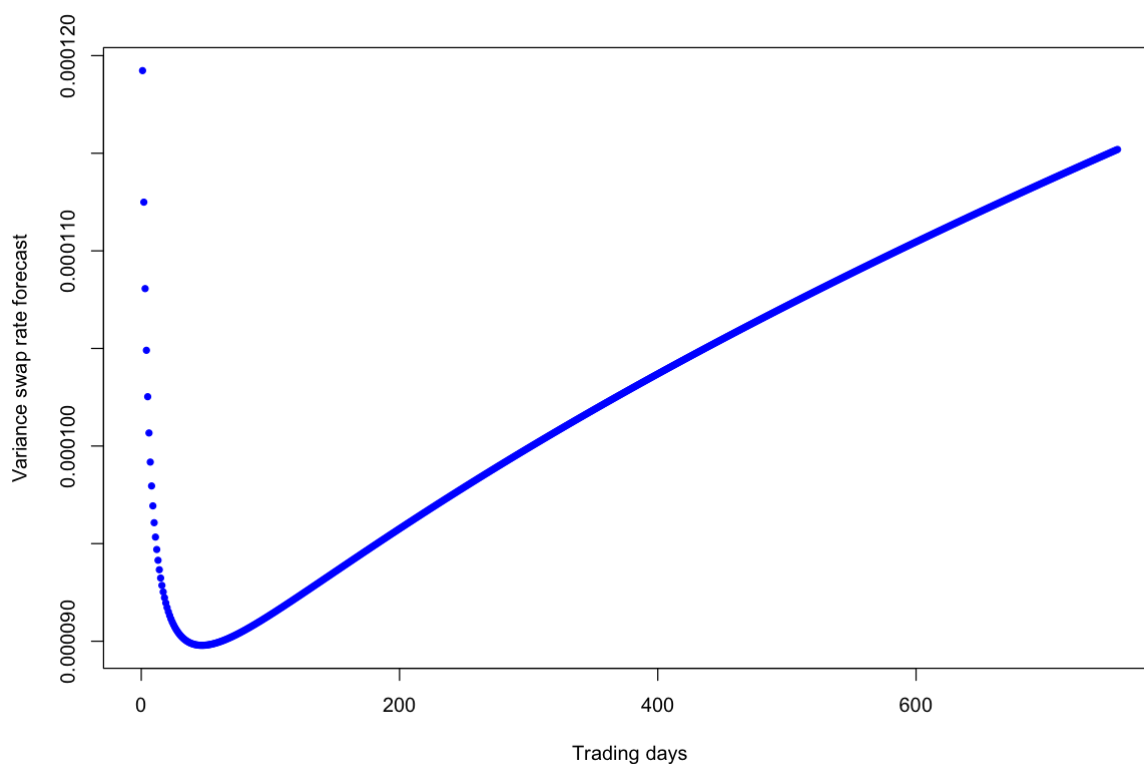

Figure 4: Forecast forward variance curve as of January 4, 2016.

A function to draw a continuous curve

```
In [25]: varSwapForecast <- function (date, tau, nu, h, onFactor)
         {
             vsc <- varSwapBDays(date, 252 * 3, nu = nu, h = h, onFactor)
             x <- vsc$BDays/252
             y <- sqrt(vsc$varswap * 252)
             res <- stinterp(x, y, tau)$y
             return(res)
         }
```

```
In [26]: curve(varSwapForecast("2016-01-04",x,nu=nu.spx.regression,h=h.spx.regres
         sion,onFactor=2.5),
              from=0,to=3,col="red",xlab="Maturity (years)", ylab="Forecast varian
         ce swap quote",n=1000)
```
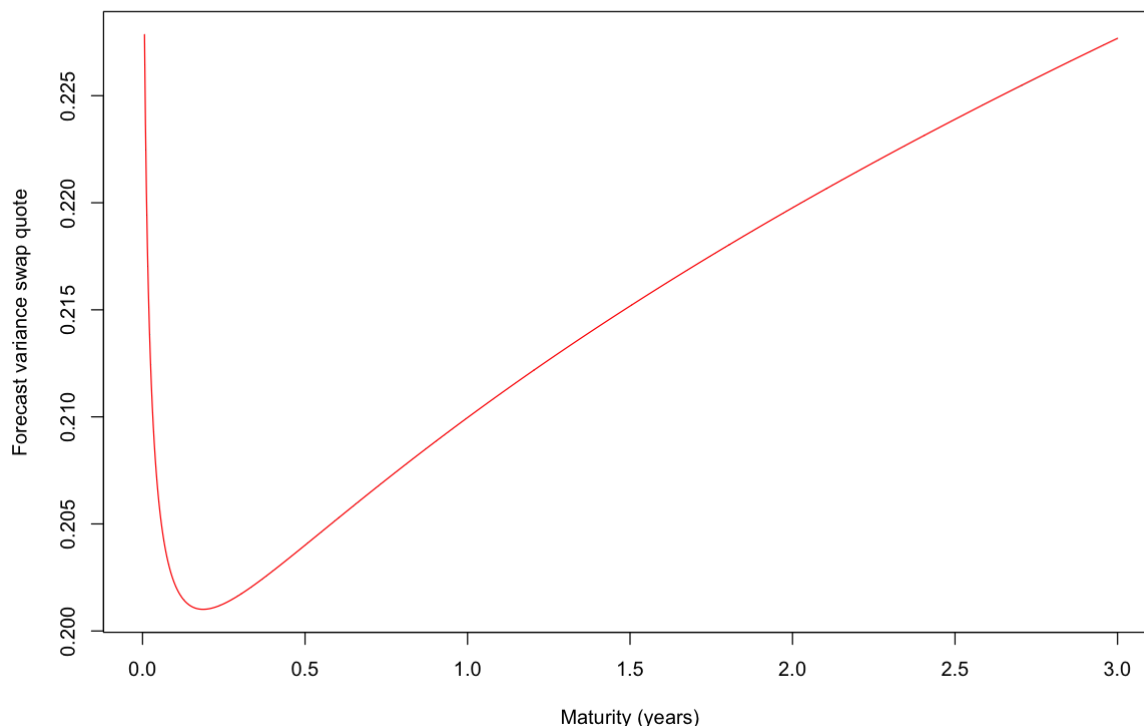

Figure 5: Forecast of annualized variance swap quotes as of January 4, 2016.

Constructing a time series of variance swap curves

For each of 2,658 days from Jan 27, 2003 to August 31, 2013:

- We computed proxy variance swaps from closing prices of SPX options sourced from OptionMetrics (www.optionmetrics.com) via WRDS.

- We form the forecasts $\mathbb{E}^{\mathbb{P}}\left[v_u \mid \mathcal{F}_t\right]$ using (3) with 200 lags of SPX RV data sourced from The Oxford-Man Institute of Quantitative Finance (http://realized.oxford-man.ox.ac.uk (http://realized.oxford-man.ox.ac.uk)).

- We note that the actual variance swap curve is a factor (of roughly 1.4) higher than the forecast, which we may attribute to a combination of overnight movements of the index and the price of volatility risk.

- Forecasts must therefore be rescaled to obtain close-to-close realized variance forecasts.
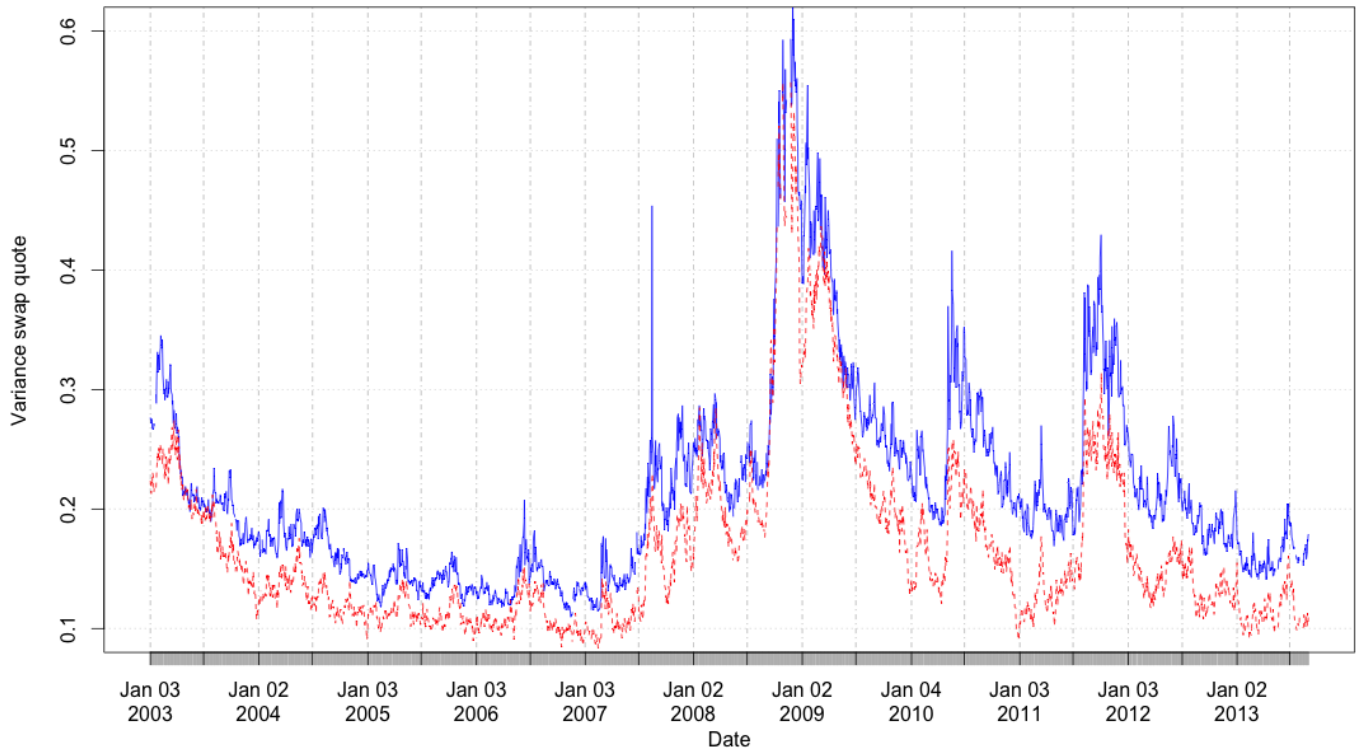
3-month forecast vs actual variance swaps

Figure 18: Actual (proxy) 3-month variance swap quotes in blue vs forecast in red (with no scaling factor).
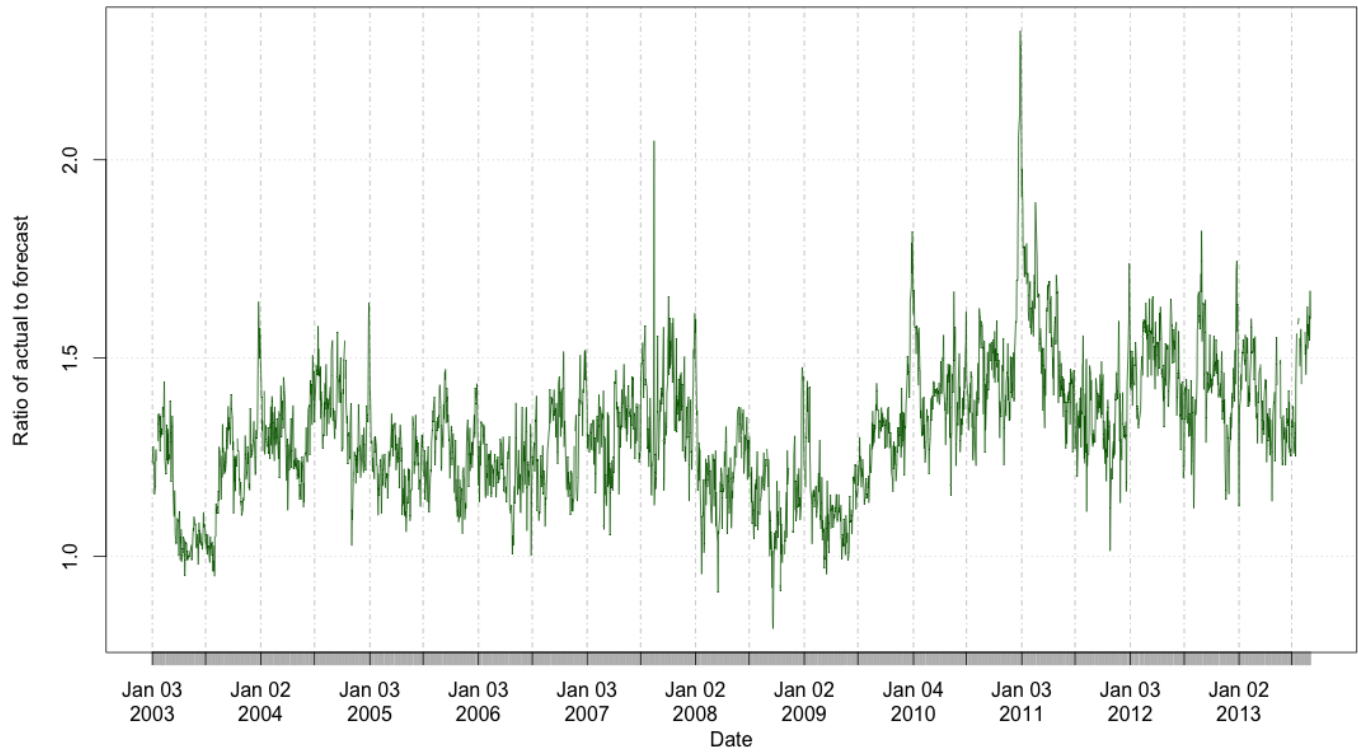
Ratio of actual to forecast

Figure 19: The ratio between 3-month actual variance swap quotes and 3-month forecasts.

The Lehman weekend

- Empirically, it seems that the variance curve is a simple scaling factor times the forecast, but that this scaling factor is time-varying.
  - We can think of this factor as having two multiplicative components: the overnight factor, and the price of volatility risk.

- Recall that as of the close on Friday September 12, 2008, it was widely believed that Lehman Brothers would be rescued over the weekend. By Monday morning, we knew that Lehman had failed.

- In Figure 19, we see that variance swap curves just before and just after the collapse of Lehman are just rescaled versions of the RFSV forecast curves.

- It appears that most of the evolution of the variance swap curve may be explained by a single extra data point - intraday realized variance from the open to the close of trading on Monday September 15, 2008.
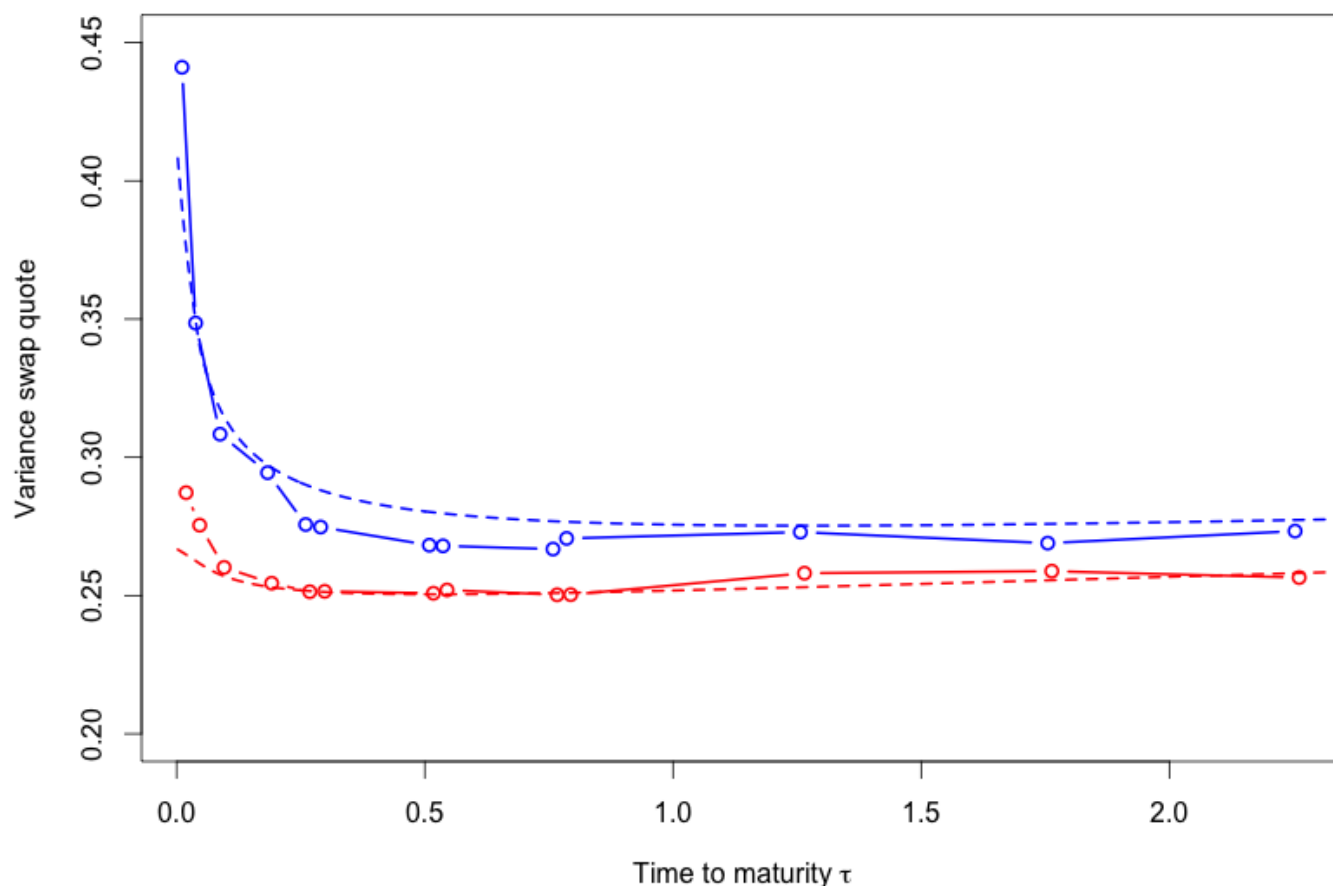
Figure 5: SPX variance swap curves as of September 12, 2008 (red) and September 15, 2008 (blue). The dashed curves are RFSV model forecasts rescaled by the 3-month ratio (1.29) as of the Friday close. Time measured in years.

Remarks

We note that

- The actual variance swaps curves are very close to the forecast curves, up to a scaling factor.

- We are able to explain the change in the variance swap curve with only one extra observation: daily variance over the trading day on Monday 15-Sep-2008.

- The SPX options market appears to be backward-looking in a very sophisticated way.

Compare forecast with actual variance swap curves

- We focus on January 4, 2016
  - A day with an interesting term structure of VIX futures from [Jacquier, Martini and Muguruza][5] </sup>.

```
In [27]:  source("plotIvols.R")
          source("fwdVarCurve.R")
          source("roughBergomiVixFutures.R")
```

ATM volatilities on January 4, 2016

```
In [28]:  ivolData <- read.csv("spxVols20160104.csv")
          res <- plotIvols(ivolData,plot=F)
          plot(res$expiries,res$atmVol,col="green4",pch=20,type="b",xlab="Expiry
            (years)", ylab="Implied volatility")
```
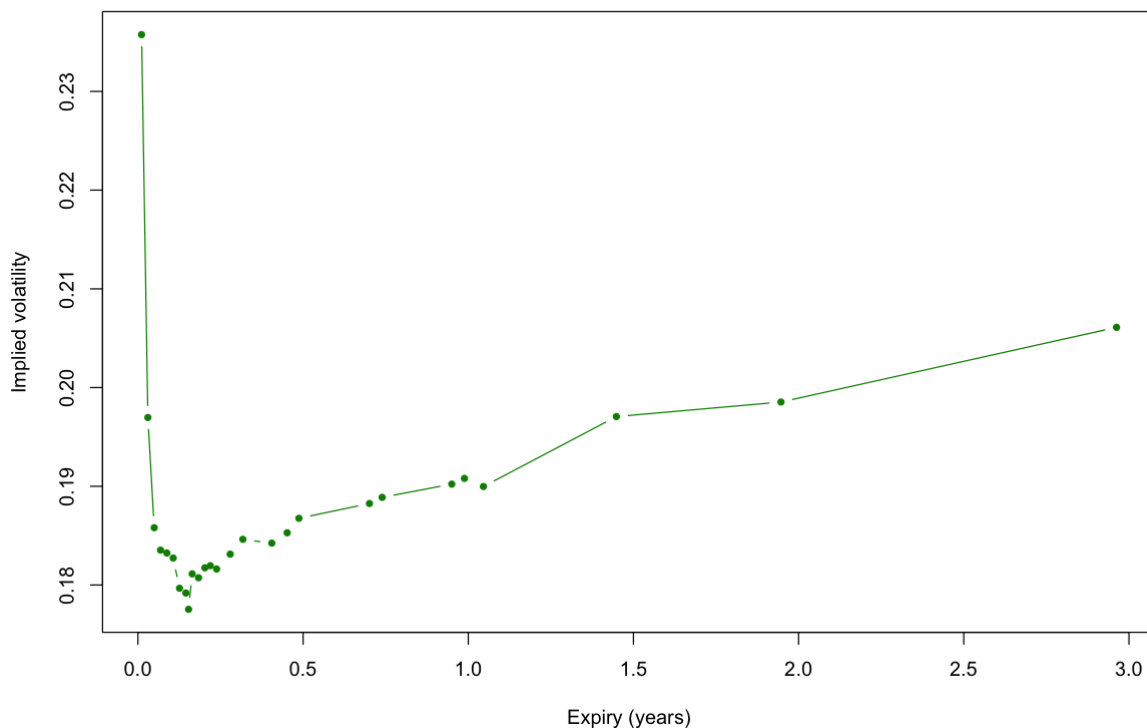

Figure 6: Term structure of SPX ATM volatility as of January 4, 2016

Variance swap estimates from eSSVI fits

- [Jacquier, Martini and Muguruza][5]</sup> explain in detail how to fit eSSVI to the volatility surface.

```
In [29]:  vsData.20160104 <- read.csv("varianceSwaps20160104.csv")
```

```
In [30]:  plot(vsData.20160104$texp,sqrt(vsData.20160104$vs),pch=20,col="blue",yla
          b="Variance rate",xlab="Expiry")
          curve(varSwapForecast("2016-01-04",x,nu=nu.spx.regression,h=h.spx.regres
          sion,onFactor=3.),
                from=0,to=3,col="red",xlab="Maturity (years)", ylab="Forecast varia
          nce swap quote",n=1000000,add=T)
```
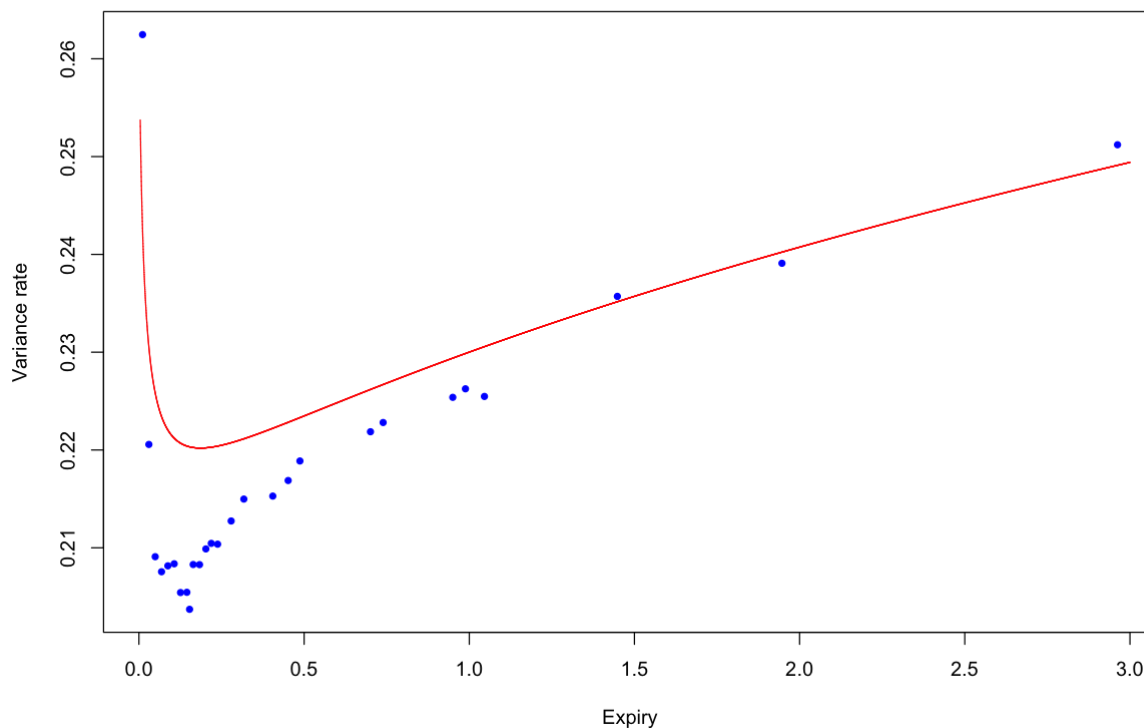

Figure 7: SPX variance swap rates estimates (in blue) as of Jan 4, 2016 with variance curve forecast superimposed in red.

Remarks

- The forecast doesn't look that great, although the characteristic shape is there.

- We can ask a different question: Is there any choice of the parameters H and ν and the overnight factor that would make the forecast close to the actual?

- In Figure 8, we see the answer is affirmative.
    - Moreover the parameters make sense.

Optimized forecast vs actual variance swap curve as of January 4, 2016

```
In [31]: plot(vsData.20160104$texp,sqrt(vsData.20160104$vs),pch=20,col="blue",yla
         b="Variance rate",xlab="Expiry")
         curve(varSwapForecast("2016-01-04",x,nu=.391,h=.135,onFactor=2.37),
             from=0,to=3,col="red",xlab="Maturity (years)", ylab="Forecast varian
         ce swap quote",n=1000000,add=T)
```
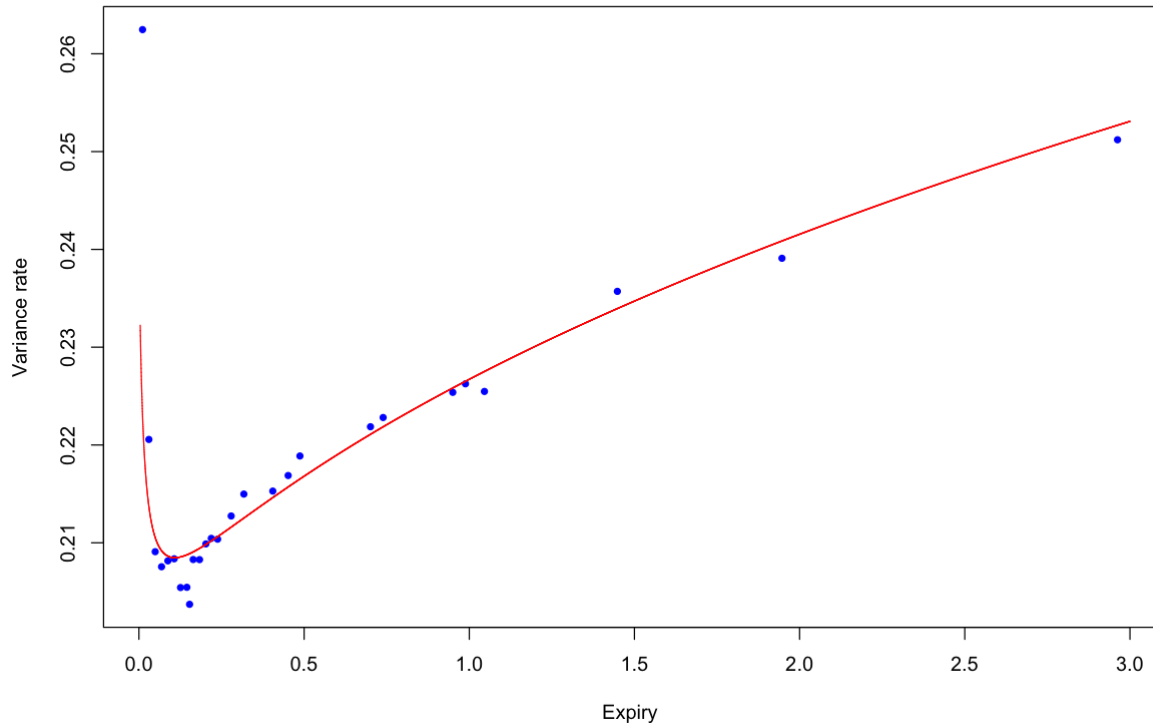

Figure 8: SPX variance swap rates estimates (in blue) as of Jan 4, 2016 with variance curve forecast superimposed in red.

Sensitivity of variance swaps to extrapolation

- In general, the value of a variance swap is very sensitive to how the smile is extrapolated.
    - Two curves that appear to fit traded strikes very well can be very different in the wings.

- The key point here is that rough volatility models are consistent with the shape of the forward variance cuve.

VIX futures

- If we can forecast the variance swap curve, we can also forecast the foreward variance curve.

- The VIX payoff is the square-root of the forward variance swap.
  - A convexity adjustment relates $\mathbb{E}[VIX]$ to $\sqrt{\mathbb{E}[VIX^2]}$.

- In principle, we can therefore forecast VIX futures

- [Jacquier, Martini and Muguruza][5]</sup> did the reverse:
  - Given the estimated forward variance curve and the observed VIX futures prices, what are H and ν?

The distribution of VIX future payoffs

- Denote the terminal value of the VIX futures by $\sqrt{\zeta(T)}$. Then, by definition (see Chapter 11 of [The Volatility Surface][3] for more details),

$$\zeta(T) = \frac{1}{\Delta} \int_T^{T+\Delta} \mathbb{E}[v_u|\mathcal{F}_T]\, du.$$

where Δ is one month.

- In the rough Bergomi model,

$$v_u = \xi_t(u)\, \mathcal{E}\left(\eta \sqrt{2H} \int_t^u \frac{dW_s}{(u-s)^\gamma}\right)$$

with $\gamma = 1/2 - H$ so v_u is lognormal.

The lognormal approximation under rough Bergomi

- Under rough Bergomi, the VIX payoff and its square $\zeta(T)$ should be approximately lognormally distributed.
  - The quality of this approximation was confirmed by [Jacquier, Martini and Muguruza][5]</sup>.
  - In that case, the terminal distribution of $\zeta(T)$ is completely determined by $\mathbb{E}\left[\zeta(T)|\mathcal{F}_t\right]$ and $\mathrm{var}[\log \zeta(T)|\mathcal{F}_t]$.

- Obviously

$$\mathbb{E}\left[\zeta(T)|\,\mathcal{F}_t\right] = \frac{1}{\Delta}\int_T^{T+\Delta}\xi_t(u)\,du.$$

  - Recall that forward variances $\xi_t(u)$ may be estimated from variance swaps which can themselves be proxied by the log-strip (see Chapter 11 of [The Volatility Surface][8]</sup> again).
  - Alternatively they may be estimated from linear strips of VIX options.

Approximating the conditional variance of $\zeta(T)$ under rough Bergomi

- To estimate the conditional variance of $\zeta(T)$, we approximate the arithmetic mean by the geometric mean as follows:

$$\zeta(T)\approx\exp\left\{\frac{1}{\Delta}\int_T^{T+\Delta}\mathbb{E}[\log v_u|\mathcal{F}_T]\,du\right\}.$$

Let $y_u = \log v_u$ and recall that $\gamma = \frac{1}{2} - H$. Apart from $\mathcal{F}_t$ measurable terms (abbreviated as ``drift''), we have

$$\int_T^{T+\Delta} E[y_u|\mathcal{F}_T]\,du = \eta\sqrt{2H}\int_t^T\frac{dW_s}{(u-s)^\gamma}\,du + \text{drift}$$

$$= \eta\sqrt{2H}\int_t^T\int_T^{T+\Delta}\frac{du}{(u-s)^\gamma}\,dW_s + \text{drift}$$

$$= \eta\frac{\sqrt{2H}}{1-\gamma}\int_t^T\left[(T+\Delta-s)^{1-\gamma} - (T-s)^{1-\gamma}\right]dW_s + \text{drift}.$$

This gives

$$\text{var}[\log\zeta(T)|\mathcal{F}_t]\approx\frac{\eta^2}{\Delta^2}\frac{2H}{(H+1/2)^2}\int_t^T\left[(T+\Delta-s)^{1/2+H} - (T-s)^{1/2+H}\right]^2 ds$$

$$= \eta^2\,(T-t)^{2\,H}f^H\left(\frac{\Delta}{T-t}\right)$$

where

$$f^H(\theta) = \frac{2H}{(H+1/2)^2}\frac{1}{\theta^2}\int_0^1\left[(1+\theta-x)^{1/2+H} - (1-x)^{1/2+H}\right]^2 dx.$$

Calibration of H from VIX futures

- Both [Bayer, Friz and Gatheral][1] and [Jacquier, Martini and Muguruza][5]</sup> find that the volatility of volatility parameter η estimated by calibration to the SPX is roughly 20% greater than the estimate from VIX futures and options.
  - Arbitrage or model mis-specification?
- However, calibration to VIX futures could be a very quick and practical way of fixing H.

- If H is fixed, calibration using the Hybrid BSS scheme becomes fast and efficient.
  - The expensive part is the simulation of W^H which is fixed if H is fixed.

R implementation

Here is a function that takes a total variance curve as input and outputs VIX future prices for various maturities, given H and ν.

```
In [32]: f.vix

function (expiries, w, vixMaturities)
function(H, nu) {
    conv.adj <- vixConvexity(H, nu, vixMaturities, delta = 1/12)
    ev2 <- eVix2(expiries, w)(vixMaturities)
    sqrt(ev2) * conv.adj * 100
}
```

VIX futures on January 4, 2016

VIX Futures Historical Prices
vixcentral.com

vixcentral.com

Load VIX futures data

```
In [33]:  load("VIX-futures.rData")
```

```
In [34]:  vix.fut.20160104
```

Contract	Cdays	Tdays	Price
Jan	15	12	19.825
Feb	43	32	19.525
Mar	71	52	19.525
Apr	106	77	19.775
May	134	97	19.925
Jun	162	117	20.100
Jul	197	142	20.400
Aug	225	162	20.425
Sep	260	187	20.850

Compute VIX futures as of January 4, 2016 given H and ν

```
In [35]: (vix.t <- vix.fut.20160104$Cdays/365.25)
```

0.0410677618069815   0.11772758384668   0.194387405886379
0.290212183436003   0.366872005475702   0.4435318275154
0.539356605065024   0.616016427104723   0.711841204654346

Now compute VIX futures to these dates:

```
In [36]: f.vix(vsData.20160104$texp,vsData.20160104$vs*vsData.20160104$texp,
             vixMaturities=vix.t)(H=0.01,nu=3)
```

19.460172306998   20.091495223692   20.2627296849605   20.0280964596902
20.0814026159928   20.8931173006634   20.1570180729451   20.0427861789934
20.5664820341733

Find optimal H and ν

```
In [37]: fVIX <- function(paramvec){

             H <- paramvec[1]
             nu <- paramvec[2]
             return(f.vix(vsData.20160104$texp,vsData.20160104$vs*vsData.20160104
         $texp,
                 vixMaturities=vix.t)(H,nu))
         }
```

```
In [38]: fVIX(c(.01,3))
```

19.460172306998   20.091495223692   20.2627296849605   20.0280964596902
20.0814026159928   20.8931173006634   20.1570180729451   20.0427861789934
20.5664820341733

```
In [39]: obj <- function(paramvec){

             fVIX.model <- fVIX(paramvec)
             fVIX.actual <- vix.fut.20160104$Price
             return(sum((fVIX.model-fVIX.actual)^2)*1e6)
         }
```

```
In [40]: obj(c(.01,3))
```

2001325.87971786

```
In [41]:  (res.optim <- optim(c(.05,3),obj,
                     method="L-BFGS-B",lower=c(0.0001,0.01),upper=c(.49,10)))
```

$par
　　0.0109595897146603　2.94112857204862

$value
1895730.10634589

$counts

function	83
gradient	83

$convergence
0

$message
'CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH'

Compare model with market

- Our optimization gives $H = 0.0110$, $\nu = 2.941$.

- [Jacquier, Martini and Muguruza][5]</sup> find $H = 0.0509$, $\nu = 1.2937$.
- Only the produce $\tilde{\nu} = \nu\sqrt{2H}$ is really relevant.
  - Our fit gives $\tilde{\nu} = 0.31$, theirs gives $\tilde{\nu} = 0.29$.

```
In [42]:  (fVIX.fit <- fVIX(res.optim$par))
```

　　19.4247191396411　20.0237231871493　20.1754444362798　19.9253449738318
　　19.968170363363　20.7664364807375　20.0257491692953　19.9060236556735
　　20.4190736112191

```
In [43]:  plot(vix.t,vix.fut.20160104$Price,col="blue",cex=2,pch=20,ylim=c(19,21),
          ylab="VIX",xlab="Maturity")
          points(vix.t,fVIX.fit,col="red",pch=20,type="b")
          legend("topleft", c("VIX Futures (Rough Bergomi)","VIX Futures (Market)"
          ), pch = 20,
                col=c("red","blue") ,inset = .02)
```
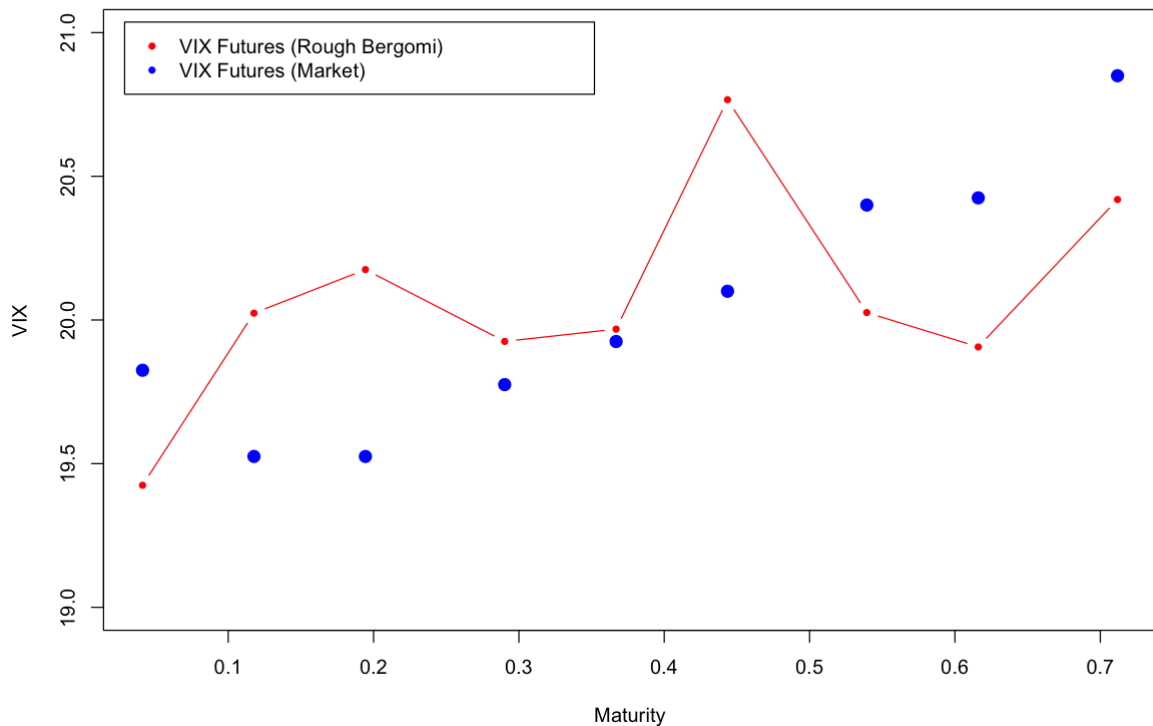

Figure 9: VIX futures as of January 4, 2016. Rough Bergomi fit in red vs market prices in blue.

Remarks

- We don't reproduce the nice futures curve in [Jacquier, Martini and Muguruza][5]</sup>.
- However, we do more or less agree with their fitted parameters.

- It seems that they really did find a nice way to fix H from the VIX futures market.

Summary

- The rough volatility forecast is both simpler than and superior to other available volatility forecasts.

- In the context of the rough Bergomi model, we can use this forecast of future variance under $\mathbb{P}$ to forecast the forward variance curve.

- In particular, we can forecast the term structure of VIX futures.

References

1. ^ Christian Bayer, Peter Friz and Jim Gatheral, Pricing under rough volatility, *Quantitative Finance* **16**(6) 887-904 (2016).
2. ^ Mikkel Bennedsen, Asger Lunde, and Mikko S. Pakkanen, Decoupling the short-and long-term behavior of stochastic volatility, (2016).
3. ^Jim Gatheral, *The Volatility Surface: A Practitioner's Guide*, John Wiley and Sons, Hoboken, NJ (2006).
4. ^ Jim Gatheral, Thibault Jaisson and Mathieu Rosenbaum, Volatility is rough, *Quantitative Finance* **18**(6) 933-949 (2018).
5. ^ Antoine Jacquier, Claude Martini, and Aitor Muguruza, On VIX Futures in the rough Bergomi model, *Quantitative Finance* **18**(1) 45-61 (2018).
6. ^ Carl J. Nuzman and H. Vincent Poor, Linear estimation of self-similar processes via Lamperti's transformation, *Journal of Applied Probability* **37**(2) 429-452 (2000).

```
In [ ]:
```