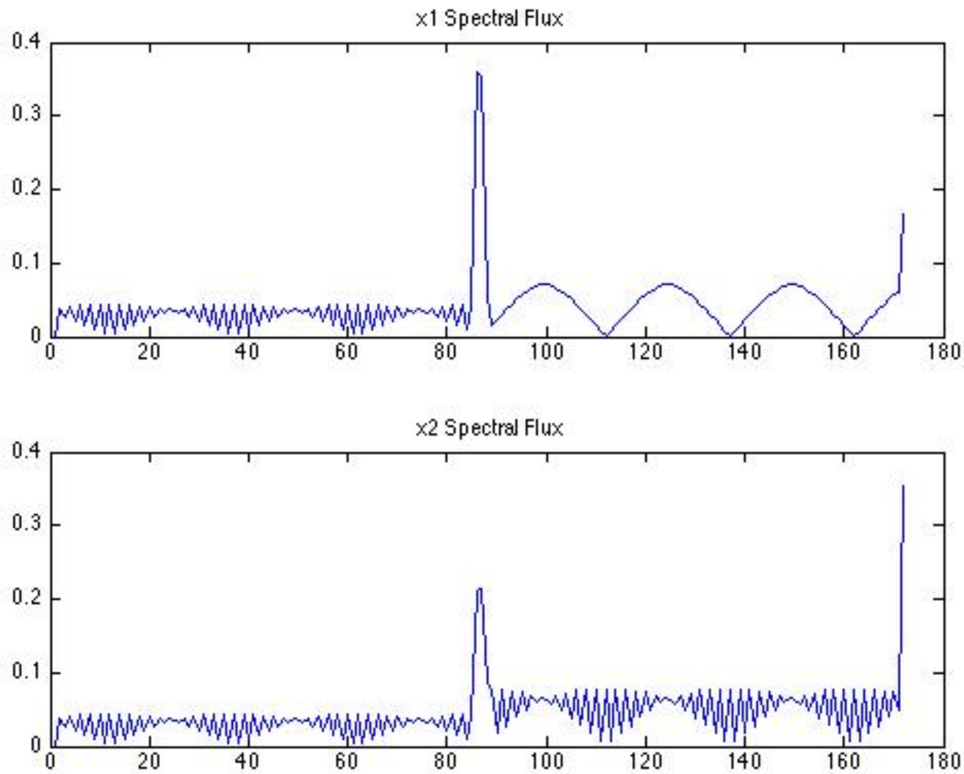
HW2 Report
Chris Laguna, Ying Zhan

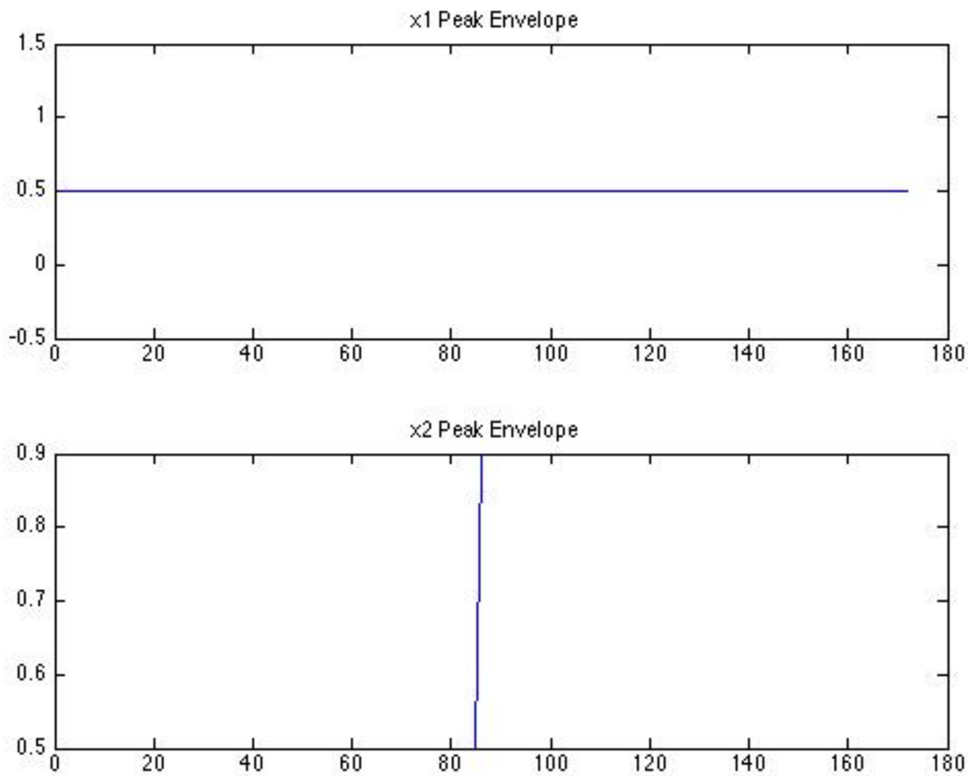
1.



windowSize = 1024, hopSize = 512

An onset occurs at 0.98 seconds (block 86) in x1, and 0.99 seconds (block 87) in x2. This function works for the test signals, because it accounts for changes in frequency as well as changes in amplitude.

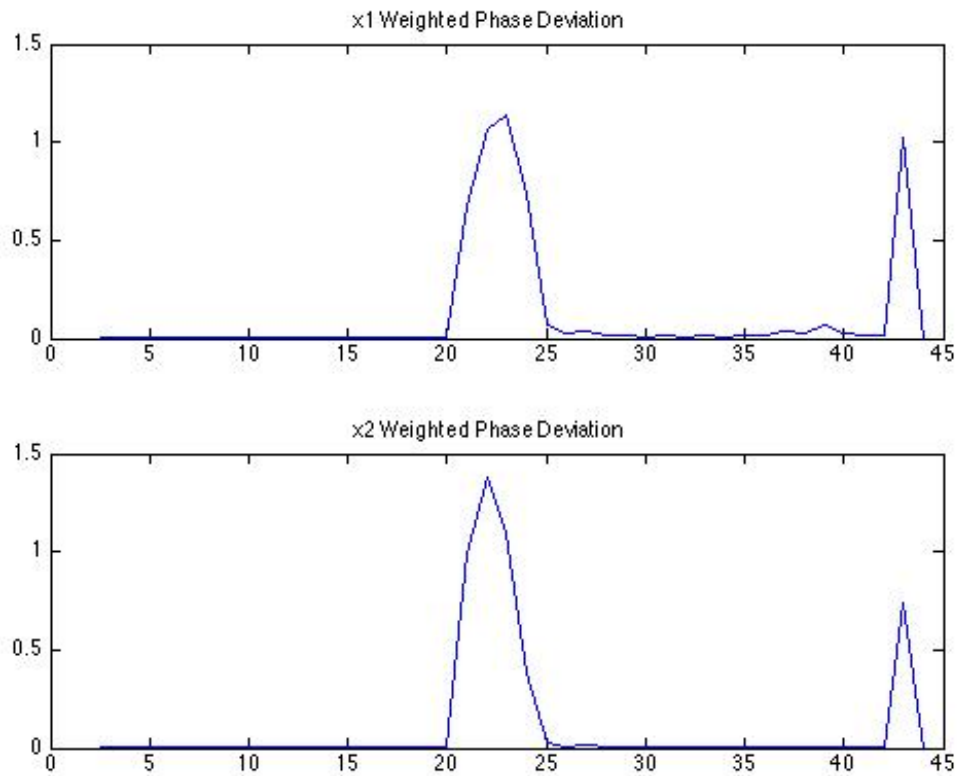
2.



windowSize = 1024, hopSize = 512

An onset occurs at .998 seconds (block 87), apparent in the plot for x2. Peak envelope does not work for changing frequencies, because it does not do frequency analysis. So it only works on the x2 signal.

3.



windowSize = 4096, hopSize = 2048

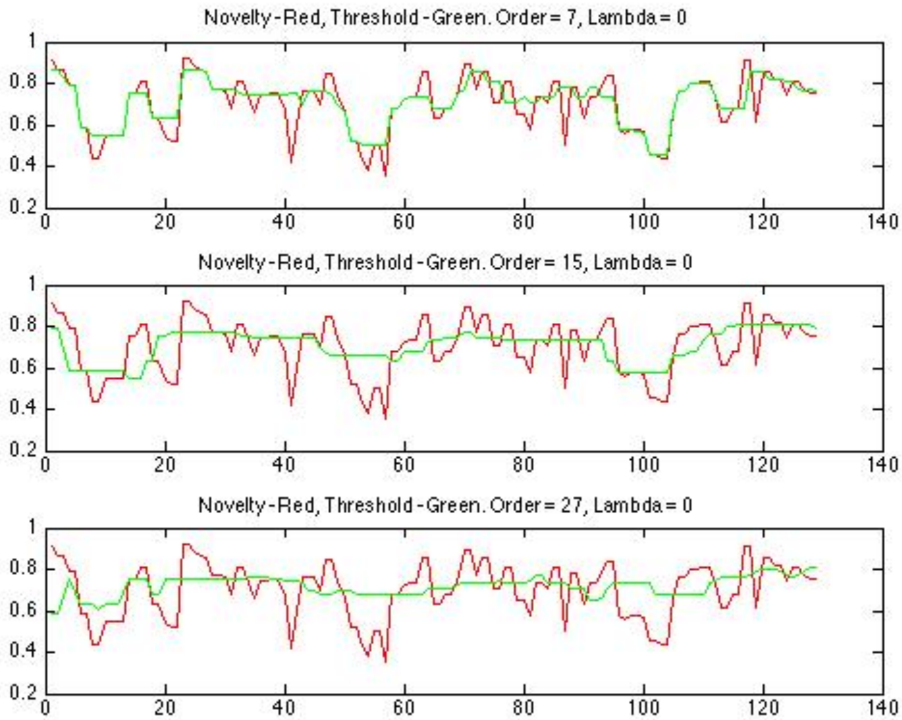
An onset occurs at 1.02 seconds (block 23) in x1 and 0.97 seconds (block 22) in x2; the time resolution is worse here (a larger window size was necessary in order to get rid of noise from the sinusoid smearing into neighboring bins). This method works for both changes in amplitude and changes in frequency.

Bonus: We implemented phase unwrapping by first estimating the number of cycles between hops using the frequency (estimated from the bin index) and time between hops(hopSize / fs). Then we found the difference between the measured phase and our estimate, and called that difference offset. We wrapped the offset and then added that to the unwrapped phase to get our ‘instantaneous phase’.

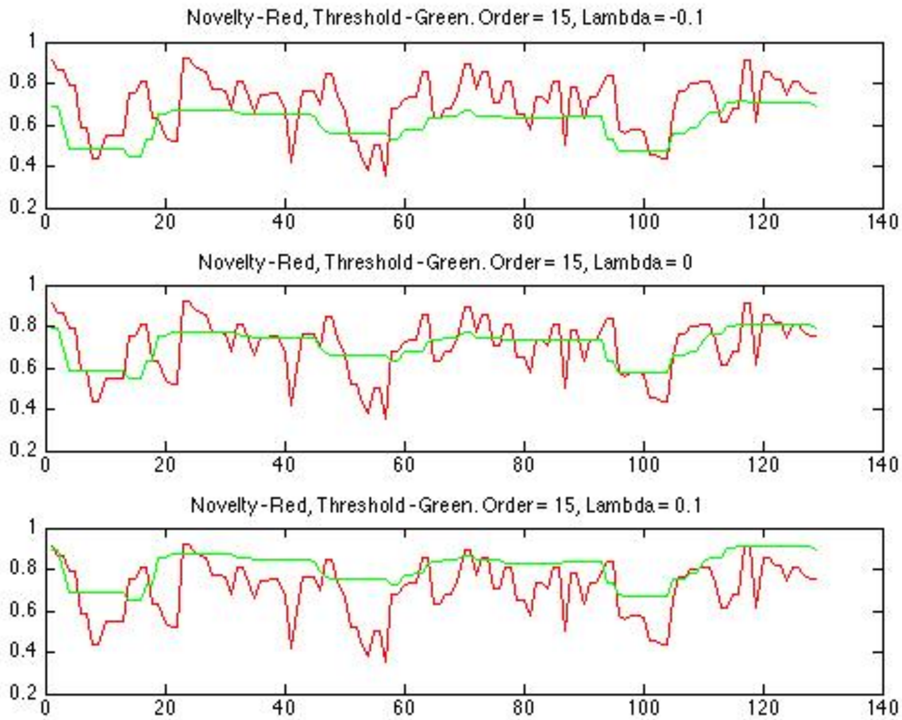
We implemented phase wrapping by taking advantage of modulus. Instead of trying to wrap to [-pi, pi], we wrap (modulus) to [0, 2pi]. To make this work, we first add pi (advance data to [0, 2pi]), then take the modulus, and then subtract pi (go back to [-pi, pi]).

4.

These plots were generated using the peak envelope method.



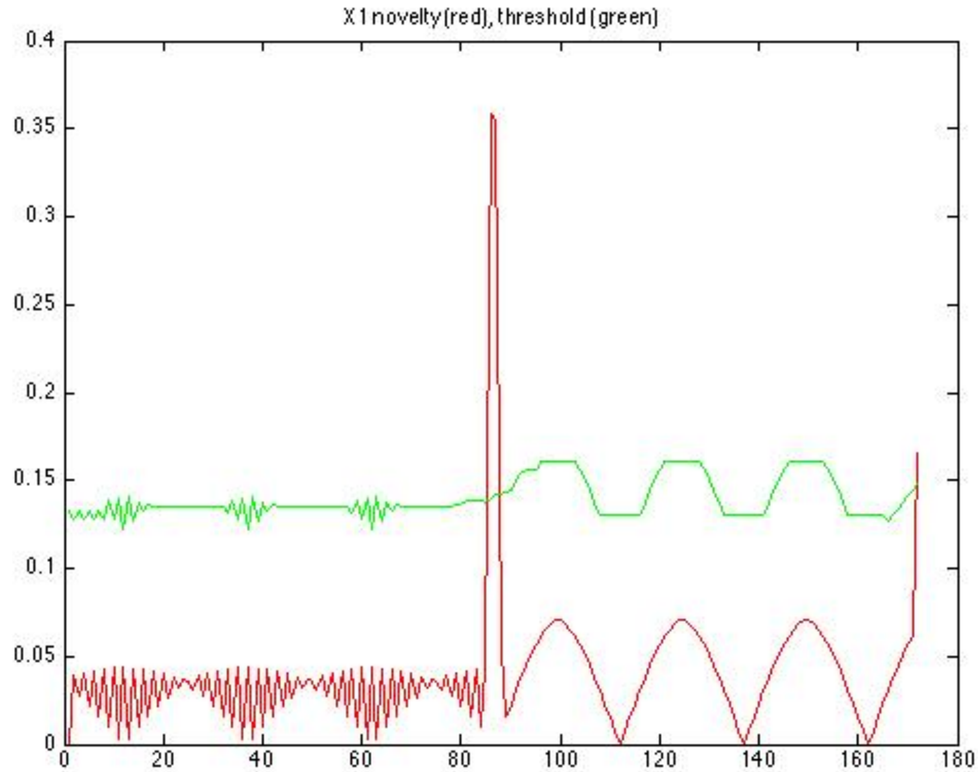
You can see as the order increases, the smoothing is more prominent – fast changes happen less and less.



As lambda increases, the threshold shifts up. This is how we try and balance precision and recall later.

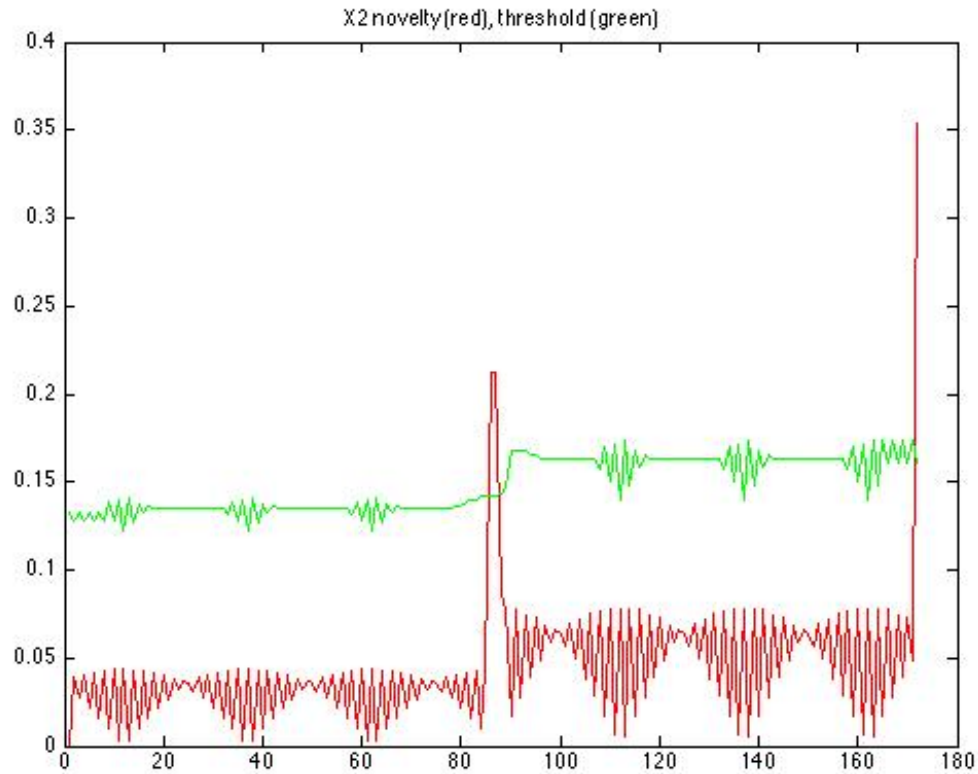
We don’t actually have any delay in the threshold – just look at order = 7 to see that. This is because we started out with a block at [-order/2, order/2] and ended with a block at [length – order/2, length + order/2]. In other words, we (implicitly) zero padded the endpoints.

5. Plots using spectral flux.



onsetTimeSecs_1 =

    1.0101



onsetTimeSecs_2 =

    1.0101

ORDER = 15;
LAMBDA = 0.1;

Spectral flux works well for both tests signals. Honestly we just chose a random order and lambda and it worked. If not, we would have changed them…we can make lambda larger if the threshold was too low, and we can make the order lower if the adaption was too slow.

6. Nothing necessary for the report – we get the correct results for testEval.mat and empty test and ref.

7.
ORDER = 19;
LAMBDA = 0.074;

Flux: Precision - 0.53192 Recall - 0.55468 F-measure - 0.51687
Peak Env: Precision - 0.73768 Recall - 0.40842 F-measure - 0.50803
WPD: Precision - 0.56223, Recall - 0.58894 F-measure - 0.54402
We tried to find the best lambda to equalize precision and recall. Then, if results needed improvement, we would increase or decrease the order, and then optimize for lambda again, and repeat…