10th European Summer School in Financial Mathematics

Dresden, August 28, 2017

Rough volatility

Lecture 1: Econometrics and forecasting

Jim Gatheral
Department of Mathematics

Baruch College
The City University of New York

Outline of Lecture 1

- The briefest possible introduction to R and iPython notebook
- The time series of historical volatility
  - Scaling properties
  - Approximate normality of increments of log volatility
- Approximate multifractality
- Estimation of H
- Forecasting

What is R? (http://cran.r-project.org (http://cran.r-project.org))

From Wikipedia:

- In computing, R is a programming language and software environment for statistical computing and graphics. It is an implementation of the S programming language with lexical scoping semantics inspired by Scheme.
- R was created by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand, and is now developed by the R Development Core Team. It is named partly after the first names of the first two R authors (Robert Gentleman and Ross Ihaka), and partly as a play on the name of S. The R language has become a de facto standard among statisticians for the development of statistical software.

- R is widely used for statistical software development and data analysis. R is part of the GNU project, and its source code is freely available under the GNU General Public License, and pre-compiled binary versions are provided for various operating systems. R uses a command line interface, though several graphical user interfaces are available.

The IPython Notebook (http://ipython.org/notebook.html (http://ipython.org/notebook.html))

From ipython.org:

The IPython Notebook is a web-based interactive computational environment where you can combine code execution, text, mathematics, plots and rich media into a single document:

The IPython notebook with embedded text, code, math and figures. These notebooks are normal files that can be shared with colleagues, converted to other formats such as HTML or PDF, etc. You can share any publicly available notebook by using the IPython Notebook Viewer service which will render it as a static web page. This makes it easy to give your colleagues a document they can read immediately without having to install anything.

http://nbviewer.ipython.org/github/dboyliao/cookbook-code/blob/master/notebooks/chapter07_stats/08_r.ipynb (http://nbviewer.ipython.org/github/dboyliao/cookbook-code/blob/master/notebooks/chapter07_stats/08_r.ipynb) has instructions on using R with iPython notebook.

The econometrics of rough volatility

- In this lecture, we will explore the scaling properties of the time series of historical volatility.
- With very few reasonable assumptions, these scaling relationships lead to a *unique* model of volatility under the physical measure $\mathbb{P}$.

The time series of realized variance

- We would like to study the time series of instantaneous variance v_t but of course cannot because v_t is latent.
- On the other hand, integrated variance $\frac{1}{\delta}\int_t^{t+\delta} v_s\, ds$ may (in principle) be estimated arbitrarily accurately given enough price data.
  - In practice, market microstructure noise makes estimation harder at very high frequency.
  - Sophisticated estimators of integrated variance have been developed to adjust for market microstructure noise. See Gatheral and Oomen [8] (for example) for details of these.

- The Oxford-Man Institute of Quantitative Finance makes historical realized variance (RV) estimates freely available at http://realized.oxford-man.ox.ac.uk (http://realized.oxford-man.ox.ac.uk). These estimates are updated daily.

- Each day, for 21 different indices, all trades and quotes are used to estimate realized (or integrated) variance over the trading day from open to close.

- Using daily RV estimates as proxies for instantaneous variance, we may investigate the time series properties of integrated variance empirically.

First update and save the latest Oxford-Man data:

In [1]:
```
download.file(url="http://realized.oxford-man.ox.ac.uk/media/1366/oxfordmanre
unzip(zipfile="oxfordRvData.zip")
```

There are many different estimates of realized variance, all of them very similar. We will use the realized kernel estimates denoted by ".rk".

In [2]:
```
library(quantmod)
```

```
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 [3]:
```
rv.data <- read.csv("OxfordManRealizedVolatilityIndices.csv")
colnums <- which(sapply(rv.data, function(x) grep(".rk",x))>0)
col.names <- names(colnums)

rv1 <- rv.data[,colnums]
index.names <- rv1[2,]

datesRaw <- rv.data[-(1:2),1]
dates <- strptime(datesRaw,"%Y%m%d")

rv.list <- NULL
index.names <- as.matrix(index.names)

n <- length(index.names)
for (i in 1:n){
    tmp.krv1 <- xts(rv1[-(1:2),i],order.by=dates)
    rv.list[[i]] <- tmp.krv1[(tmp.krv1!="")&(tmp.krv1!="0")]
}

names(rv.list)<- index.names

save(rv.list, file="oxfordRV.rData")
```

Let's plot SPX realized variance.

In [4]:
```
# Load Oxford-Man KRV data
load("oxfordRV.rData")
names(rv.list)

spx.rk <- rv.list[["SPX2.rk"]]
stoxx.rk <- rv.list[["STOXX50E.rk"]]
```

'SPX2.rk' 'FTSE2.rk' 'N2252.rk' 'GDAXI2.rk' 'RUT2.rk' 'AORD2.rk' 'DJI2.rk' 'IXIC2.rk' 'FCHI2.rk' 'HSI2.rk' 'KS11.rk' 'AEX.rk' 'SSMI.rk' 'IBEX2.rk' 'NSEI.rk' 'MXX.rk' 'BVSP.rk' 'GSPTSE.rk' 'STOXX50E.rk' 'FTSTI.rk' 'FTSEMIB.rk'

In [5]:
```
library(repr)
options(repr.plot.width=14,repr.plot.height=8)
```

In [6]:
```
plot(spx.rk, main="SPX realized variance",plot=NULL)
lines(spx.rk,col="red")
```

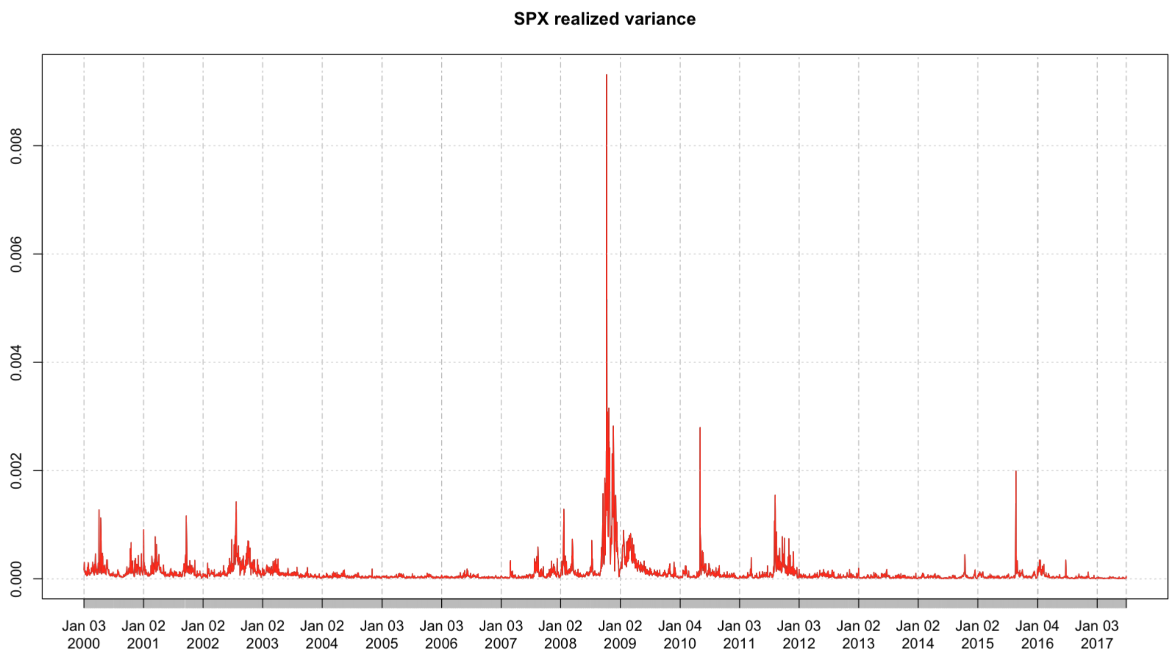


Figure 1: Oxford-Man KRV estimates of SPX realized variance from January 2000 to the current date.

In [7]:
```
print(head(spx.rk))
print(tail(spx.rk))
```

```
                  [,1]
2000-01-03 "0.000160726642338866"
2000-01-04 "0.000264396469319473"
2000-01-05 "0.000304650302935347"
2000-01-06 "0.000148582063339039"
2000-01-07 "0.000123266970191763"
2000-01-10 "0.000130693391920629"
                  [,1]
2017-06-23 "7.22920707078143E-06"
2017-06-26 "1.29697604360725E-05"
2017-06-27 "1.94994906185805E-05"
2017-06-28 "1.16316730312408E-05"
2017-06-29 "4.75396805951542E-05"
2017-06-30 "1.31704798919367E-05"
```

Scaling of the volatility process

For $q \geq 0$, we define the qth sample moment of differences of log-volatility at a given lag Δ.($\langle \cdot \rangle$ denotes the sample average):

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

For example

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

is just the sample variance of differences in log-volatility at the lag Δ.

Scaling of $m(q, \Delta)$ with lag Δ

In [8]:
```
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)})}

# Plot mq.del(1:100,q) for various q

x <- 1:100

mycol <- rainbow(5)

ylab <- expression(paste(log," ",m(q,Delta)))
xlab <- expression(paste(log, " ", Delta))

qVec <- c(.5,1,1.5,2,3)
zeta.q <- numeric(5)
q <- qVec[1]
```

In [9]:
```
options(repr.plot.height=7, repr.plot.width=10)
```

In [10]:
```
plot(log(x),log(mq.del(x,q)),pch=20,cex=.5,
          ylab=ylab, xlab=xlab,ylim=c(-3,-.5))
fit.lm <- lm(log(mq.del(x,q)) ~ log(x))
abline(fit.lm, col=mycol[1],lwd=2)
zeta.q[1] <- coef(fit.lm)[2]

for (i in 2:5){
    q <- qVec[i]
    points(log(x),log(mq.del(x,q)),pch=20,cex=.5)
    fit.lm <- lm(log(mq.del(x,q)) ~ log(x))
    abline(fit.lm, col=mycol[i],lwd=2)
    zeta.q[i] <- coef(fit.lm)[2]
    }
 legend("bottomright", c("q = 0.5","q = 1.0","q = 1.5","q = 2.0","q = 3.0"),:

print(zeta.q)
```

```
[1] 0.07045758 0.13775850 0.20181204 0.26268569 0.37540217
```

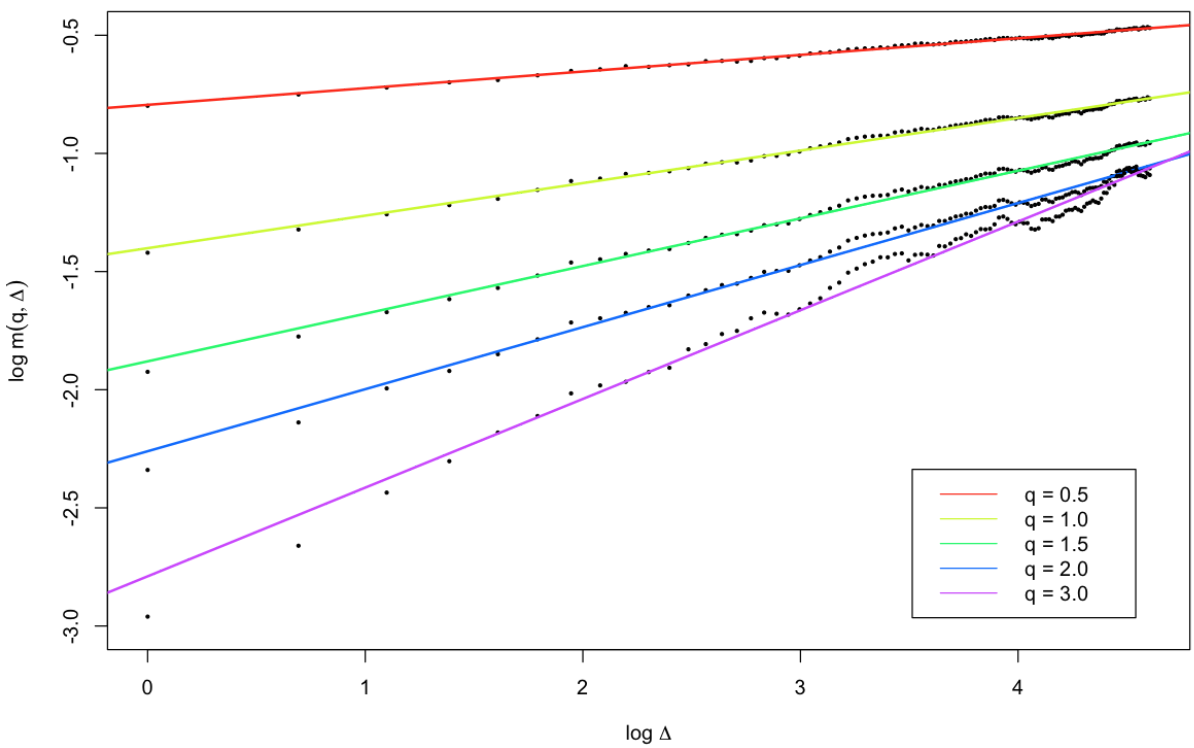


Figure 2: $\log m(q,\Delta)$ as a function of $\log \Delta$, SPX.

Monofractal scaling result

- From the above log-log plot, we see that for each q, $m(q,\Delta) \propto \Delta^{\zeta_q}$.
- How does ζ_q scale with q?

Scaling of ζ_q with q

In [11]:
```
plot(qVec,zeta.q,xlab="q",ylab=expression(zeta[q]),pch=20,col="blue",cex=2)
fit.lm <- lm(zeta.q[1:4] ~ qVec[1:4]+0)
abline(fit.lm, col="red",lwd=2)
(h.est <- coef(fit.lm)[1])
```

qVec[1:4]: 0.133476898325346

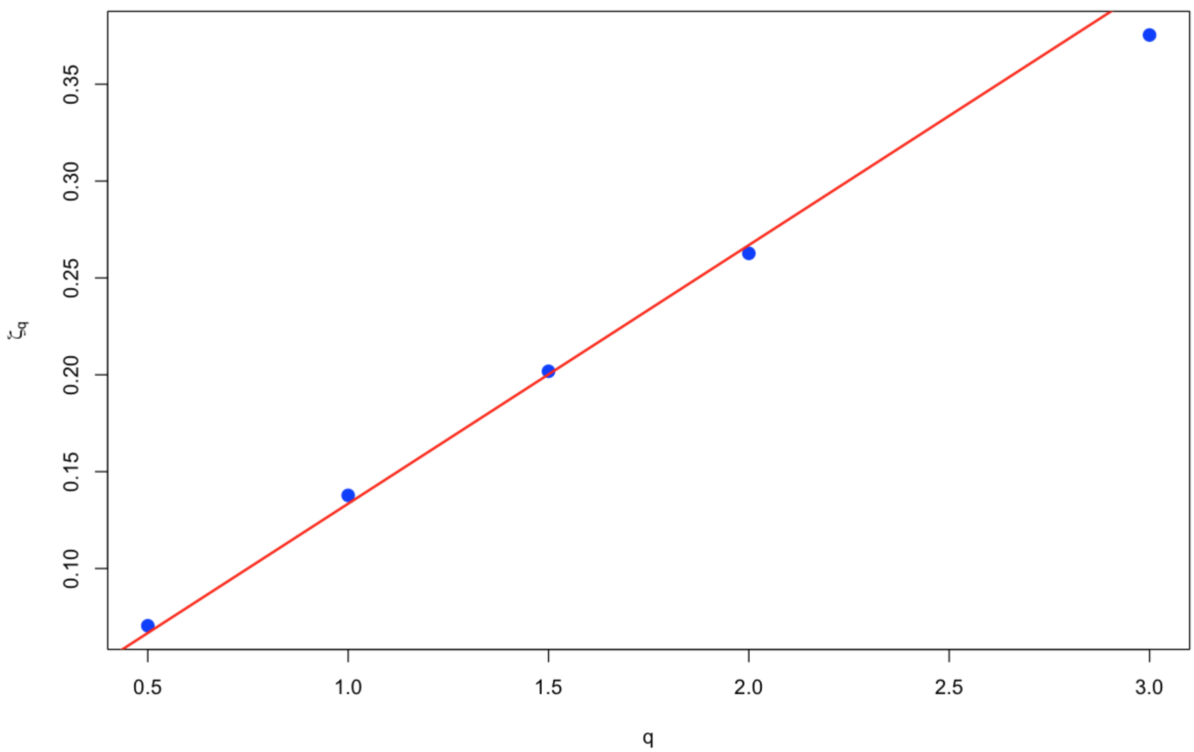


Figure 3: Scaling of ζ_q with q.

We find the monofractal scaling relationship

$$\zeta_q = q\,H$$

with $H \approx 0.13$.

- Note however that H does vary over time, in a narrow range, as we will see later.
- Note also that our estimate of H is biased high because we proxied instantaneous variance v_t with its average over each day $\frac{1}{T}\int_0^T v_t\,dt$, where T is one trading day.
  - On the other hand, the time series of realized variance is noisy and this causes our estimate of H to be biased low.

- This scaling property as $\Delta \to 0$ is equivalent to H-Hölder continuity of paths of the volatility.
  - Since $H \ll 1/2$, *volatility is rough*!

Estimated H for all indices

We now repeat this analysis for all 21 indices in the Oxford-Man dataset.

In [12]:
```
n <- length(rv.list)
h <- numeric(n) # H is estimated as half of the slope
nu <- numeric(n)

for (i in 1:n){ # Run all the regressions
  v <- rv.list[[i]]
  sig <- sqrt(as.numeric(v))

  x <- 1:100
  dlsig2 <- function(lag){mean((diff(log(sig),lag=lag))^2)}
  dlsig2Vec <- function(x){sapply(x,dlsig2)}

  fit.lm <- lm(log(dlsig2Vec(x)) ~ log(x))

  nu[i] <- sqrt(exp(coef(fit.lm)[1]))
  h[i] <- coef(fit.lm)[2]/2

}
```

In [13]:
```
(OxfordH <- data.frame(names(rv.list),h.est=h,nu.est=nu))
```

names.rv.list.	h.est	nu.est
SPX2.rk	0.13134285	0.3229718
FTSE2.rk	0.14167059	0.2679552
N2252.rk	0.11038885	0.3270906
GDAXI2.rk	0.14684291	0.2767171
RUT2.rk	0.11860173	0.3305500
AORD2.rk	0.08150011	0.3601606
DJI2.rk	0.12925495	0.3177534
IXIC2.rk	0.12481339	0.2982734
FCHI2.rk	0.12745028	0.2935064
HSI2.rk	0.10098560	0.2802403
KS11.rk	0.11967108	0.2798421
AEX.rk	0.14233346	0.2918653
SSMI.rk	0.17712822	0.2218513
IBEX2.rk	0.12585654	0.2829205
NSEI.rk	0.10978364	0.3217780
MXX.rk	0.08940493	0.3257193
BVSP.rk	0.10428838	0.3150431
GSPTSE.rk	0.11669892	0.3064453
STOXX50E.rk	0.11656293	0.3396190
FTSTI.rk	0.12710425	0.2289036
FTSEMIB.rk	0.13246218	0.2962336

In [14]:
```
save(OxfordH,file="OxfordH.rData")
```

Distributions of $(\log \sigma_{t+\Delta} - \log \sigma_t)$ for various lags Δ

Having established these beautiful scaling results for the moments, how do the histograms look?

In [15]:
```
plotScaling <- function(j,scaleFactor){
  v <- as.numeric(rv.list[[j]])
  x <- 1:100

  xDel <- function(x,lag){diff(x,lag=lag)}
  sd1 <- sd(xDel(log(v),1))
  sdl <- function(lag){sd(xDel(log(v),lag))}

  h <- OxfordH$h.est[j]

  plotLag <- function(lag){
    y <- xDel(log(v),lag)
    hist(y,breaks=100,freq=F,main=paste("Lag =",lag,"Days"),xlab=NA)# Very l
    curve(dnorm(x,mean=mean(y),sd=sd(y)),add=T,col="red",lwd=2)
    curve(dnorm(x,mean=0,sd=sd1*lag^h),add=T,lty=2,lwd=2,col="blue")
  }

  (lags <- scaleFactor^(0:3))
  print(names(rv.list)[j])
  par(mfrow=c(2,2))
  par(mar=c(3,2,1,3))
  for (i in 1:4){plotLag(lags[i])}
  par(mfrow=c(1,1))
}
```

In [16]:
```
options(repr.plot.height=5, repr.plot.width=10)
```

In [17]:
```
plotScaling(1,5)
```

```
[1] "SPX2.rk"
```

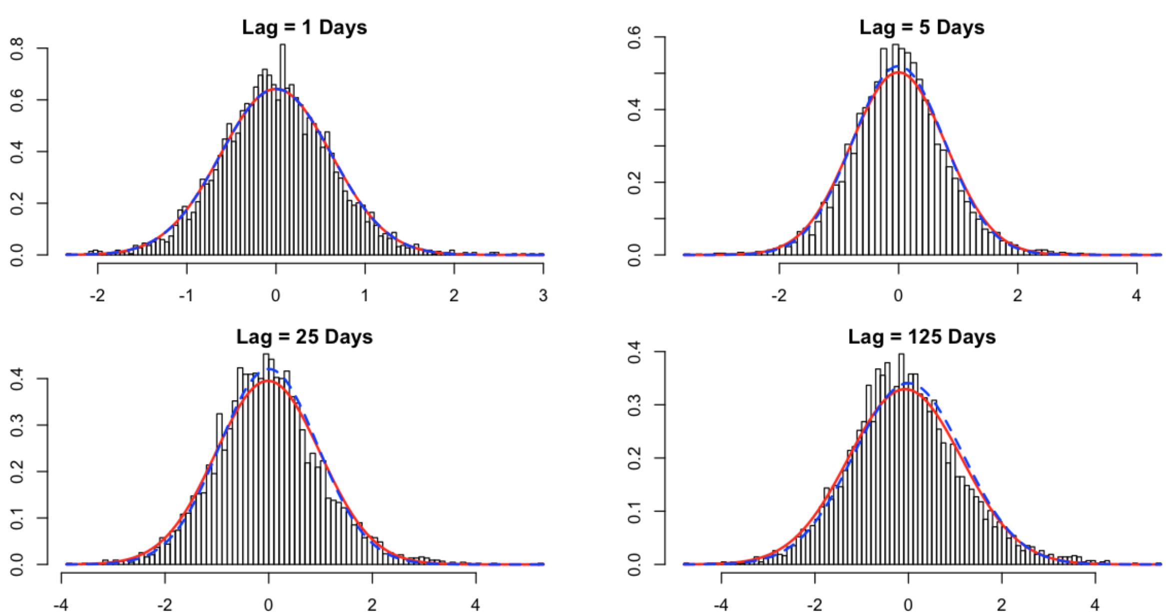


Figure 4: Histograms of $(\log \sigma_{t+\Delta} - \log \sigma_t)$ for various lags Δ; normal fit in red; $\Delta = 1$ normal fit scaled by Δ^H in blue.

Universality?

- [Gatheral, Jaisson and Rosenbaum][7] compute daily realized variance estimates over one hour windows for DAX and Bund futures contracts, finding similar scaling relationships.
- We have also checked that Gold and Crude Oil futures scale similarly.
  - Although the increments $(\log \sigma_{t+\Delta} - \log \sigma_t)$ seem to be fatter tailed than Gaussian.
- [Bennedsen et al.][2], estimate volatility time series for more than five thousand individual US equities, finding rough volatility in every case.

A microstructural explanation: A Hawkes model of price formation

- Why might rough volatility be universal?
- [Jaisson and Rosenbaum][9] show that rough volatility can be obtained as a scaling limit of a simple model of price dynamics in terms of Hawkes processes.
- Remarkably, [El Euch and Rosenbaum][6] were able to compute the characteristic function of the resulting *rough Heston* model.
- Mathieu will go into much more detail in his lectures...

A natural model of realized volatility

- Distributions of differences in the log of realized variance are close to Gaussian.
  - This motivates us to model $\sigma_t = \log v_t$ as a lognormal random variable.
- Moreover, the scaling property of variance of RV differences suggests the model:
  (1)
  $$\log \sigma_{t+\Delta} - \log \sigma_t = \nu \left(W^H_{t+\Delta} - W^H_t \right)$$
  where W^H is fractional Brownian motion.
- Indeed, if H is constant, (1) is the *unique* model consistent with Gaussianity of log differences, the observed scaling, and continuity of the volatility process.

Fractional Brownian motion (fBm)

- *Fractional Brownian motion* (fBm) $\{W^H_t; t \in \mathbb{R}\}$ is the unique Gaussian process with mean zero and autocovariance function
  $$\mathbb{E}\left[W^H_t \, W^H_s\right] = \frac{1}{2}\left\{|t|^{2\,H} + |s|^{2\,H} - |t-s|^{2\,H}\right\}$$
  where $H \in (0,1)$ is called the *Hurst index* or parameter.
  - In particular, when $H = 1/2$, fBm is just Brownian motion.
  - If $H > 1/2$, increments are positively correlated ("trending").
  - If $H < 1/2$, increments are negatively correlated ("reverting").

Representations of fBm

There are infinitely many possible representations of fBm in terms of Brownian motion. For example, with $\gamma = \frac{1}{2} - H$,

Mandelbrot-Van Ness

$$W_t^H = C_H \left\{ \int_{-\infty}^t \frac{dW_s}{(t-s)^\gamma} - \int_{-\infty}^0 \frac{dW_s}{(-s)^\gamma} \right\}$$

where the choice

$$C_H = \sqrt{\frac{2\,H\,\Gamma(3/2-H)}{\Gamma(H+1/2)\,\Gamma(2-2\,H)}}$$

ensures that

$$\mathbb{E}\left[W_t^H\,W_s^H\right] = \frac{1}{2}\left\{t^{2H} + s^{2H} - |t-s|^{2H}\right\}.$$

Efficient estimation of H

- So far, we just used simple regression to estimate H.
- When H is small, as we find empirically, out of all the estimators that we tested, the ACF estimator adopted by [Bennedsen et al.][4] is the most efficient.

Heuristic derivation of the ACF estimator

Once again, the covariance structure of fBm is given by

$$\mathbb{E}\left[W_t^H\,W_s^H\right] = \frac{1}{2}\left\{t^{2H} + s^{2H} - |t-s|^{2H}\right\}.$$

Up to a multiplicative factor, our model is

$$y_t = \log v_t = W_t^H.$$

Then $\mathrm{var}[y_t] = t^{2H}$. and

$$\mathrm{cov}[y_t, y_{t+\Delta}] = \frac{1}{2}\left\{t^{2H} + (t+\Delta)^{2H} - \Delta^{2H}\right\}$$

Dividing one by the other gives

$$\rho(\Delta) = \frac{1}{2}\left\{1 + \left(1+\frac{\Delta}{t}\right)^{2H} - \left(\frac{\Delta}{t}\right)^{2H}\right\}$$

Thus, for Δ/t sufficiently small,

$$1 - \rho(\Delta) = \frac{1}{2}\left(\frac{\Delta}{t}\right)^{2H} + O\left(\frac{\Delta}{t}\right).$$

- Note in particular that we expect the ACF estimator to work best when $H \ll \frac{1}{2}$.
- Also, when $H = \frac{1}{2}$, we have $\rho(\Delta) = 1$ as we would expect for Brownian motion.

The ACF estimator

Taking logs of each side, we obtain

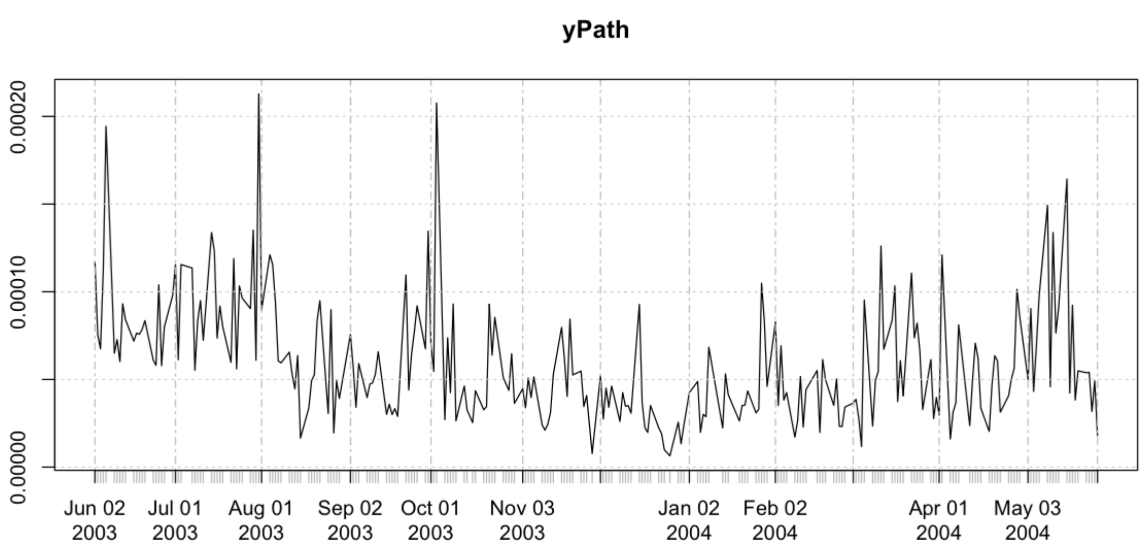
$$\log(1 - \rho(\Delta)) = a + 2\,H\ \log\Delta.$$

- Thus H can be estimated efficiently by regression.

In [18]:
```
h.acf <- function(path){
    y.acf <- acf(path,plot=F)
    log.del <- log(y.acf$lag[-1])
    log.lhs <- log(1-y.acf$acf[-1])
    fit.lm <- lm(log.lhs ~ log.del)
    return(fit.lm$coef[2]/2)
        }
```

An example

In [19]:
```
yPath <- spx.rk["2003-06-01::2004-05-31"]
plot(yPath)
```



In [20]:
```
h.acf(as.numeric(yPath))
```

log.del: 0.0601516764681327

Time series of H using ACF

- We now draw the time series of H using the ACF estimator.

In [21]:
```
h.acf.i <- function(series)function(del)function(i){
    rk.path <- as.numeric(series[(i-del):i])
    h.acf(rk.path)
    }
```

In [22]:
```
h.acf.i(spx.rk)(252)(1234)
```

log.del: 0.0877924997525288

In [23]:
```
h.acf.series <- function(series)function(del){
    require(xts)
    n <- length(series)
    res <- sapply((1+del):n,h.acf.i(series)(del))
    return(xts(res,order.by=index(series[(1+del):length(series)]),tzone = Sy
}
```

Compare the two estimates of H

In [24]:
```
n.spx <- length(spx.rk)
h.spx.acf <- as.numeric(h.acf.series(spx.rk)(n.spx-1))
h.spx.regression <- OxfordH$h.est[1]
nu.spx.regression <- OxfordH$nu.est[1]
data.frame(h.spx.acf,h.spx.regression)
```

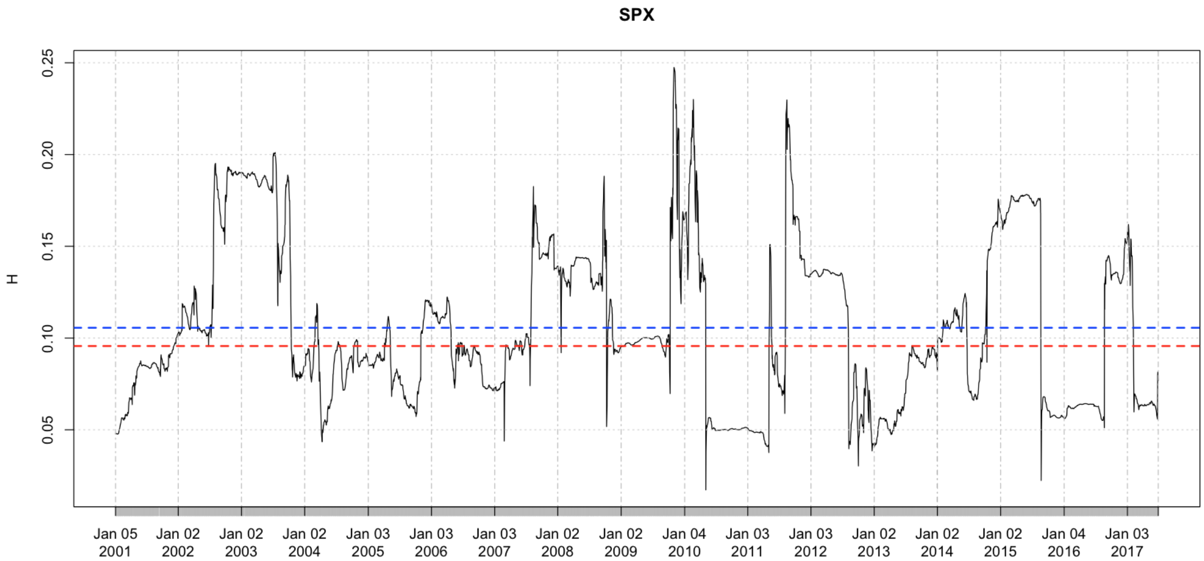
h.spx.acf	h.spx.regression
0.09801913	0.1313428

- Looking again at the log-log plots of $m_q(\Delta)$ against Δ, we note that the points don't quite lie on a straight line.
- A more careful analysis that takes account of the bias due to averaging and the noisiness of the time series of realized variance gives us an estimate of H more consistent with the ACF estimate.

Time series of H for SPX

In [25]:
```
h.spx.252 <-  h.acf.series(spx.rk)(252)
```

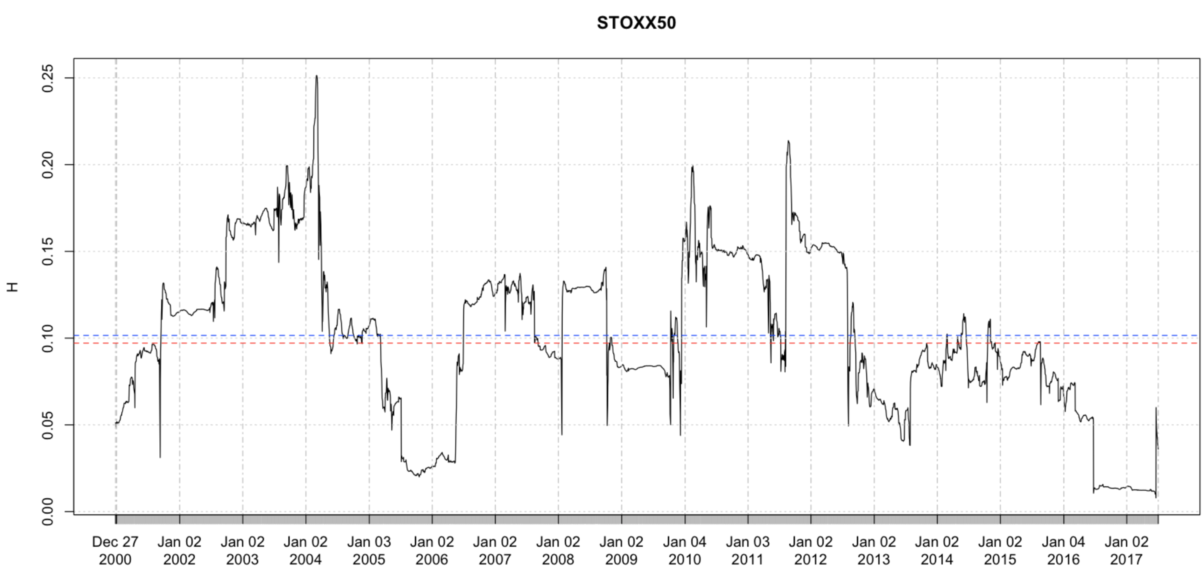
In [26]:
```
options(repr.plot.width=14,repr.plot.height=7)
plot(h.spx.252,main="SPX",ylab="H")
abline(h=median(h.spx.252),lty=2,col="red",lwd=2)
abline(h=mean(h.spx.252),lty=2,col="blue",lwd=2)
```



Time series of H for STOXX50

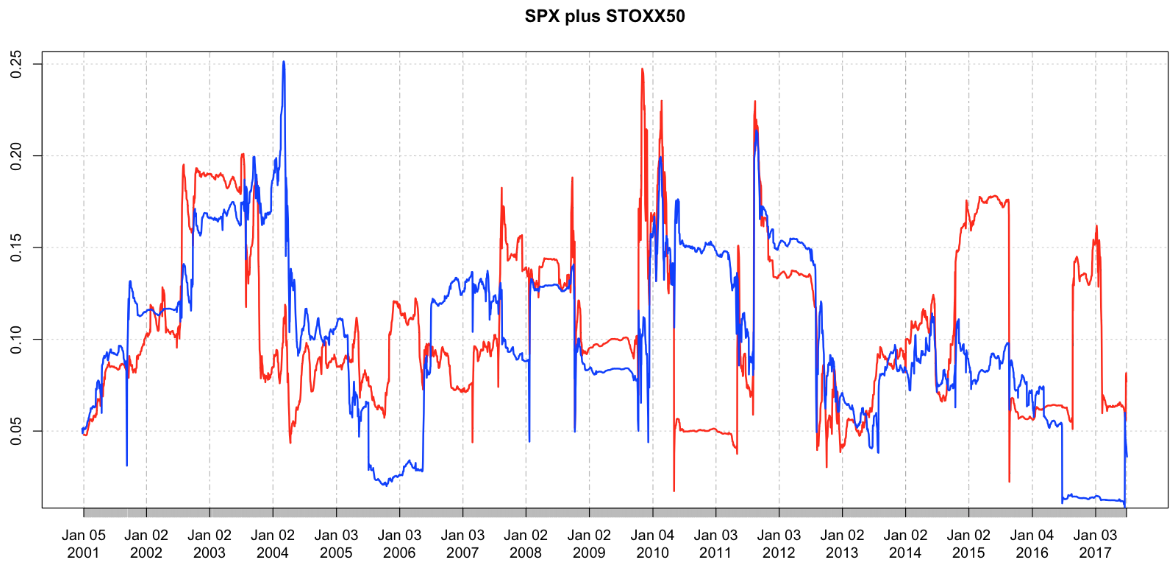
In [27]:
```
h.stoxx.252 <- h.acf.series(stoxx.rk)(252)
```

In [28]:
```
plot(h.stoxx.252,main="STOXX50",ylab="H")
abline(h=median(h.stoxx.252),lty=2,col="red")
abline(h=mean(h.stoxx.252),lty=2,col="blue")
```



Plot both together

In [29]:
```
plot(h.spx.252,type="n",main="SPX plus STOXX50")
lines(h.spx.252,col="red",lwd=2)
lines(h.stoxx.252,col="blue",lwd=2)
```



- Sometimes the peaks line up, and sometimes not.

Line up time series of H with SPX

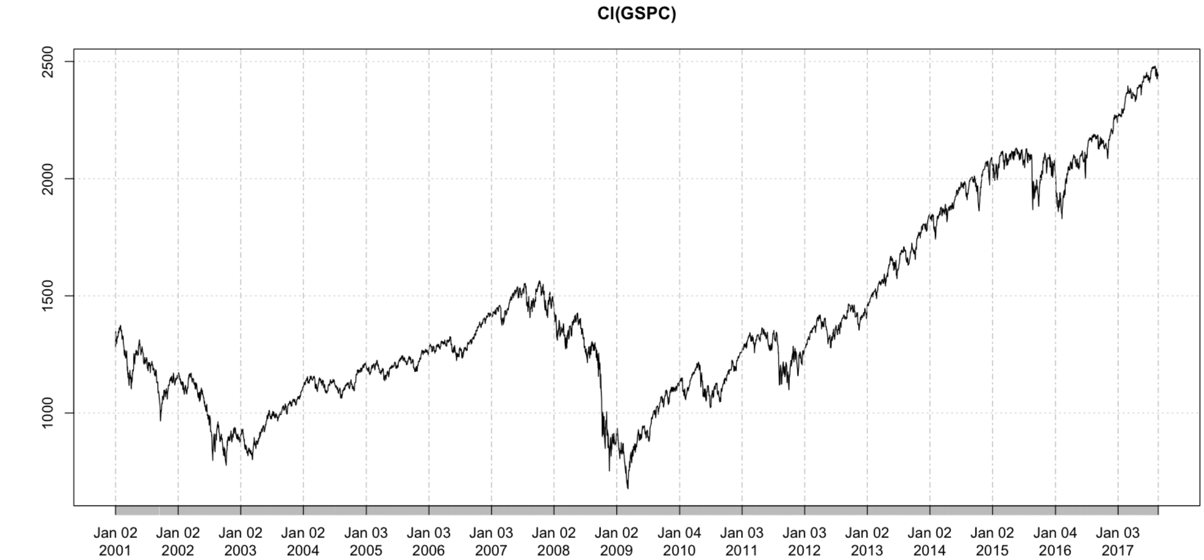
- First we use `quantmod` to download SPX data.

In [30]:
```
getSymbols('^GSPC',from="2001-01-01")
```

```
'getSymbols' currently uses auto.assign=TRUE by default, but will
use auto.assign=FALSE in 0.5-0. You will still be able to use
'loadSymbols' to automatically load data. getOption("getSymbols.env")
and getOption("getSymbols.auto.assign") will still be checked for
alternate defaults.

This message is shown once per session and may be disabled by setting
options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.

WARNING: There have been significant changes to Yahoo Finance data.
Please see the Warning section of '?getSymbols.yahoo' for details.

This message is shown once per session and may be disabled by setting
options("getSymbols.yahoo.warning"=FALSE).
```

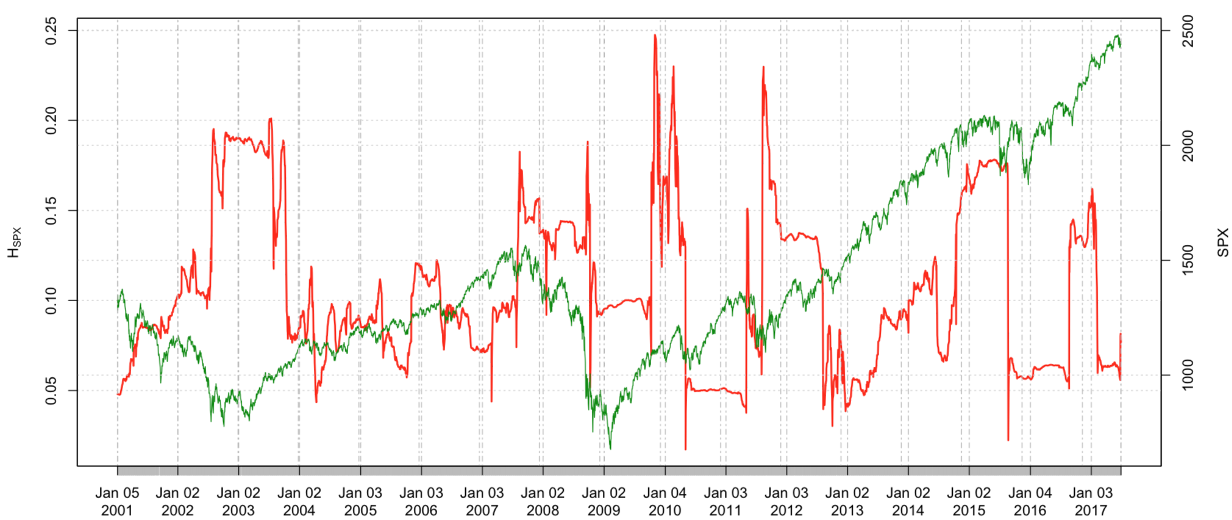
'GSPC'

In [31]:
```
plot(Cl(GSPC))
```



Superimpose the two time series

In [32]:
```
z <- Cl(GSPC)
par(mar = c(5, 4, 4, 4) + 0.3)  # Leave space for z axis
plot(h.spx.252,type="n",ylab=expression(H[SPX]),main=NULL)
lines(h.spx.252,col="red",lwd=2)
par(new = TRUE)
plot(z, type = "l", axes = FALSE, bty = "n", xlab = "", ylab = "",col="green
axis(side=4, at = pretty(range(z)))
mtext("SPX", side=4, line=3)
```



Observations

- H tends to spike when the market is under stress.
  - And seems close to zero when the market is calm.

- Note the following peaks
  - The Greek debt crisis in late 2011.
  - The Brexit vote in 2015. In this case H rises with uncertainty then collapses.

- When the market crashes, H rises. But often H rises without the market crashing.

- In particular, H of the volatility time series seems to be a meaningful and relevant statistic.

Comte and Renault: FSV model

[Comte and Renault][5] were perhaps the first to model volatility using fractional Brownian motion.

In their fractional stochastic volatility (FSV) model,

$$\begin{aligned} \frac{dS_t}{S_t} &= \sigma_t \, dZ_t \\ d \log \sigma_t &= -\alpha \, (\log \sigma_t - \theta) \, dt + \gamma \, d\hat{W}_t^H \end{aligned}$$

with

$$\hat{W}_t^H = \int_0^t \frac{(t-s)^{H-1/2}}{\Gamma(H+1/2)} \, dW_s, \quad 1/2 \leq H < 1$$

and $\mathbb{E}\left[dW_t \, dZ_t\right] = \rho \, dt$.

RFSV and FSV

- The model (1):
  $$\log \sigma_{t+\Delta} - \log \sigma_t = \nu \, \left(W_{t+\Delta}^H - W_t^H \right)$$
  is not stationary.
  - Stationarity is desirable both for mathematical tractability and also to ensure reasonableness of the model at very large times.

- The RFSV model (the stationary version of (1)) is formally identical to the FSV model. Except that
  - $H < 1/2$ in RFSV vs $H > 1/2$ in FSV.
  - $\alpha \, T \gg 1$ in RFSV vs $\alpha \, T \sim 1$ in FSV, where T is a typical timescale of interest.

Heuristic derivation of autocorrelation function

We assume that $\sigma_t = \bar{\sigma}_t \, e^{\nu \, W_t^H}$. Then

$$
\begin{aligned}
&\operatorname{cov}\left[\sigma_t, \sigma_{t+\Delta}\right] \\
&= \bar{\sigma}_t \, \bar{\sigma}_{t+\Delta} \, \left[\exp\left\{ \frac{\nu^2}{2} \left(t^{2H} + (t+\Delta)^{2H} - \Delta^{2H} \right) \right\} - 1\right] \\
&\sim \bar{\sigma}_t \, \bar{\sigma}_{t+\Delta} \, \exp\left\{ \frac{\nu^2}{2} \left(t^{2H} + (t+\Delta)^{2H} - \Delta^{2H} \right) \right\} \text{ as } t \to \infty.
\end{aligned}
$$

Similarly, $\operatorname{var}\left[\sigma_t\right] \sim \bar{\sigma}_t^2 \, \exp\left\{\nu^2 \, t^{2H}\right\}$ as $t \to \infty$. Thus

$$
\rho(\Delta) = \frac{\operatorname{cov}\left[\sigma_t, \sigma_{t+\Delta}\right]}{\sqrt{\operatorname{var}\left[\sigma_t\right] \, \operatorname{var}\left[\sigma_{t+\Delta}\right]}} \sim \exp\left\{ -\frac{\nu^2}{2} \, \Delta^{2H} \right\}.
$$

FSV and long memory

- Why did [Comte and Renault][6] choose $H > 1/2$?
  - Because it has been a widely-accepted stylized fact that the volatility time series exhibits long memory.
- In this technical sense, *long memory* means that the autocorrelation function of volatility decays as a power-law.
- One of the influential papers that established this was [Andersen, Bollerslev, Diebold and Ebens][1] which estimated the degree d of fractional integration from daily realized variance data for the 30 DJIA stocks.
  - Using the GPH (Geweke-Porter-Hudak) estimator, they found d around 0.35 which implies that the ACF $\rho(\tau) \sim \tau^{2\,d-1} = \tau^{-0.3}$ as $\tau \to \infty$.

- But every statistical estimator assumes the validity of some underlying model!
- In the RFSV model,

$$
\rho(\Delta) \sim \exp\left\{ -\frac{\nu^2}{2} \, \Delta^{2H} \right\}.
$$

- Using the same GPH estimator on the Oxford-Man RV data we find $d = 0.48$ which according to their test would indicate extreme long memory. But our model (1) is different from that of [Andersen, Bollerslev, Diebold and Ebens][1]; it does not have long memory.

Correlogram and test of scaling

In [33]:
```
v <- rv.list[[1]]  # Pick spx.rk
sig <- sqrt(as.numeric(v))
aclog <-acf(log(sig),lag=100,plot=F)
plot(aclog$lag[-1],aclog$acf[-1],pch=20,ylab=expression(rho(Delta)),xlab=expr
```

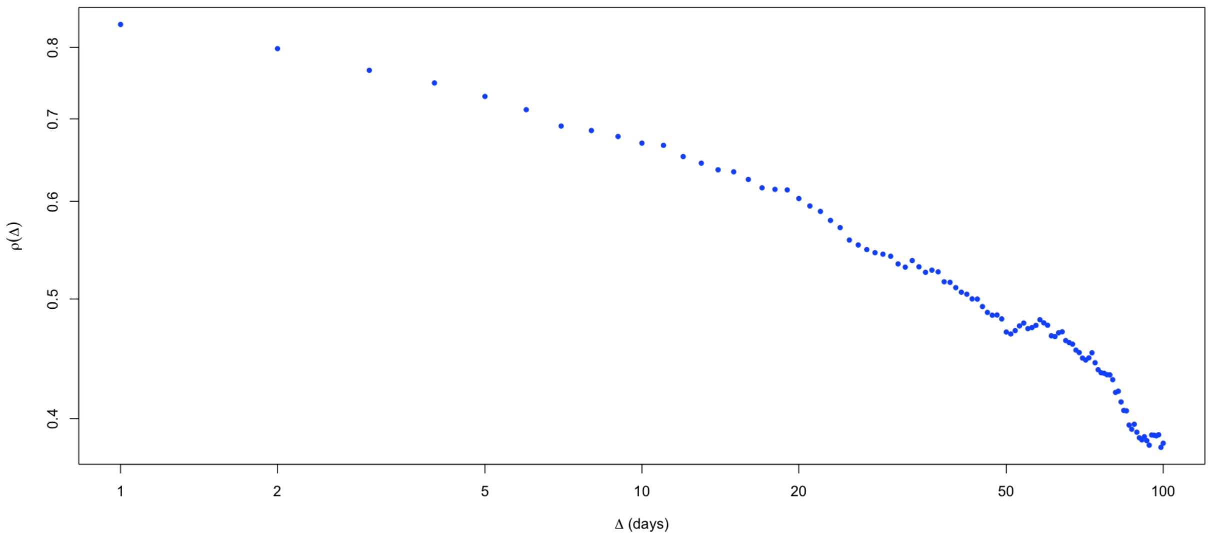


Figure 5: A correlogram of $\sigma_t = \sqrt{RV_t}$; it doesn't look linear!

In [34]:
```
esig2 <- mean(sig)^2
covdel <- acf(sig,lag.max=100,type="covariance",plot=F)$acf[-1]
x <- (1:100)^(2*h.spx.regression)
plot(x,log(covdel+esig2),pch=20,col="dark green",ylab=expression(phi(Delta))
abline(lm(log(covdel+esig2)~x),col="red",lwd=2)
```

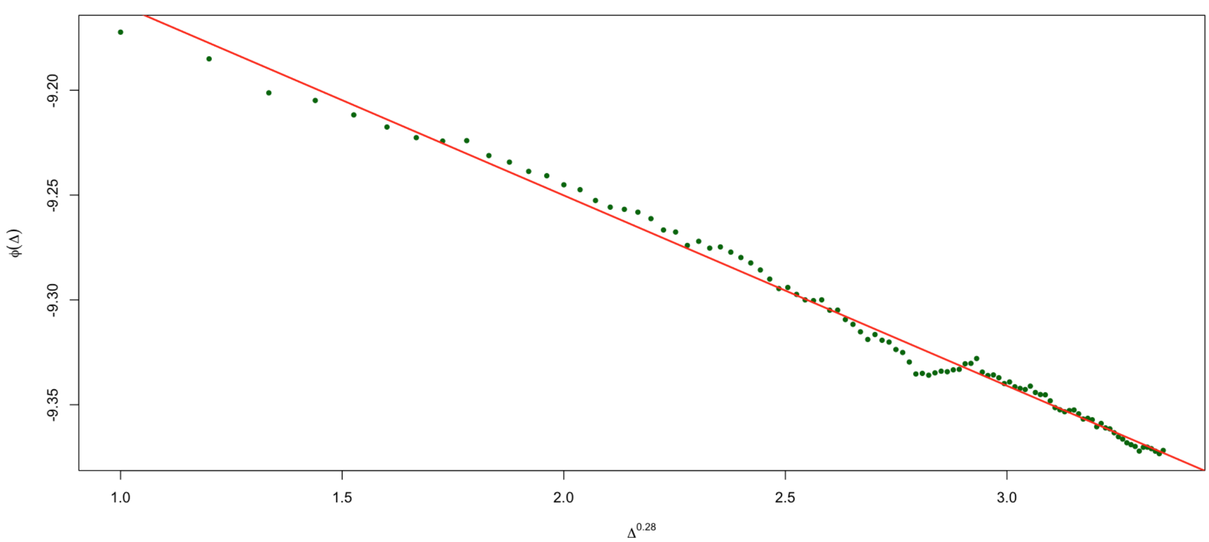


Figure 6: A plot of $\phi(\Delta) := \langle \log\left(\text{cov}(\sigma_{t+\Delta}, \sigma_t) + \langle \sigma_t \rangle^2\right) \rangle$ vs $\Delta^{2\,H}$ with $H \approx 0.14$. Clearly consistent with the scaling relationship $m(2, \Delta) \propto \Delta^{2\,H}$.

Model vs empirical autocorrelation functions

In [35]:
```
v <- rv.list[[1]]  # Pick spx.rk
sig <- sqrt(as.numeric(v))

aclog <-acf(log(sig),lag=100,plot=F)
y <- log(aclog$acf)
x <- aclog$lag^(2*h.spx.regression)
fit.lm <- lm(y[-1]~x[-1])
a <- fit.lm$coef[1]
b <- fit.lm$coef[2]
```

In [36]:
```
plot(aclog$lag[-1],aclog$acf[-1],pch=20,ylab=expression(rho(Delta)),
     xlab=expression(paste(Delta," (days)")),log="xy",col="blue")
curve(exp(a+b*x^(2*h.spx.regression)),from=0.001,to=100,col="red",add=T,lwd=
```

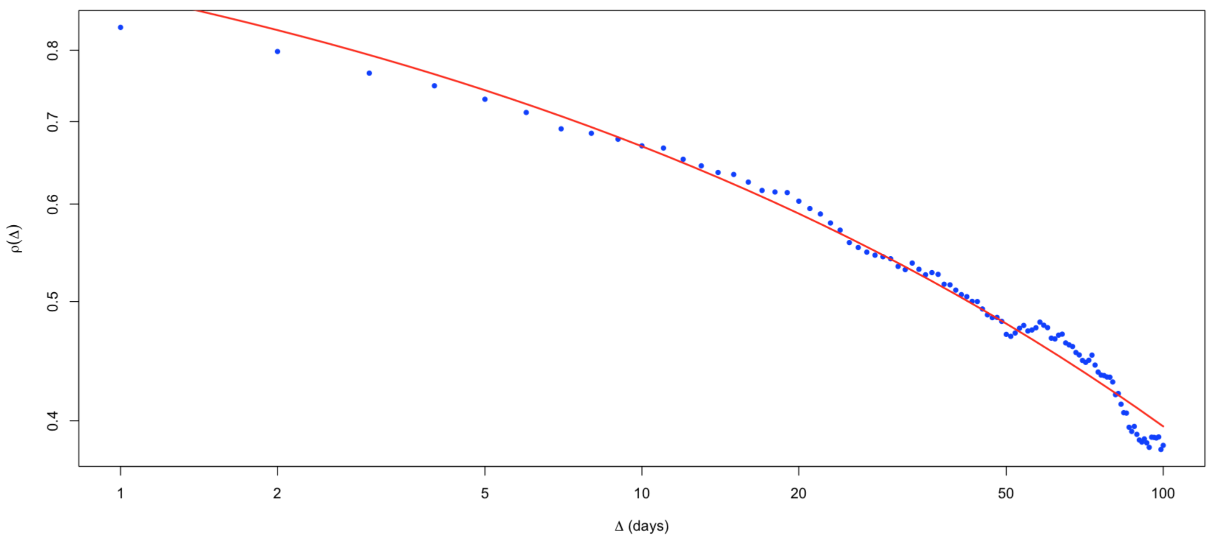


Figure 7: Here we superimpose the RFSV functional form $\rho(\Delta) \sim \exp\left\{-\frac{\nu^2}{2}\,\Delta^{2\,H}\right\}$ (in red) on the empirical curve (in blue).

Long memory of volatility may be spurious

- Looking at Figures 5, 6 and 7, there is no reason to suppose that volatility is long memory.
- Moreover, the RFSV model reproduces the observed autocorrelation function very closely.
- [Gatheral, Jaisson and Rosenbaum][6] further simulate volatility in the RFSV model and apply standard estimators to the simulated data.
  - Real data and simulated data generate very similar plots and similar estimates of the long memory parameter to those found in the prior literature.

- The RSFV model does not have the long memory property.

- Classical estimation procedures seem to identify spurious long memory of volatility.
  - We cannot of course exclude long memory. Our only point is that tests used so far do not support long memory.

Incompatibility of FSV with realized variance (RV) data

- In Figure 8, we demonstrate graphically that long memory volatility models such as FSV with $H > 1/2$ are not compatible with the RV data.
- In the FSV model, the autocorrelation function $\rho(\Delta) \propto \Delta^{2\,H-2}$. Then, for long memory, we must have $1/2 < H < 1$.
  - For $\Delta \gg 1/\alpha$, stationarity kicks in and $m(2,\Delta)$ tends to a constant as $\Delta \to \infty$.
  - For $\Delta \ll 1/\alpha$, mean reversion is not significant and $m(2,\Delta) \propto \Delta^{2\,H}$.

RFSV vs FSV

- We can compute $m(2,\Delta)$ explicitly in both the FSV and RFSV models.
- The smallest possible value of H in FSV is $H = 1/2$. One empirical estimate in the literature says that $H \approx 0.53$ some time in 2008.

- Let's see how the theoretical estimates of $m(2,\Delta)$ compare with data.

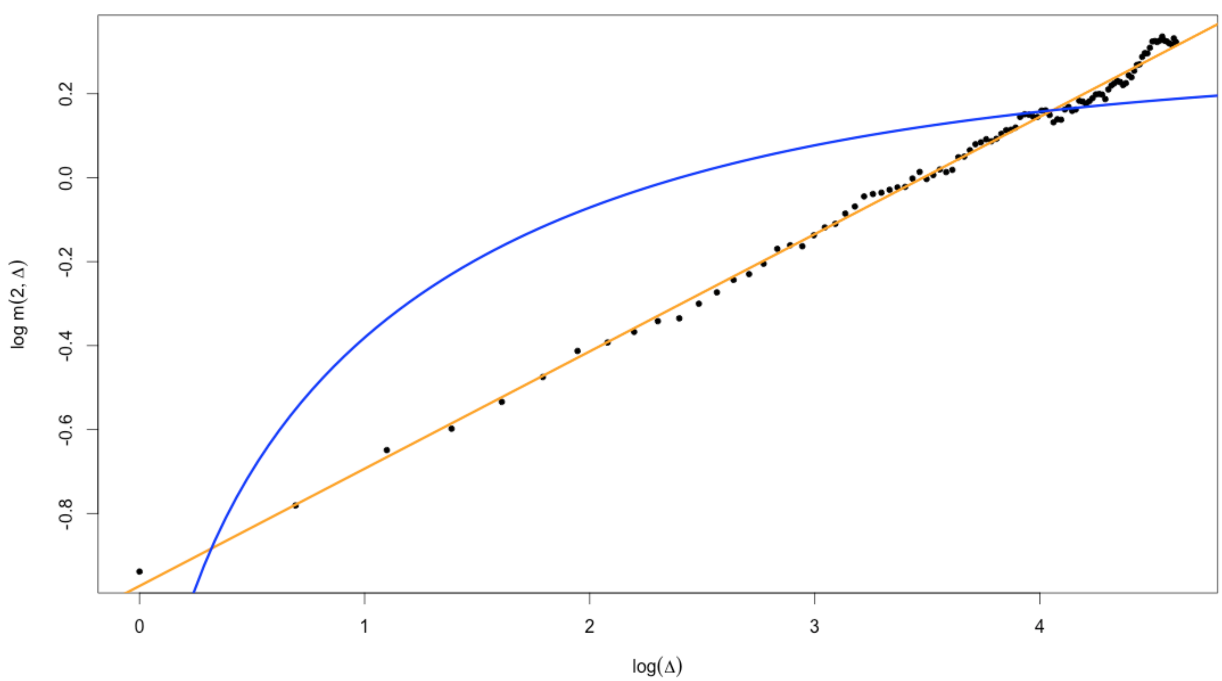


Figure 8: Black points are empirical estimates of $m(2,\Delta)$; the blue line is the FSV model with $\alpha = 0.5$ and $H = 0.53$; the orange line is the RFSV model with $\alpha = 0$ and $H = 0.14$.

Does simulated RSFV data look real?

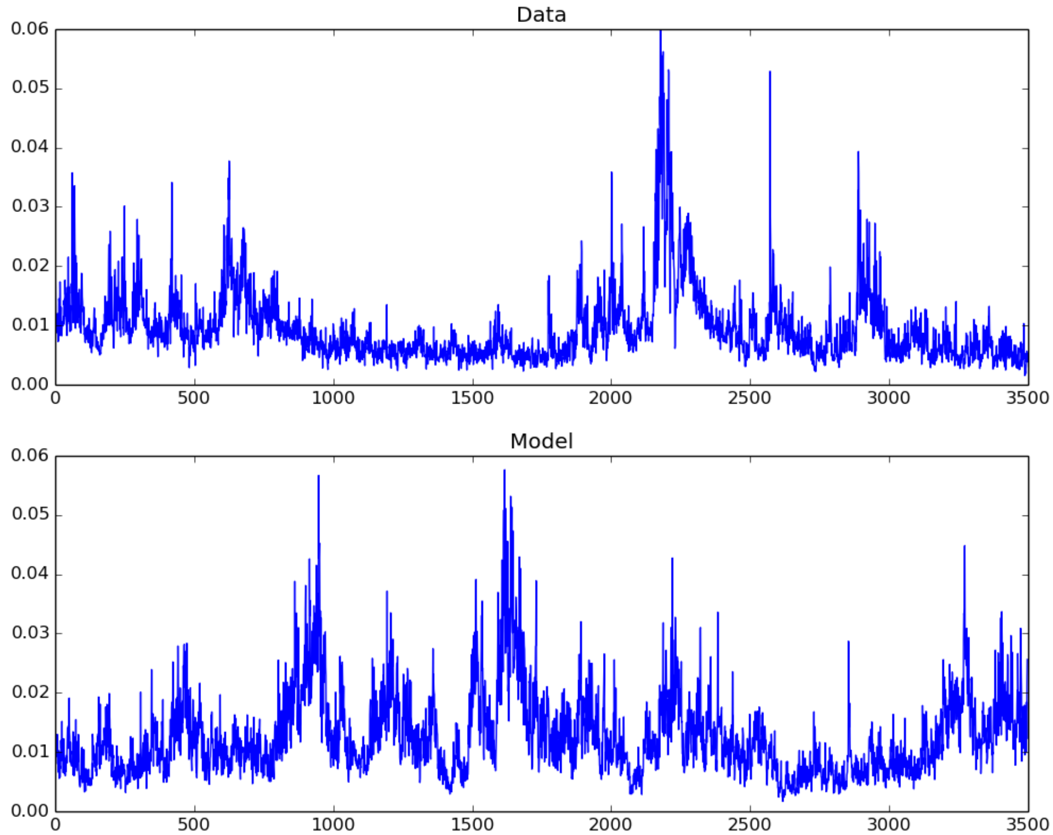


Figure 9: Volatility of SPX (above) and of the RFSV model (below).

Remarks on the comparison

- In respect of roughness, the simulated and actual graphs look very alike.
- Persistent periods of high volatility alternate with low volatility periods.
- $H \sim 0.1$ generates very rough looking sample paths (compared with $H = 1/2$ for Brownian motion).
- Hence *rough volatility*.

- On closer inspection, we observe fractal-type behavior.
- The graph of volatility over a small time period looks like the same graph over a much longer time period.
- This feature of volatility has been investigated both empirically and theoretically in, for example, [Bacry and Muzy][2] .

- In particular, their Multifractal Random Walk (MRW) is related to a limiting case of the RSFV model as $H \to 0$.

Applications

- What is this rough volatility model good for?
- If we could change measure from $\mathbb{P}$ to $\mathbb{Q}$, we would be able to price options.
  - We will explore this in Lecture 2.
- Another obvious application is to volatility forecasting.

Forcasting fBm

- In the RFSV model (1), $\log \sigma_t \approx \nu\, W^H_t + C$ for some constant C.
- [Nuzman and Poor][10] show that $W^H_{t+\Delta}$ is conditionally Gaussian with conditional expectation

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

and conditional variance

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

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 $W^H_{t+\Delta}$ against the W^H_s with $s < t$.
- 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 $0 < H < \frac{1}{2}$,

$$\int_0^\infty \beta(u,\Delta)\,|t-u|^{2H}\,du = (t+\Delta)^{2H}.$$

  - In particular,

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

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

$$\mathbb{E}\left[W^H_s\,\left(W^H_{t+\Delta} - \int_{-\infty}^t \beta(t-u,\Delta)\,W^H_u\,du\right)\right] = 0$$

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

where

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

Discretization of the forecast formula

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

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

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

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

Inspired by [Bennedsen, Lunde and Pakkanen][4], we approximate the first term in the sum more accurately as follows.

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

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

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

Implement variance forecast in R

In [37]:
```
# Find all of the dates
dateIndex <- substr(as.character(index(spx.rk)),1,10) # Create index of date:

cTilde <- function(h){gamma(3/2-h)/(gamma(h+1/2)*gamma(2-2*h))} # Factor bec:

# 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 :
    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 o:
    pick <- which(!is.na(ldata))
    norm <- sum(cf[pick])
    fcst <- cf[pick]%*%ldata[rev(pick)]/norm # Most recent dates get the hig:
    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 [38]:
```
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(rvdata=
    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 [39]:
```
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 [40]:
```
c(mean(vol.actual-vol.predict),sd(vol.actual-vol.predict))
```

-0.000416216523888338 0.0030816071726882

In [41]:
```
plot(vol.predict,vol.actual,col="blue",pch=20, ylab="Actual vol.", xlab="Pre(
abline(coef=c(0,1),col="red")
```

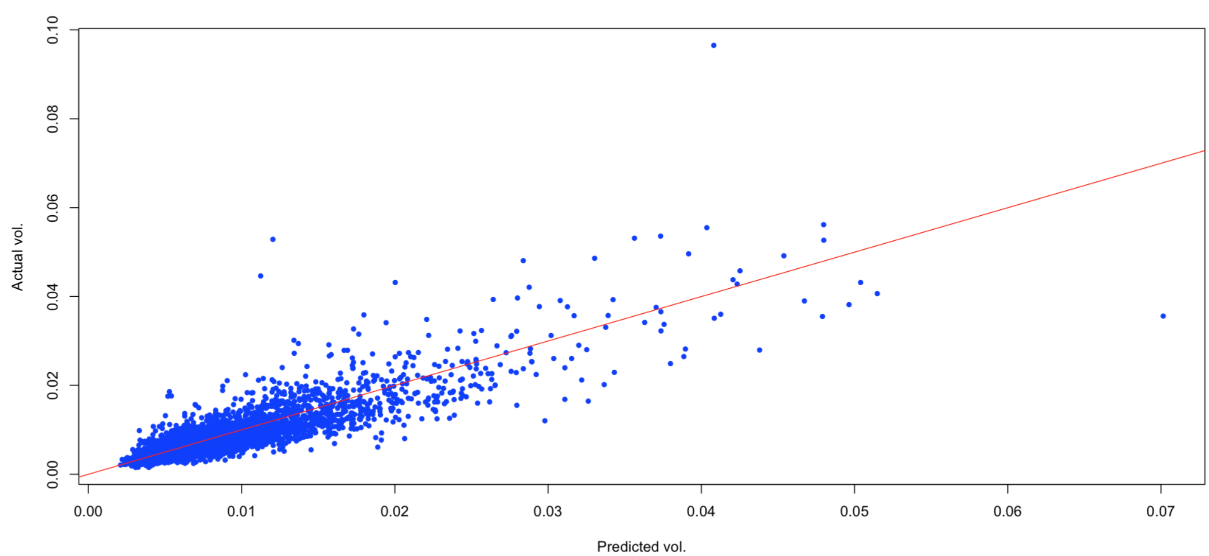


Figure 15: Actual vols vs predicted vols.

Which point is the outlier?

In [42]:
```
rv.actual[which(as.numeric(vol.actual)>.09)]
```

```
             [,1]
2008-10-10 "0.00931287184862693"
```

In [43]:
```
rv.predict[which(as.numeric(vol.predict)>.06)]
```

2008-10-10: 0.00491980615904601

Superimpose actual and predicted vols

In [44]:
```
plot(vol.actual, col="blue",type="l")
lines(vol.predict, col="red",type="l")
```

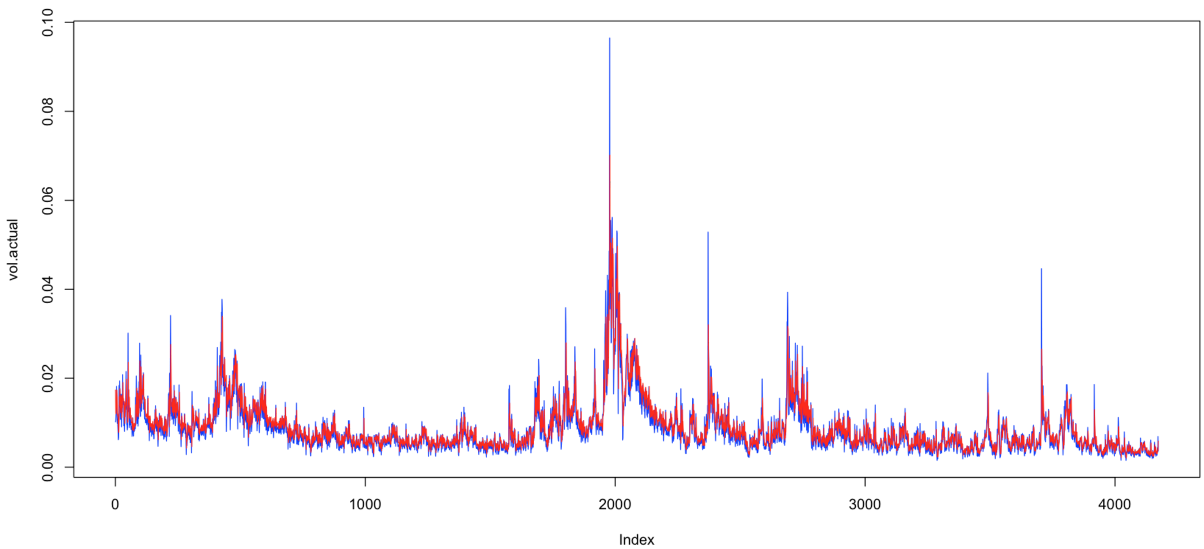
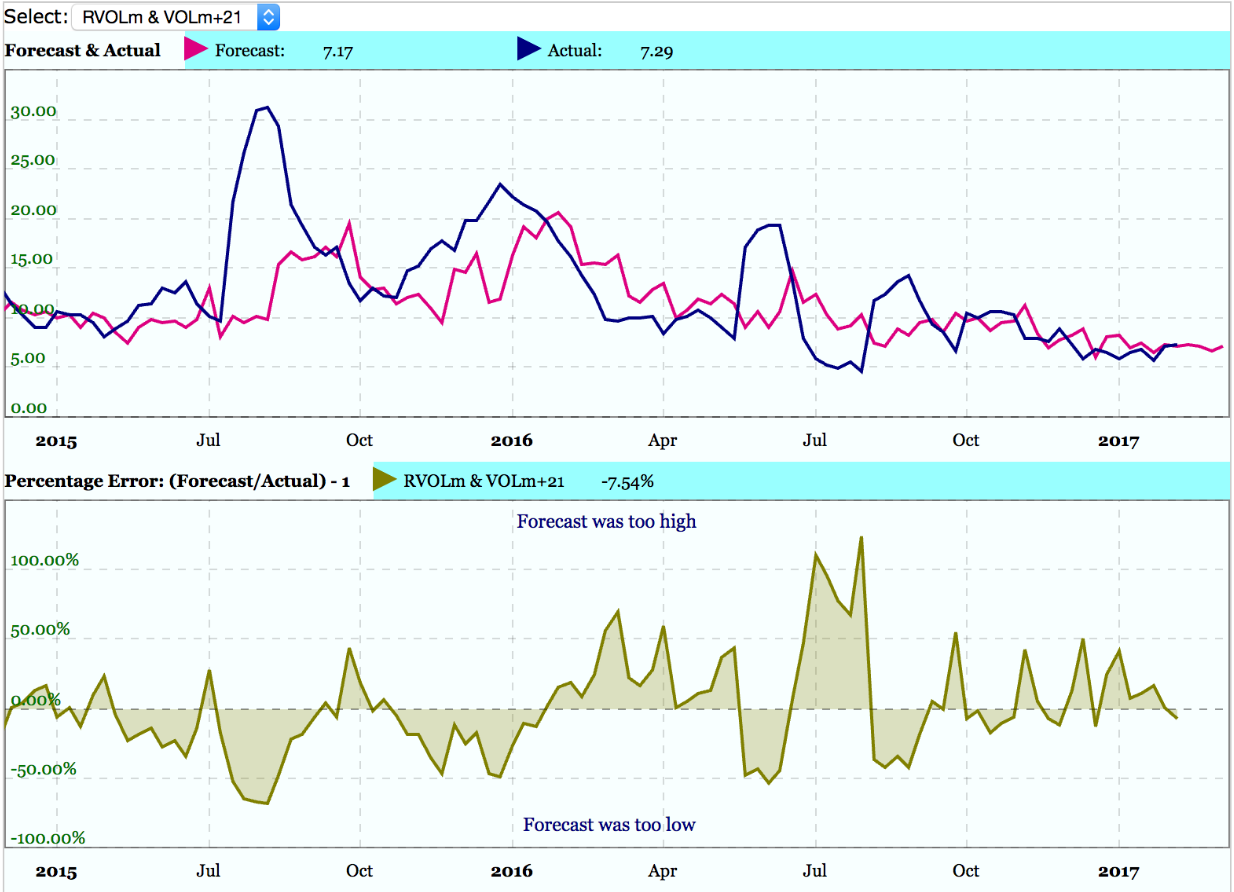


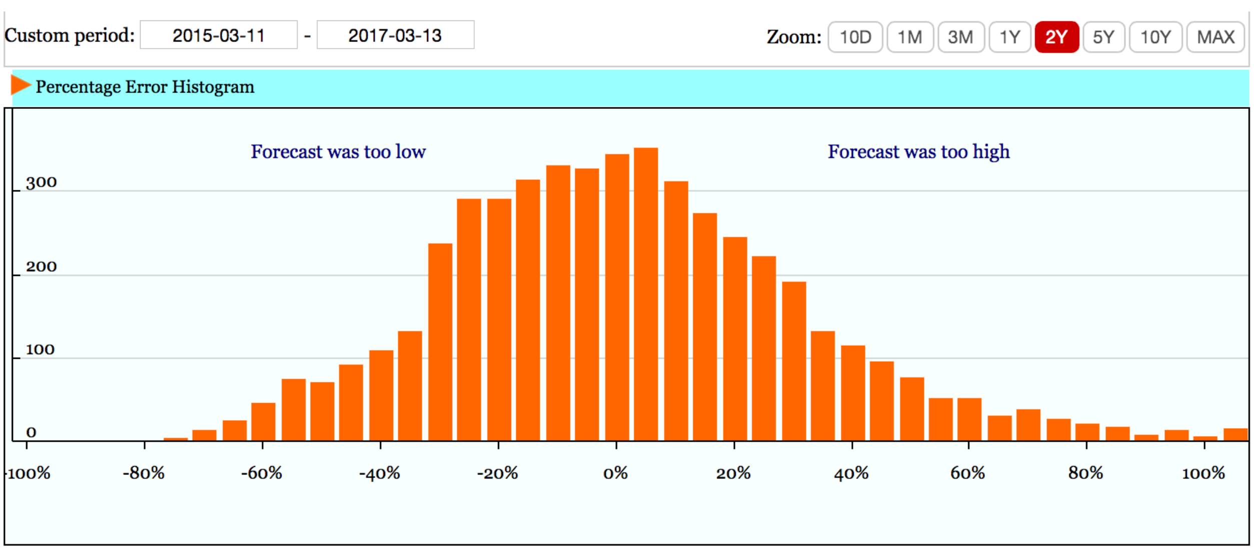
Figure 16: Actual volatilities in blue; predicted vols in red. Note that volatilities are in daily terms.

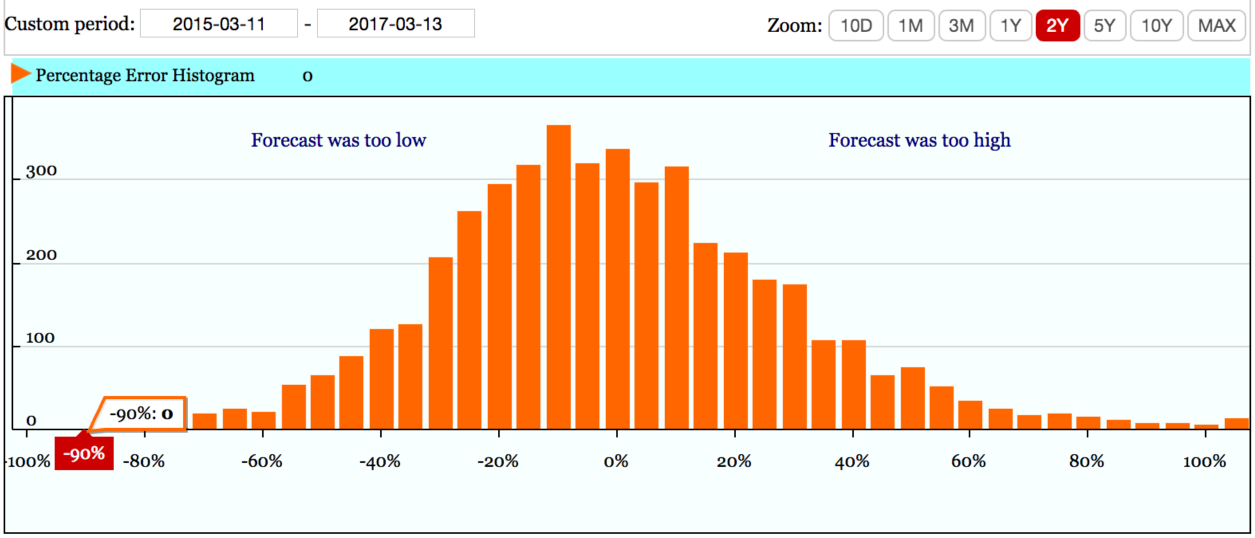
VolX

- 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


Select: RVOLm & VOLm+21
Forecast & Actual
Forecast: 7.17
Actual: 7.29
30.00
25.00
20.00
15.00
10.00
5.00
0.00
2015
Jul
Oct
2016
Apr
Jul
Oct
2017
Percentage Error: (Forecast/Actual) - 1
RVOLm & VOLm+21 -7.54%
Forecast was too high
100.00%
50.00%
0.00%
-50.00%
-100.00%
Forecast was too low
2015
Jul
Oct
2016
Apr
Jul
Oct
2017


Custom period: 2015-03-11 - 2017-03-13
Zoom: 10D 1M 3M 1Y 2Y 5Y 10Y MAX
Percentage Error Histogram
Forecast was too low
Forecast was too high
300
200
100
0
-100%
-80%
-60%
-40%
-20%
0%
20%
40%
60%
80%
100%



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] 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 [45]:
```
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 [46]:
```
var.log.err <- function(del){
    var(log.vol.err(del))
}
```

In [47]:
```
var.log.err(10)
```

0.138051752140461

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

In [54]:
```
del <- 1:100
var.log.err.del <- sapply(del,var.log.err)
```

In [55]:
```
save(var.log.err.del ,file="varerr.rData")
load(file="varerr.rData")
```

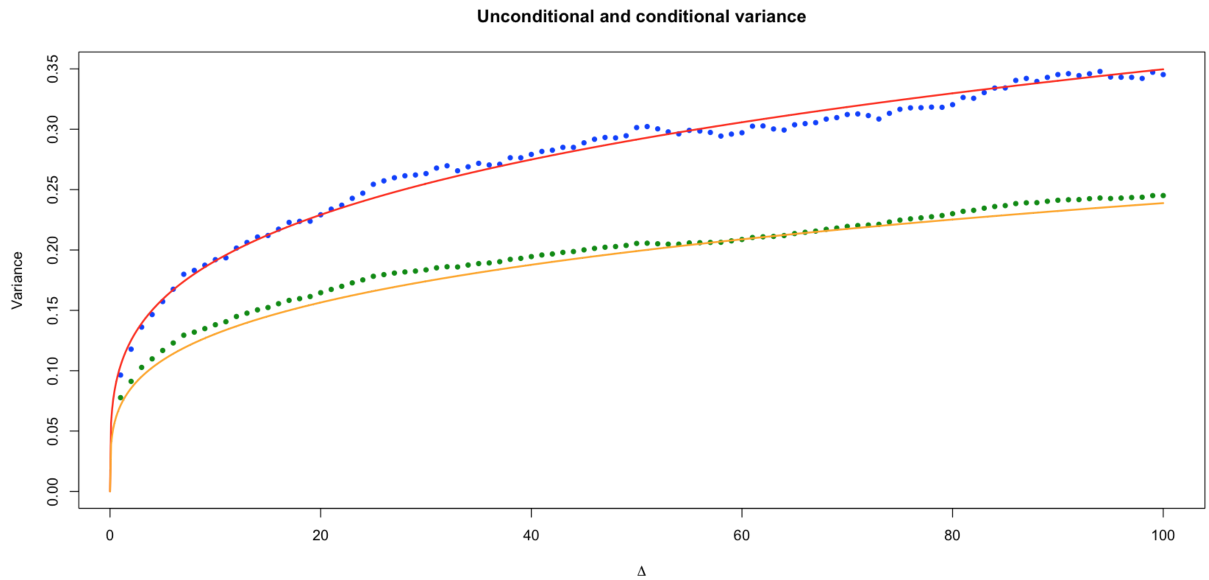
Plot of conditional and unconditional variance

- 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 `var.log.err`(Δ).

In [56]:
```
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="re
points(del,var.log.err.del,col="green4",pch=20)
curve(cTilde(h.spx.regression)* nu.spx.regression^2*x^(2*h.spx.regression),f
      add=T,col="orange",lwd=2,n=1000)
```



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

- The forward variance curve $\xi_t(u)$ is defined as
$$\xi_t(u) = \mathbb{E}^{\mathbb{Q}}\left[v_u \middle| \mathcal{F}_t\right], \qquad u \geq t.$$
- The forecasting formula allows us to estimate $\mathbb{E}^{\mathbb{P}}\left[v_u \middle| \mathcal{F}_t\right]$.
- In [Bayer, Friz and Gatheral][3], we demonstrate that it is possible to forecast the variance swap curve giving two dramatic examples.
  - The Lehman weekend.
  - The Flash Crash.
- We thus have a simple model that is both consistent with historical volatility dynamics and able to forecast option prices.

Summary

- We uncovered a remarkable monofractal scaling relationship in historical volatility.
  - Conventional long memory models are inconsistent with this scaling relationship.
  - Prior long memory estimates appear to be spurious.
- The Hurst exponent H varies over time.
  - Peaks typically correspond to periods of market stress.

- This leads to a natural non-Markovian stochastic volatility model under $\mathbb{P}$.
- One application is to forecasting.
  - The resulting forecast is both simpler than and superior to other available volatility forecasts.

References

1. ^ Torben G Andersen, Tim Bollerslev, Francis X Diebold, and Heiko Ebens, The distribution of realized stock return volatility, *Journal of Financial Economics* **61**(1) 43-76 (2001).
2. ^ Emmanuel Bacry and Jean-François Muzy, Log-infinitely divisible multifractal processes, *Communications in Mathematical Physics* **236**(3) 449-475 (2003).
3. ^ Christian Bayer, Peter Friz and Jim Gatheral, Pricing under rough volatility, *Quantitative Finance* **16**(6) 887-904 (2016).
4. ^ Mikkel Bennedsen, Asger Lunde, and Mikko S. Pakkanen, Decoupling, (2016).
5. ^ Fabienne Comte and Eric Renault, Long memory in continuous-time stochastic volatility models, *Mathematical Finance* **8** 29-323(1998).
6. ^ Omar El Euch and Mathieu Rosenbaum, The characteristic function of rough Heston models, available at https://arxiv.org/abs/1609.02108 (https://arxiv.org/abs/1609.02108), (2016).
7. ^ Jim Gatheral, Thibault Jaisson and Mathieu Rosenbaum, Volatility is rough, available at http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2509457 (http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2509457), (2014).
8. ^ Jim Gatheral and Roel Oomen, Zero-intelligence realized variance estimation, *Finance and Stochastics* **14**(2) 249-283 (2010).
9. ^ Thibault Jaisson and Mathieu Rosenbaum, Rough fractional diffusions as scaling limits of nearly unstable heavy tailed Hawkes processes, available at https://arxiv.org/pdf/1504.03100.pdf (https://arxiv.org/pdf/1504.03100.pdf), (2015).
10. ^ 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).