***Subsampling***

Many of the estimators can be subsampled.

RV, Bipower, RQV\*, MinRV\*, MedRV\*, TSRV\*, MSRV\*, Optimal Sampling RV, RSV, Threshold RV, Realized Range\*

This leaves Kernel and Realized QMLE as the estimators which cannot be subsampled, or maybe should not be subsampled.

\* These estimators are all use all blocks. They can still be subsampled by shifting the start of the first return, which produces a different set of blocks.

I would recommend that we implement subsampling using the following rule so that the we never subsample more frequently than 1/second (these are millisecond data, do you could subsample 1000 times/second), but never more than 10 subsamples. Let nSeconds by the number of seconds in a day and let m be the number of returns. Then the average duration is nSeconds/m, and so we could set subsampled = min(10,ceil(nSeconds/m)). So for example, when using 5 minute data on US equities, nSeconds = 23400 and m = 78, so we would use 10. If using 5 second sampling on US equities, nSeconds = 23400 and m = 4690, so subsamples = 5.

***Quantile Realized Variance***

I have consolidated all of the emails about QRV and subsampling into one reply.

The basic usage of QRV, in the default case should now be

blockSize = 20;

quantiles = [13 15 18];

rq = realized\_quantile\_variance(price,time,timeType,samplingType,samplingInterval, **quantiles/blockSize,blockSize,false,true**)

**Bold to indicate the important bits.**

* It is easier to think about which returns to use, so that is why I parameterized blockSize and quantiles
* The original paper did not use the symmetric version, so it is false
* There is no good reason not to set “overlap” to true (the 4th bolded).

You can get minRV as

blockSize = 2;

quantiles = ;

rq = realized\_quantile\_variance(price,time,timeType,samplingType,samplingInterval, **quantiles/blockSize,blockSize,true,**true)

Min and MedRV can be computed as

MinRV

realized\_quantile\_variance(price,time,timeType,samplingType,samplingInterval, **1/2,2,true,**true)

MedRV

realized\_quantile\_variance(price,time,timeType,samplingType,samplingInterval, **2/3,3,true,**true)

I have also included a new function realized\_min\_med\_variance which only compute Min and MedRV. These are virtually identical; the only difference is that min\_med\_variance uses analytical constants, while the QRV version uses constants computed using numerical integration. In practice estimates differ in the 4th decimal place, so it doesn’t make much difference.

***Subsapling***

QRV and Min/Med can be subsampled. In all cases we should use the “overlap” version of the estimators (realized\_min\_med\_var only does overlap), but subsampling can be used too. Whiel overlap uses all overlapping blocks of returns, subsampling shifts the start of the block and so ended up with a new sequence of blocks.

***Threshold Variance***

I have also written realized\_threshold\_variance which implements the Mancini-type threshold following the 2010 Corsi, et. al. article. The basic usage is

[rtv,rtsd,rtsss,rtsssd]=realized\_threshold\_variance(price,time,timeType,samplingType,samplingInterval,[],[],subsamples).

The two []’s are the threshold scale (1st) and the local variance estimate (2nd). The default scale is 3 and the local variance is computed from the data using a kernel weighted average. Returns are thresholded is r(t)>scale \* V(t) where V(t) is the loca variance.

It can also be subsampled like RV.

***Optimal Sampling RV***

I have changed the call to optimal sampling RV to allow the maximum frequency to be set in the call to the function.

[rv,rvDebiased,rvSS,rvDebiasedSS,diagnostics] = realized\_variance\_optimal\_sampling(price,time,timeType,samplingType,samplingInterval,subsamples,options)

In the usual case where all data is used, the standard call is

[rv,rvDebiased,rvSS,rvDebiasedSS]=realized\_variance\_optimal\_sampling(price,time,timeType,’BusinessTime’,1,subsamples)

I have been looking at the QRV code, and I have some questions.  Before we started talking about subsampling, I was implementing the asymmetric version of QRV with subsamples = 0 and oversamples = 0.  (I chose asymmetric because I think the Christensen et al paper uses it in the empirical application.)  But I just want to confirm that this implementation does not involves any built-in subsampling (in the way that the Christensen et al paper details); the blocks or returns they calculate quantiles over are adjacent and non-overlapping:

%[ Following code is extracted from around line 236 in MFEToolbox.zip\realized\realized\_quantile\_variance.m, from the function "realized\_quantile\_variance\_core"]

count=1;

initialBins = 1:samplesperbin:n;

bins = initialBins';

% [SOME LINES OMITTED]

for j=bins'

    binreturns = returns(j:j+samplesperbin-1);  % binreturns for each iteration are non-overlapping ~Lily

    binreturns = sort(binreturns)';

    rqindiv(count,:)=binreturns(lowerIndices).^2+binreturns(upperIndices).^2;

    count = count + 1;

end

So, now I'm trying to implement subsampled QRV, but I don't think the subsampling portion of the code works for the .m file that I have.  I've attached the version of the QRV code that I'm using, with questions and changes I made to make the program run marked with "[LILY]".  I know you mentioned that you were changing some of this code.

Also, I think the subsampling that the code intends to do is different from the subsampled QRV in the Christensen et al paper.  The paper version of the subsampled QRV uses the exact same sample of filtered price data as the non-subsampled version, but instead of taking 'non-overlapping contiguous' blocks of m returns, as in the non-subsampled QRV, they take "rolling blocks" of size m instead.  The subsampled QRV in the code seems to want to use different "subsamples" from the set of all available prices, which I guess is what a "subsampled" version of an estimator usually is.

 I am just wondering if we actually want the version in the code instead of the subsampled QRV from the paper.

I hope this made sense, and thanks in advance!

I think I overlooked something in your QRV code.  If I set oversamples = samplesperbin-1 and subsamples = 0, then the estimator becomes the "subsampled" QRV from the Christensen et al paper (Section 2.1), right?  Is this the version of QRV that you were referring to as having subsampling built in already?  So, should I only include this version of QRV and remove the previously implemented "block" QRV (Section 2.2)?

<http://public.econ.duke.edu/~get/browse/courses/201/spr11/DOWNLOADS/MicroStructure/cop_quantile_rv_joe_10.pdf>

Also, I now realize that  'oversamples' is the feature that allows the code to produce minRV and medRV as well. I coded the minRV and medRV myself because I didn't realize until now.  Just to check that I understand the QRV code:  to implement minRV, I would set samplesperbin=2, oversamples=1, quantile=(0.5+1)/2 = 0.75.

Sorry, I forgot something:  for minRV, I also need to use the "symmetric" version of QRV, along with the other things I wrote in the previous email:   samplesperbin=2, oversamples=1, quantile=(0.5+1)/2 = 0.75.

Sorry - I've been bombarding you guys with emails.  Now I have questions about subsampling TSRV & MSRV, and minRV & medRV.

I just want to describe the way I implemented the *not-subsampled* TSRV and MSRV to check that they are right:

Say, I have asset prices for SPX that are reported every 15 seconds.  For 5 minute TSRV, the estimator is calculated with a fast time scale of 5 minute sampling, and the slow-time scale is chosen optimally in the code to minimize asymptotic variance.  The slow time scale is the one that is "subsampled", and the fast time scale is used to bias-correct.  However, only 1 out of approximately 5\*60/15 - 1 = 19 subsamples of filtered 5-minute returns, i.e. ~5% of the price data, is used to calculate this TSRV estimate.  So, I could calculate 19 more estimates of TSRV using the other 18 subsamples of data, and then average them.  But do we want to include this subsampled TSRV?

Or, maybe I did this completely wrong, and 5 minute TSRV should have tick-by-tick data as the fast time scale and 5 minutes as the slow time scale, in which case, there would be no further subsampling possible.

And, for minRV and medRV, I don't quite understand how subsampling is built in.  It seems that if I calculate 1 minute minRV (or medRV) for an asset that has prices reported every 15 seconds, I am only using one sample of filtered 1 minute prices to calculate 1 minute returns, i.e. 1/3 of the total available price data, meaning that I could calculate 2 more minRV estimates on the other two subsamples and average them.  Is this right?  Or am I misinterpreting how the regular (not subsampled) minRV is calculated?

Hi Lily,

Sorry for the slow reply – I got back from Sydney just last night.

Kevin might have replied to your email already, but just in case here are my thoughts:

I just want to describe the way I implemented the *not-subsampled* TSRV and MSRV to check that they are right:

Say, I have asset prices for SPX that are reported every 15 seconds.  For 5 minute TSRV, the estimator is calculated with a fast time scale of 5 minute sampling, and the slow-time scale is chosen optimally in the code to minimize asymptotic variance.  The slow time scale is the one that is "subsampled", and the fast time scale is used to bias-correct.  However, only 1 out of approximately 5\*60/15 - 1 = 19 subsamples of filtered 5-minute returns, i.e. ~5% of the price data, is used to calculate this TSRV estimate.

* This sounds correct to me. When we choose the sampling freq for TSRV we are choosing the highest frequency, and letting the formula determine the slower freq.

So, I could calculate 19 more estimates of TSRV using the other 18 subsamples of data, and then average them.  But do we want to include this subsampled TSRV?

* Right. This is what we would call “subsampled TSRV”. I think we should include this estimator.

Or, maybe I did this completely wrong, and 5 minute TSRV should have tick-by-tick data as the fast time scale and 5 minutes as the slow time scale, in which case, there would be no further subsampling possible.

* I think you did it right already. What you describe here is the one-tick TSRV, and you are right that it cannot be subsampled (because no higher freq data is available).

And, for minRV and medRV, I don't quite understand how subsampling is built in.  It seems that if I calculate 1 minute minRV (or medRV) for an asset that has prices reported every 15 seconds, I am only using one sample of filtered 1 minute prices to calculate 1 minute returns, i.e. 1/3 of the total available price data, meaning that I could calculate 2 more minRV estimates on the other two subsamples and average them.  Is this right?

* Your thinking sounds right to me here too. Kevin – does this sound right to you?

-Andrew.