QBO Anomaly FFT Filtering

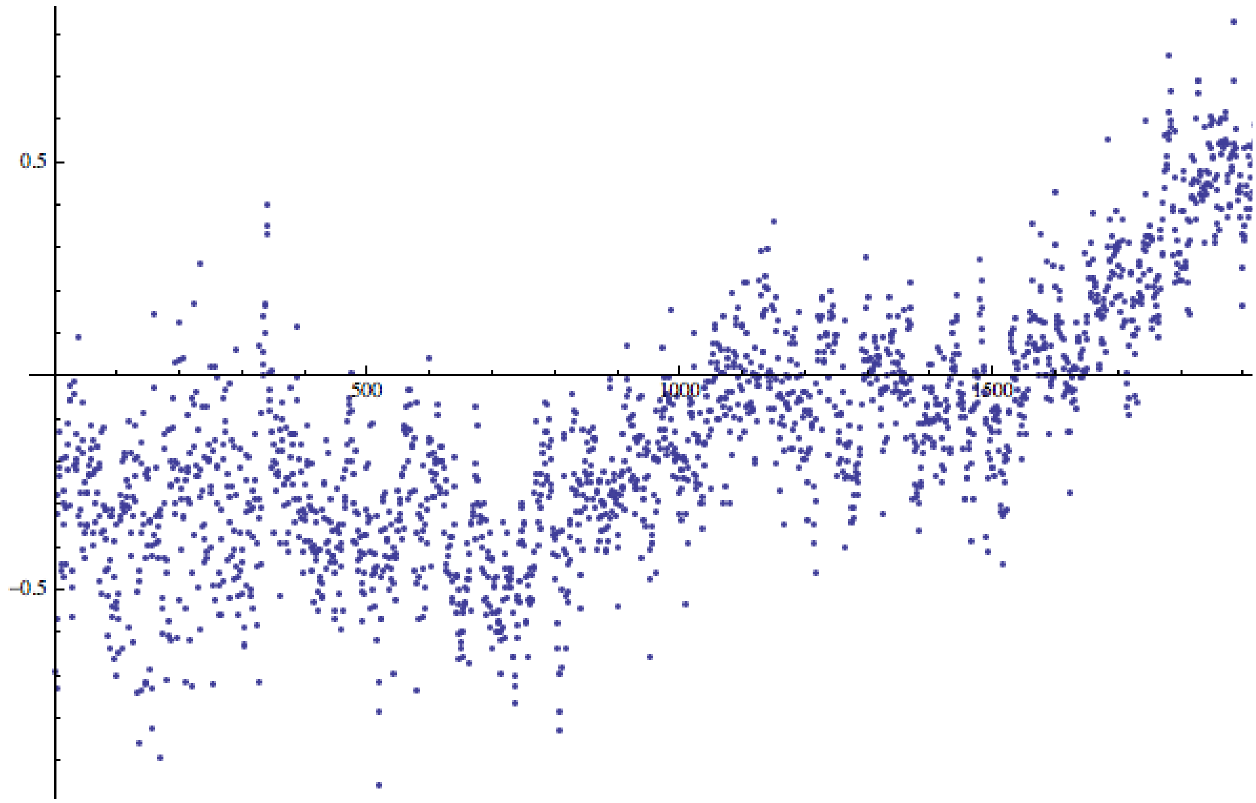
dara@lossofgenerality.com

Temperature Anomaly Data

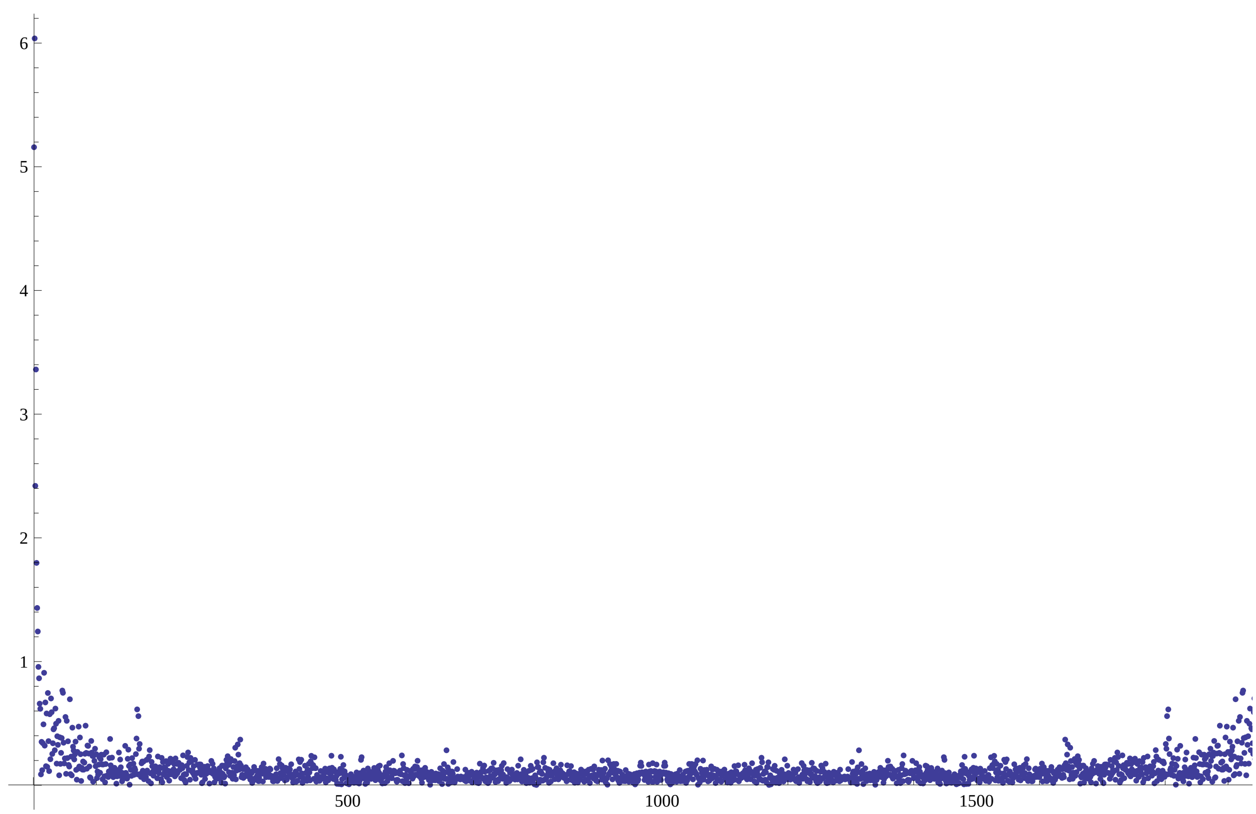
Source: http://www.cru.uea.ac.uk/cru/data/temperature/HadCRUT4-gl.dat

```
1850 -0.690 -0.279 -0.728 -0.565 -0.322 -0.215 -0.130 -0.234 -0.439 -0.455 -0.191 -0.265 -0.374
 1850     22     20     18     19     18     20     22     22     23     23     24     25
 1851 -0.295 -0.346 -0.468 -0.441 -0.304 -0.189 -0.216 -0.159 -0.111 -0.054 -0.021 -0.056 -0.219
 1851     23     22     20     21     20     20     21     23     18     20     18     20
 1852 -0.301 -0.455 -0.492 -0.561 -0.206 -0.040 -0.013 -0.205 -0.132 -0.218 -0.193  0.092 -0.223
 1852     23     22     22     23     23     23     23     23     21     21     22     25
 1853 -0.169 -0.324 -0.312 -0.346 -0.272 -0.176 -0.060 -0.148 -0.404 -0.363 -0.247 -0.424 -0.268
 1853     25     26     26     24     25     27     26     29     28     29     26     27
 1854 -0.352 -0.274 -0.271 -0.346 -0.228 -0.209 -0.230 -0.167 -0.114 -0.193 -0.366 -0.229 -0.243

 ...

 2010  0.545  0.558  0.668  0.676  0.582  0.582  0.607  0.531  0.423  0.484  0.576  0.328  0.547
 2010     86     86     87     86     85     86     86     87     86     86     86     87
 2011  0.314  0.314  0.409  0.459  0.366  0.472  0.496  0.478  0.438  0.437  0.325  0.375  0.406
 2011     86     85     85     84     84     83     83     84     84     85     84     84
 2012  0.300  0.276  0.353  0.535  0.543  0.518  0.487  0.537  0.534  0.526  0.518  0.268  0.448
 2012     84     84     82     80     79     79     79     81     81     80     82     81
 2013  0.450  0.479  0.405  0.427  0.498  0.457  0.520  0.528  0.532  0.478  0.601  0.486  0.487
 2013     81     80     80     77     77     76     77     78     77     77     78     79
```

Plot

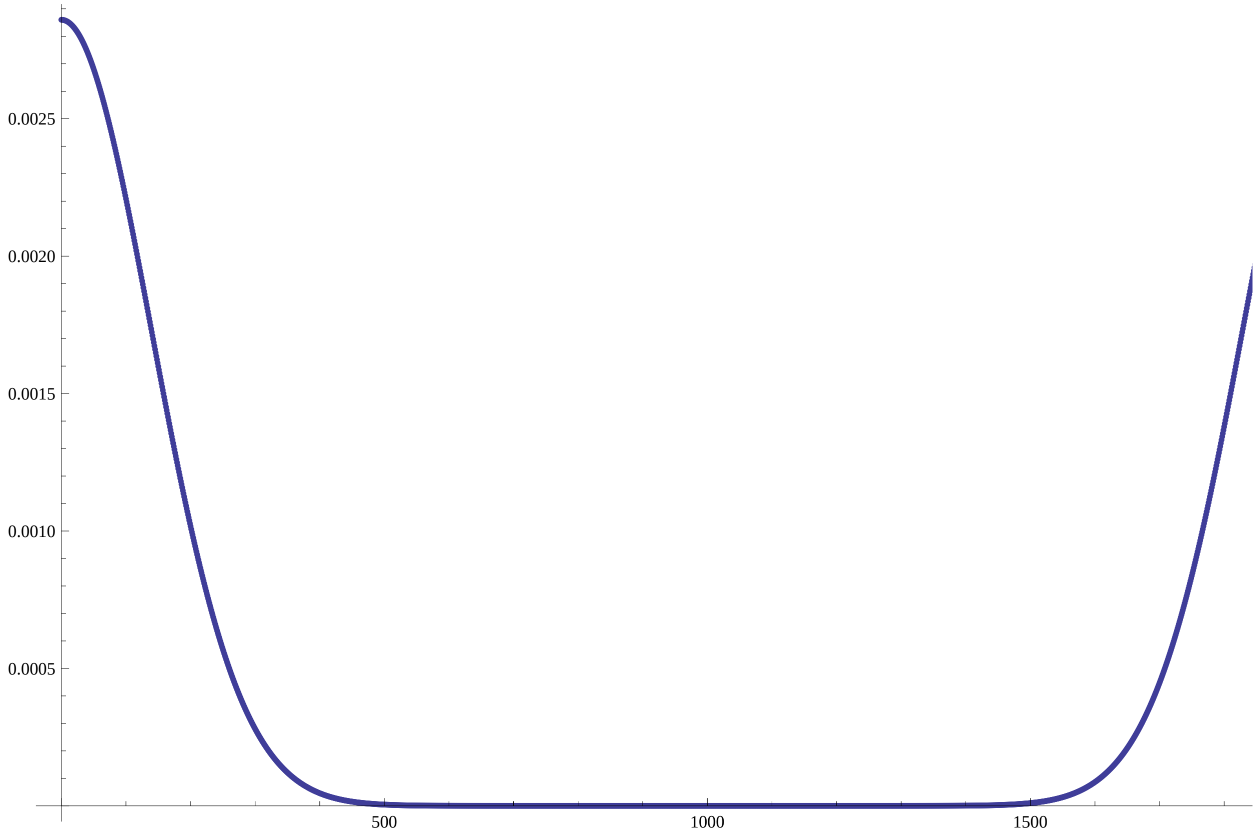


Abs[FFT[QBO]]



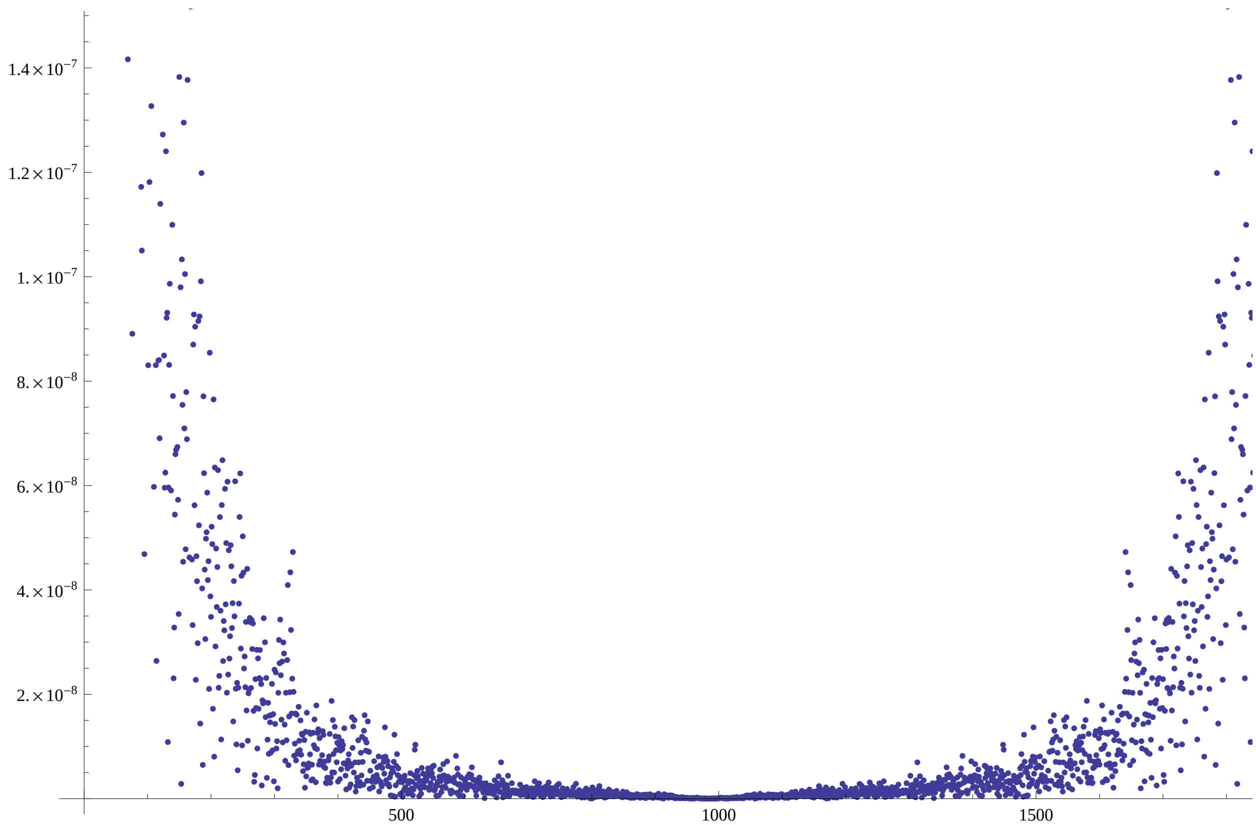
Gaussian Convolution Filter

Cut a Gaussian bell in half, mirror image one against y-axis, and glue them to each other, normalize the curve by the sum of all y values, as follows:



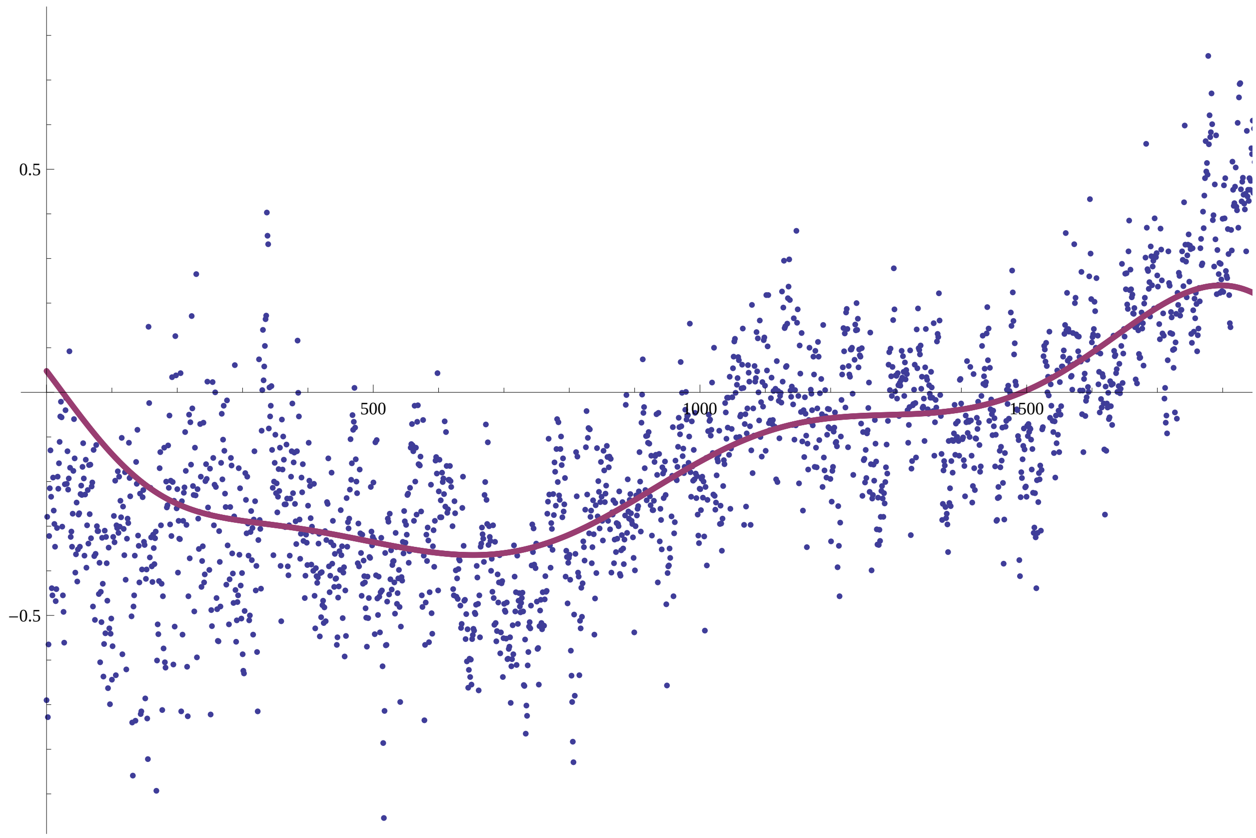
Array multiply FFT of original data FFT[QBO] by FFT of the above Gaussian (FFT[g] cut up curve and square root of the length of original data, you get this amplitude map:

FFT[QBO] × FFT[*g*]



Note: *Array multiply multiplies the corresponding members of two arrays and make a new array of same dimension.*

Do an inverse FFT to the latter, the RED curve is Trend:



Or convolve the original data and the Gaussian cuts :

Example

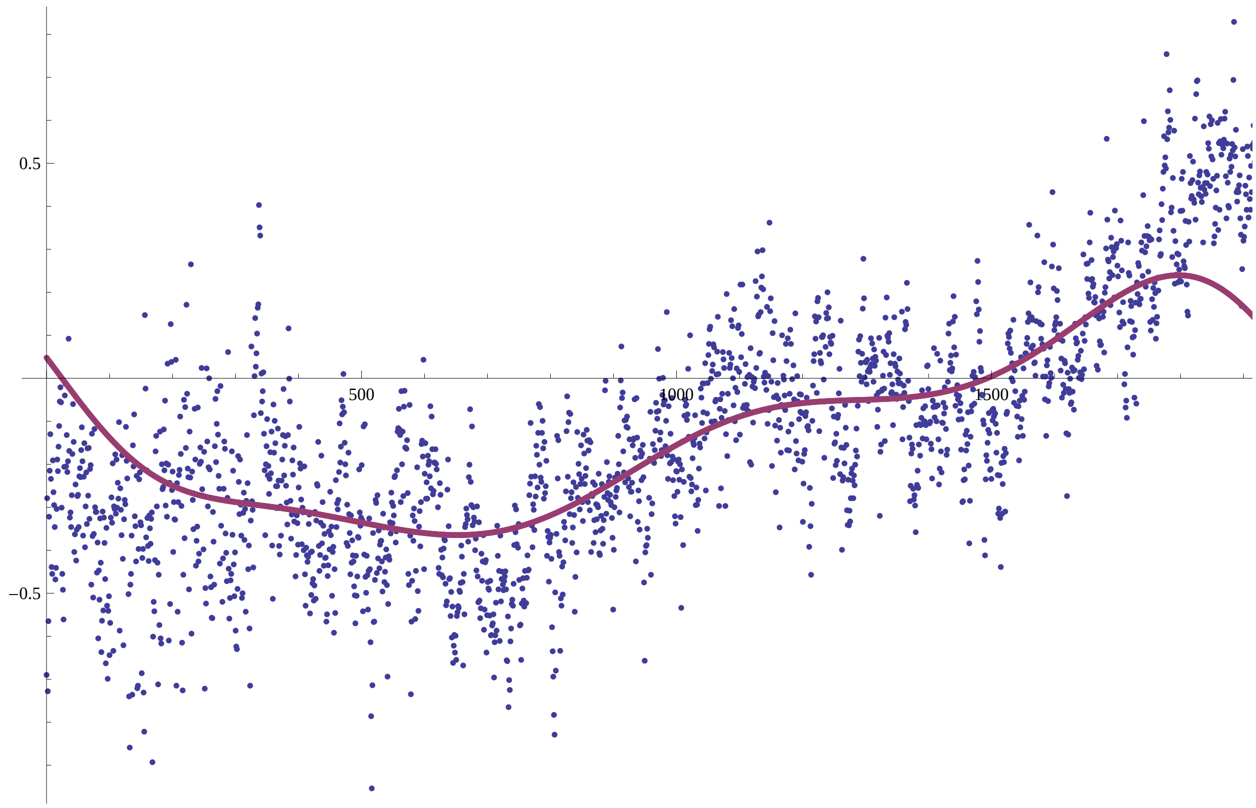
$\{x, y, z\}$ original data

$\{u, v, w\}$ Gaussian cuts

$\{u x + w y + v z, v x + u y + w z, w x + v y + u z\} = \{x, y, z\} * \{u, v, w\}$ Convolution

You get the same by Convolution Theorem:

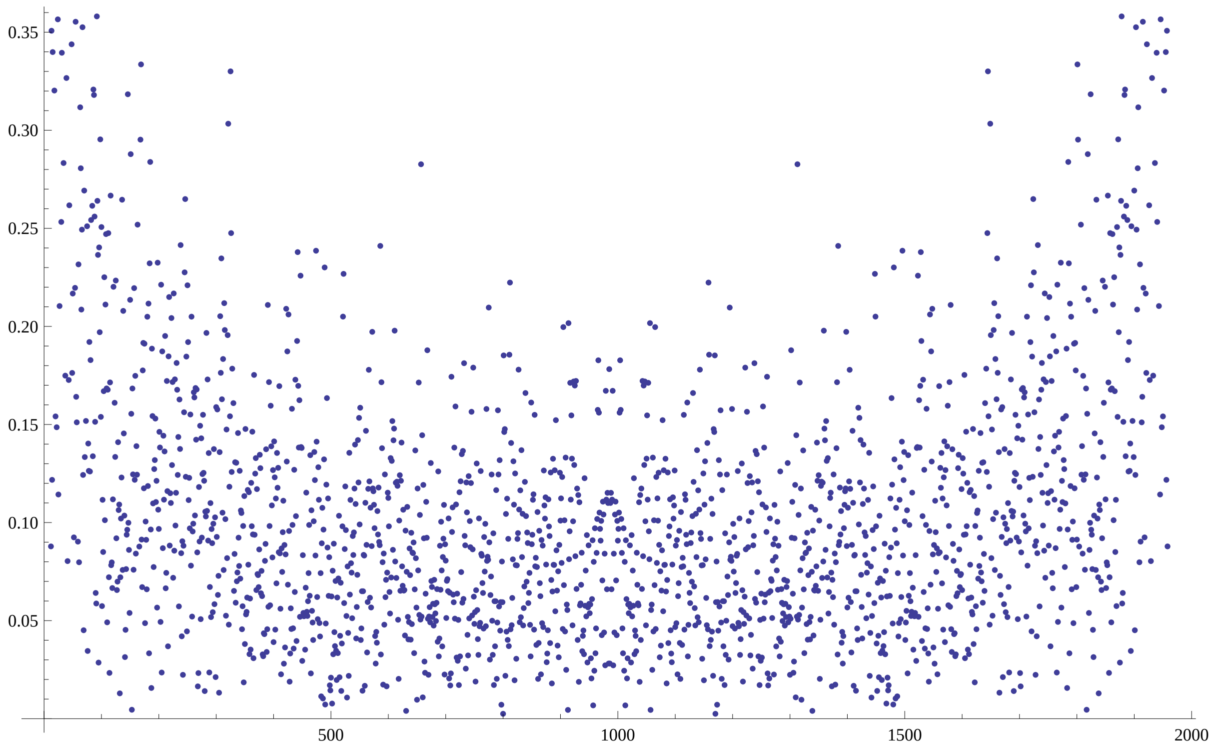
QBO $*$ ***g***



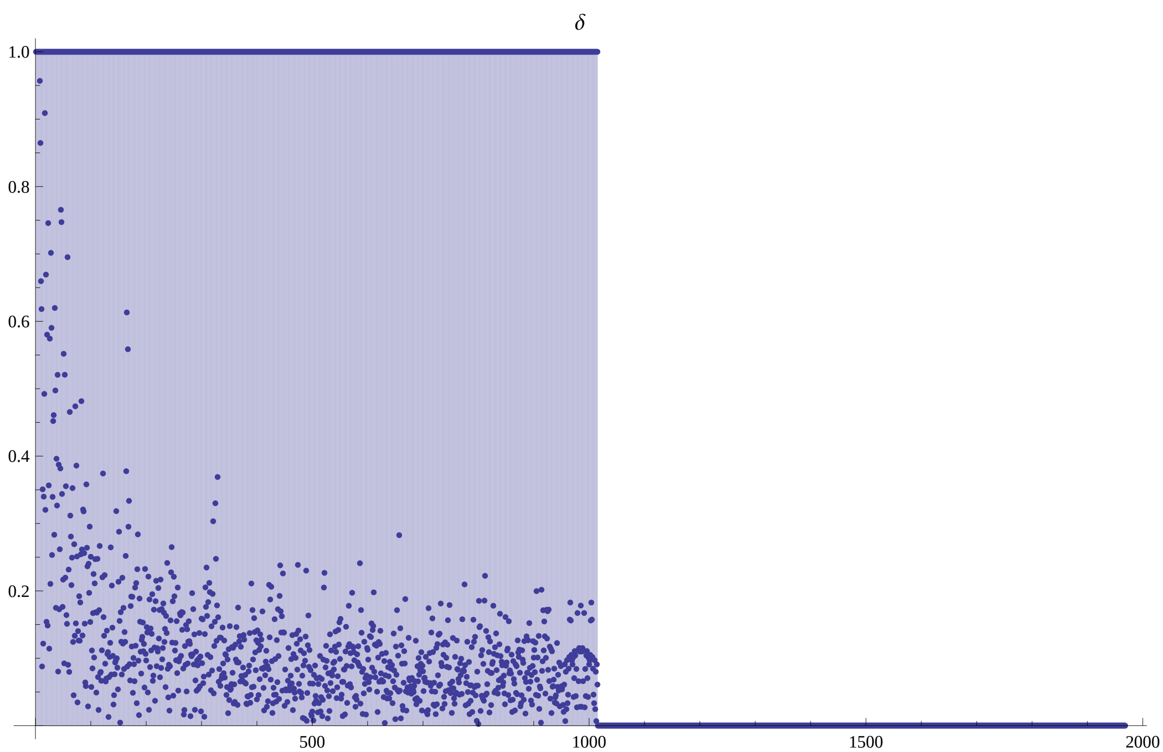
As you can see the convolution technique works but the denoised curve might be too smoothed and not much control on the output.

Denoising by Frequency Filtering

This is the frequency map of the QBO, we would like to cut out regions or individual frequencies, inverse FFT back to a smoothed new data, thus have complete control on the output:



Simple BandPass filter starting from 0 going to 1 increasing in frequency, thus filtering more and more noisy data, finally at 1 the original dataset is obtained loss-less:

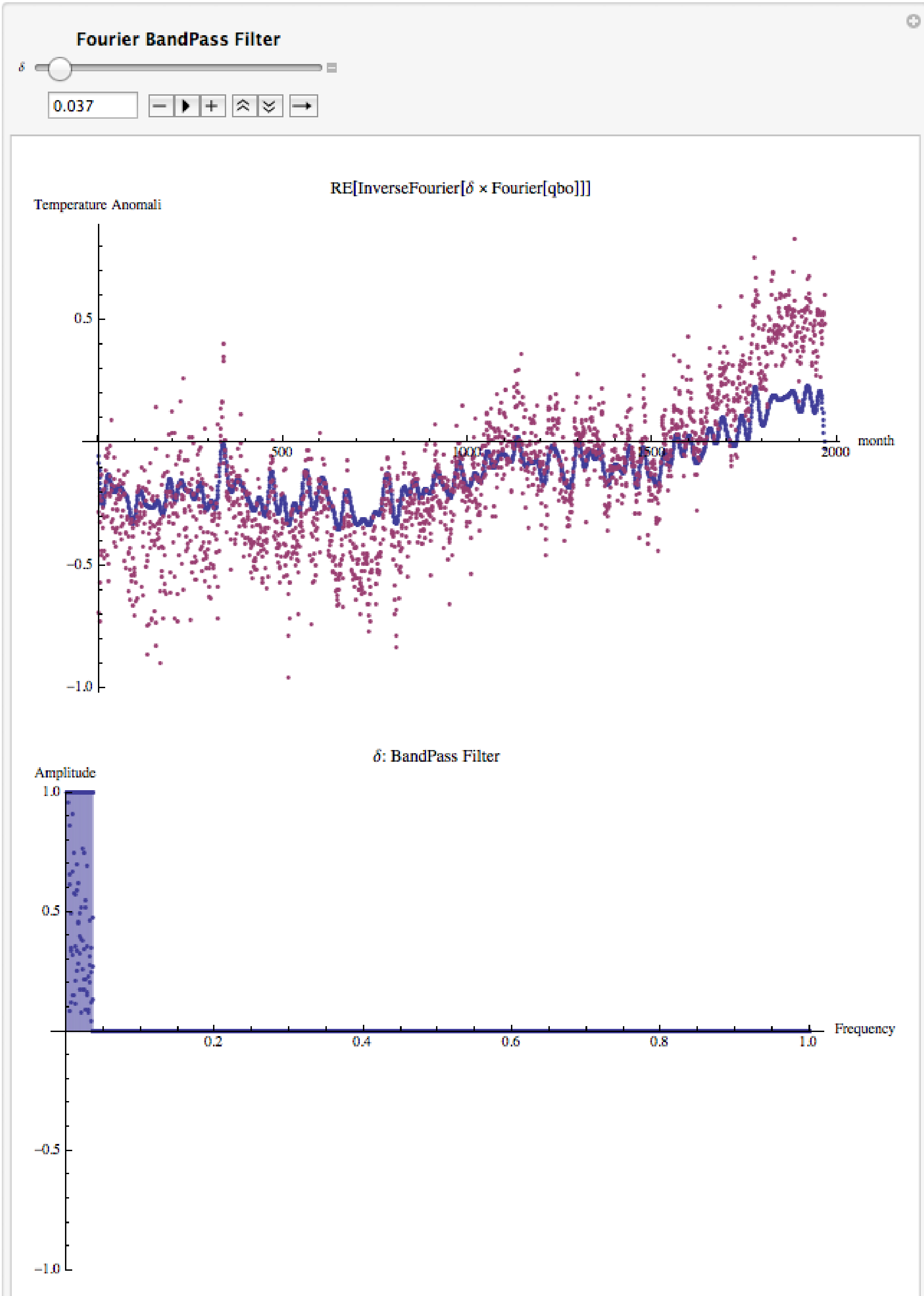


Early frequencies allow for a nice Trend for the data, but counting the peaks on this Trend actual periodicity could be estimated:

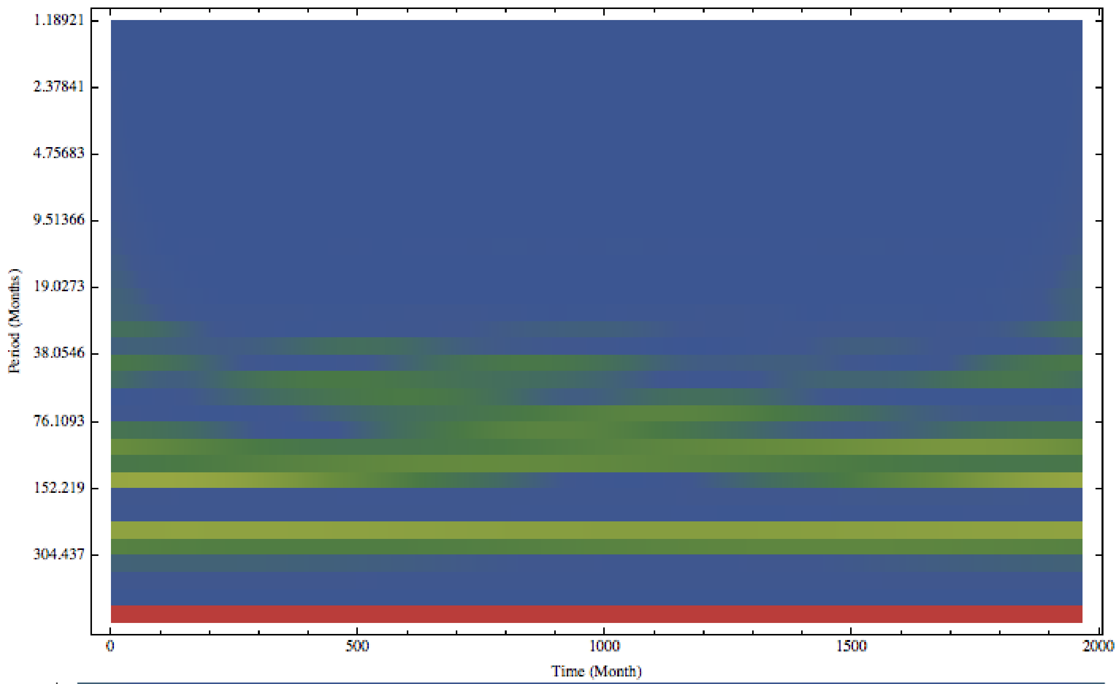
≈ 50 peaks

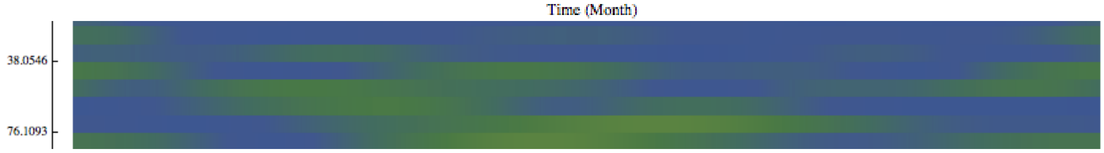
$\frac{2000}{50} = 40$ months of periodicty

Out[2446]=



Similar numbers were obtained by looking at the Wavelet Scalograms:


1.18921
2.37841
4.75683
9.51366
19.0273
38.0546
76.1093
152.219
304.437
Period (Months)
0
500
1000
1500
2000
Time (Month)


38.0546
76.1093