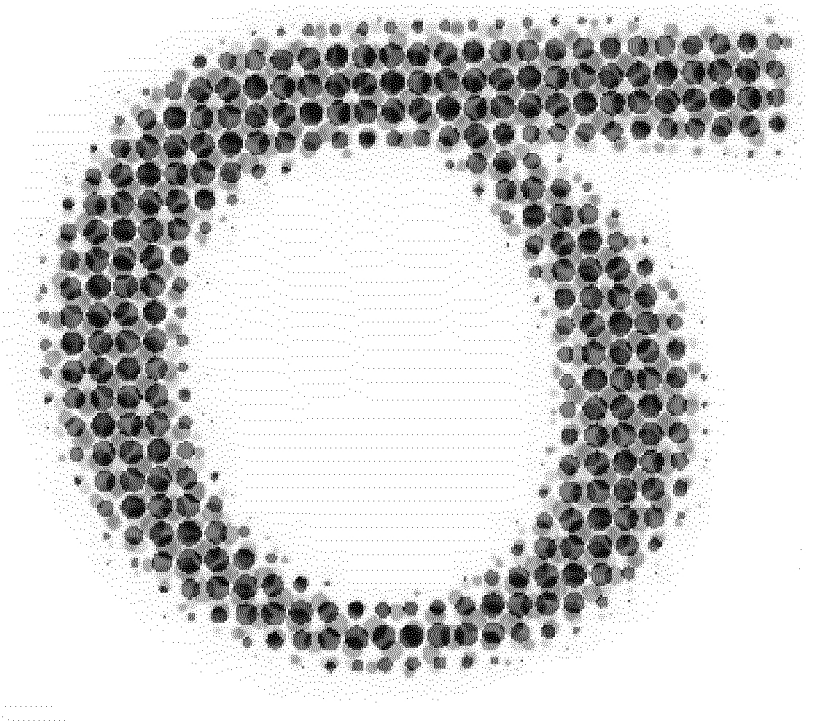


JLIAT /James Whitehead

Deviant

Plots of Standard Deviation of PCM data @ 1 second intervals

The phenomenology of Noise.

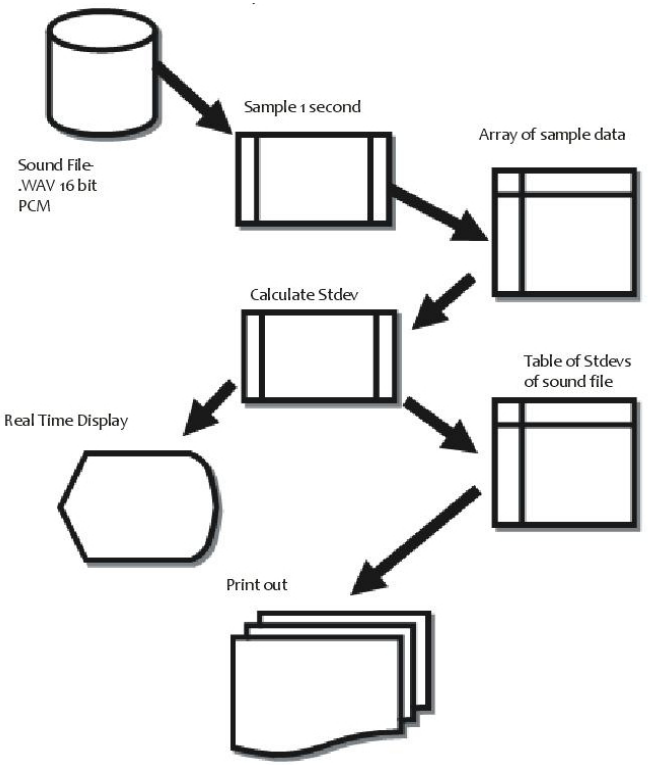
This work was made over a period of time from 2017 onwards. Originally a mix of ‘conventional’ music and noise with reference works. A latter addition concentrated more on Harsh Noise (HN) and Harsh Noise Wall (HNW).

james@jliat.com

This is a catalogue of sound files visualised by taking one second samples (88200 -stereo) and finding the Standard Deviation for each second, then plotting these in time – the X axis. (Produced using custom software.)

“In statistics, the standard deviation (SD, also represented by the Greek letter sigma or the Latin letter s) is a measure that is used to quantify the amount of variation or dispersion of a set of data values. A low standard deviation indicates that the data points tend to be close to the mean (also called the expected value) of the set, while a high standard deviation indicates that the data points are spread out over a wider range of values.” …
” .. the standard deviation represents noise and other interference… ”

Here it is used initially merely as (another) means to compare, contrast and visualise sound pieces.



‘Standard deviation’ is a statistical method widely used in science and data analysis. It is used for measuring confidence in data sets as well as having other functions. Confidence in these data sets as being meaningful and of value. Though somewhat complex to non mathematicians and those not concerned with statistical analysis – and I include myself here - the principle is fairly simple. Standard deviation measures not the average (the mean) of a dataset but the spread of data around this mean. This supplies the idea of “confidence” – if the standard deviation is low then the data clusters around a mean (average), whereas if it is high it is evenly spread out across the range of possible results, it is more random. A low standard deviation indicates that there is some structure, and so possibly meaning which is of value, a high standard deviation the opposite.
For example – a data stream might carry numbers from 1 through 10.
Here are two streams of data.

4 6 7 5 6 4 6 5 6 5

9 2 3 4 10 5 6 1 7 8

The average of these two streams is 5.4 and 5.5 respectively. The standard deviation is 0.97 and 3.03.
The first dataset seems to have a ‘fiveness’ about it, the second seems arbitrary. In other words the first data set seems to provide us with information- it has a ‘message’ – the second set has no information, no message and is termed – ‘noise’.
The higher the standard deviation indicates a more ‘noisy’ data stream.
At extreme levels no coherent ‘message’ can be found – the signal is random, chaotic, noise. It follows that random data will have a greater standard deviation, as it is not related to anything particular within the dataset or itself.

Audio Cds store sound by encoding the soundwave as a series of numbers. Each second of sound is encoded as 44100 separate numbers (samples), twice for stereo. These numbers range from +32767 to -32768. They in effect represent a graph of the soundwave. The sound now being ‘digitized’ allows for numeric processing and analysis. (An Audio CD is in effect just a series of numbers.)

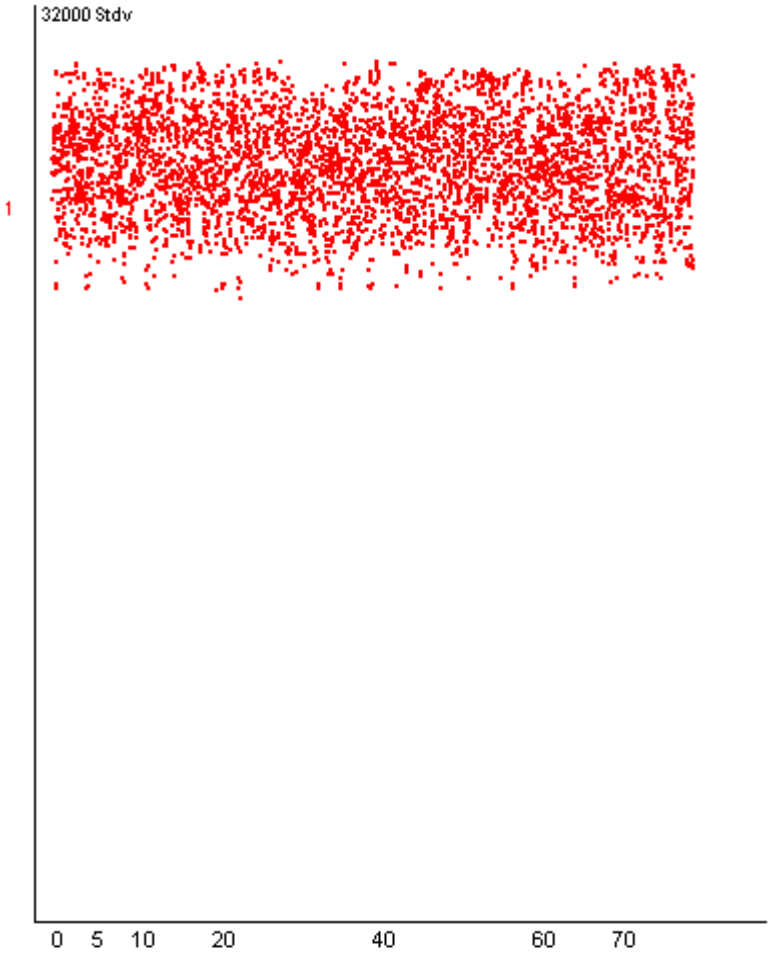
This work in part continues previous research into the objective qualities of so called 'Noise Music' and the differences it shows compared to other musics when the standard deviation of the samples of each are compared. That is the 'Noise Music' shows a far greater Stdev indicating in statistical terms a lack of confidence in any coherent data, in other words 'noise' rather than signal. This, despite many who seem unwilling to accept this, provides a clear objective measure as to the difference between 'Noise Music' and 'conventional music', even the music which is considered avant-garde and by some less 'musical'.

The work also allows for analysis and comparisons between different pieces and types of music.

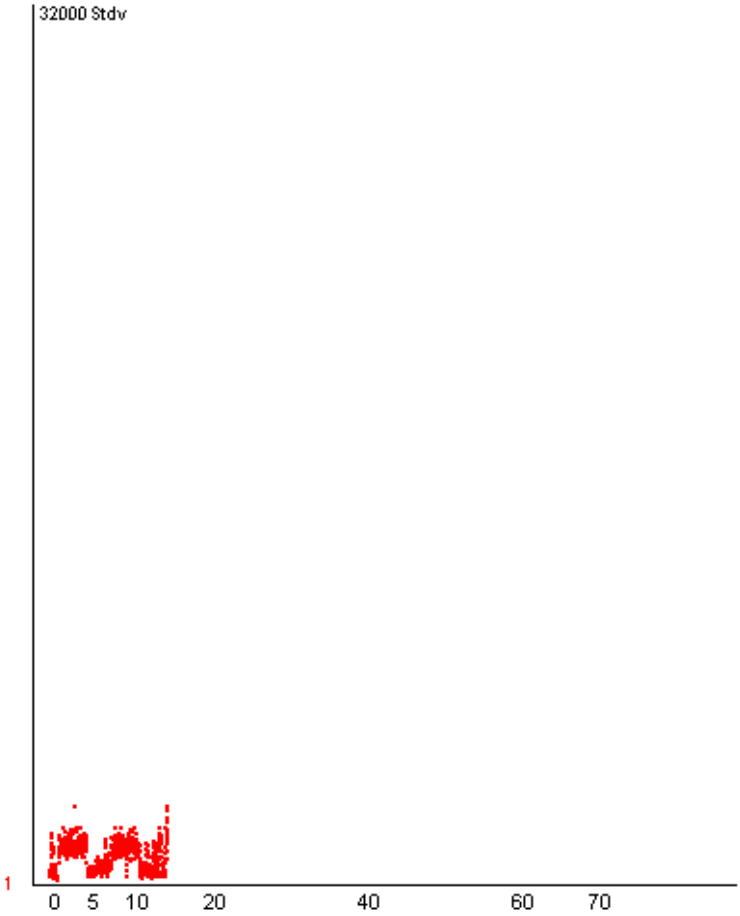
The height of the graph shows a greater Stdev (noise) and the compactness the uniformity of the sound. It should be noted that the height does not always directly correspond to volume or loudness as in the case of symphonic works, for instance, some passages are as loud as some noise. A continuous "Wall" of high volume 'noise' - (random numbers) will produce a high Standard Deviation. (A technical reason for high volumes causing noise is that the waveforms become 'clipped' -this is a source of noise, just as playing any sound or speech at high levels will introduce real distortion to the signal. The 'noise' in this case is not just a function of hearing, actual loss of data and coherence occurs in the signal.)

Examples of most of the source material can be found on the internet for instance -youtube-.

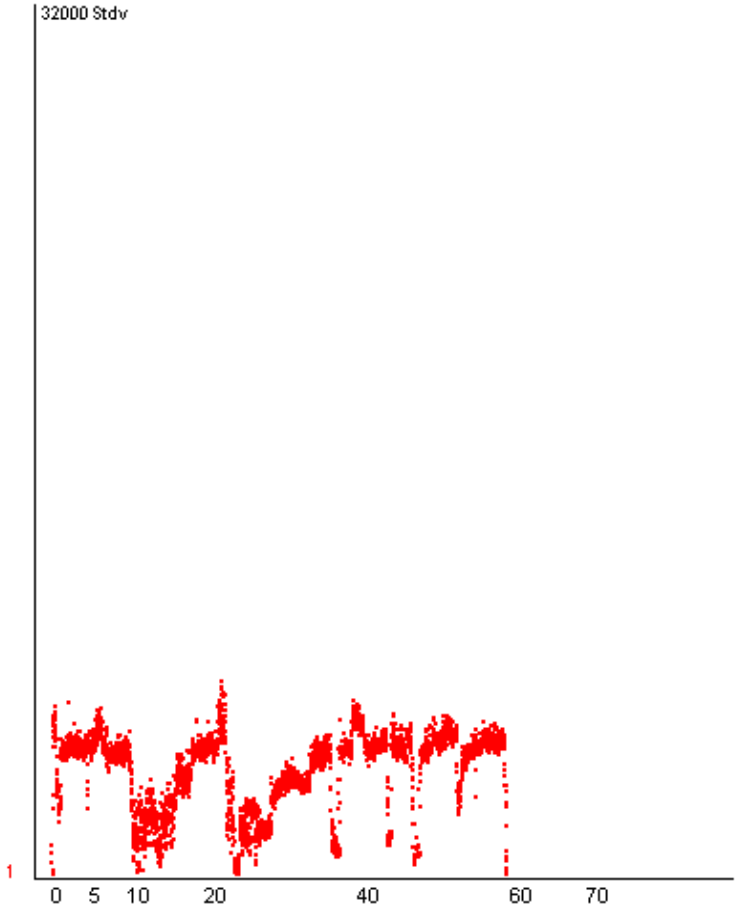
Page 1 is a Plot showing a Harsh Noise generayted piece on 07/12/2014 with a very high standard deviation indicating a very noisy and incoherent sounds. (JLIAT)



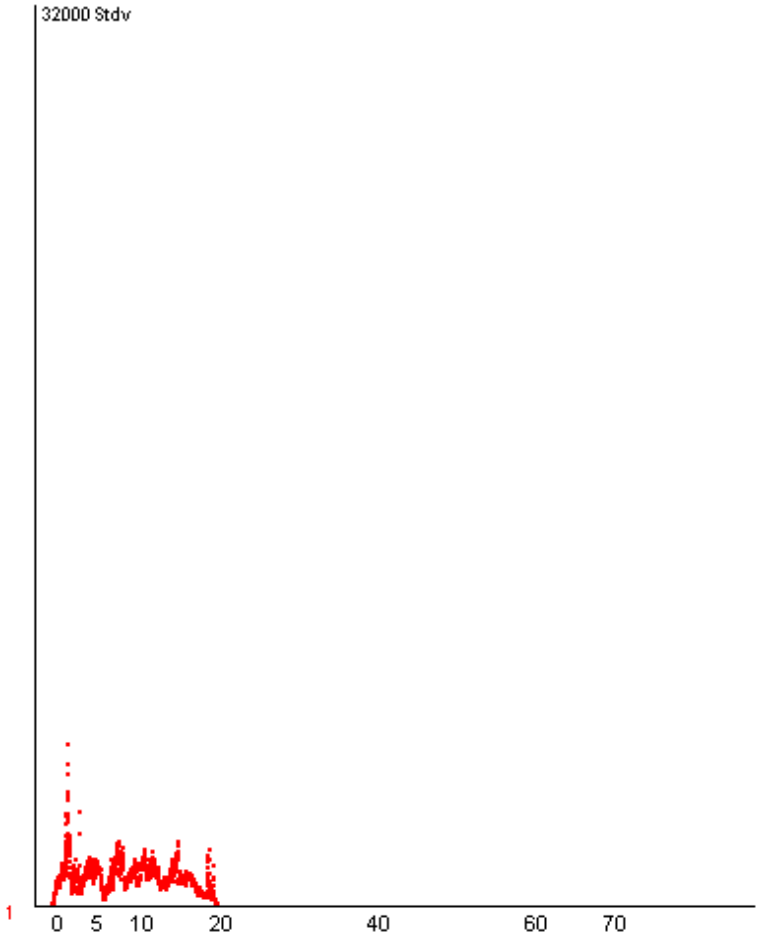
1 = My Documents\((((((((((((std\ms_07_12_2014_18_30_15_8-HN.wav plot



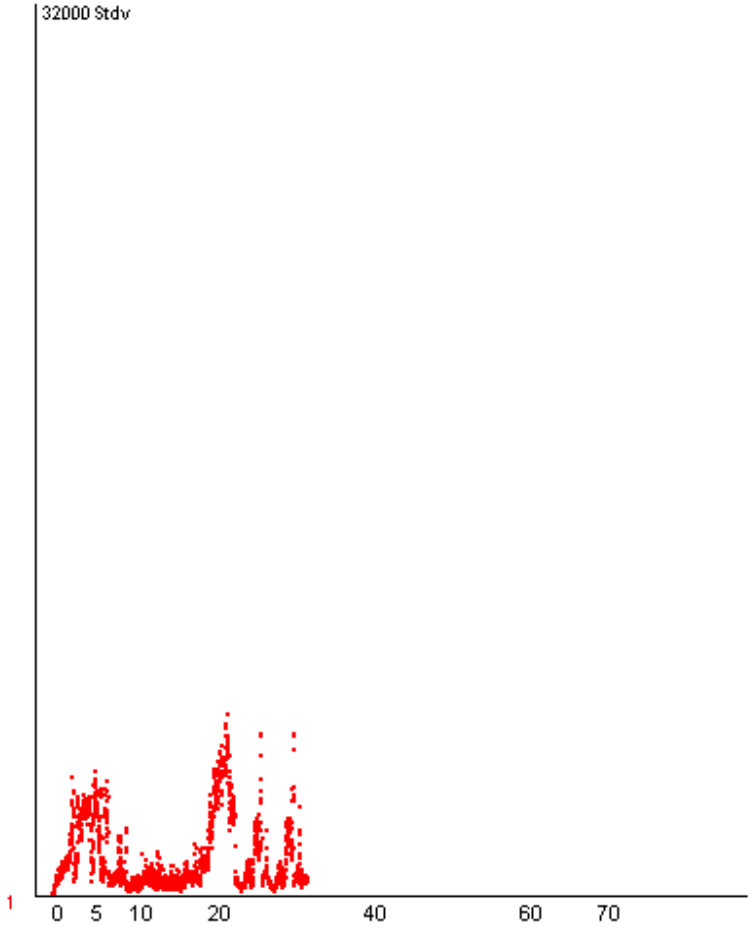
1 = tano String Quartet Plays Beethoven Op. 133, Grosse Fuge.wav plot



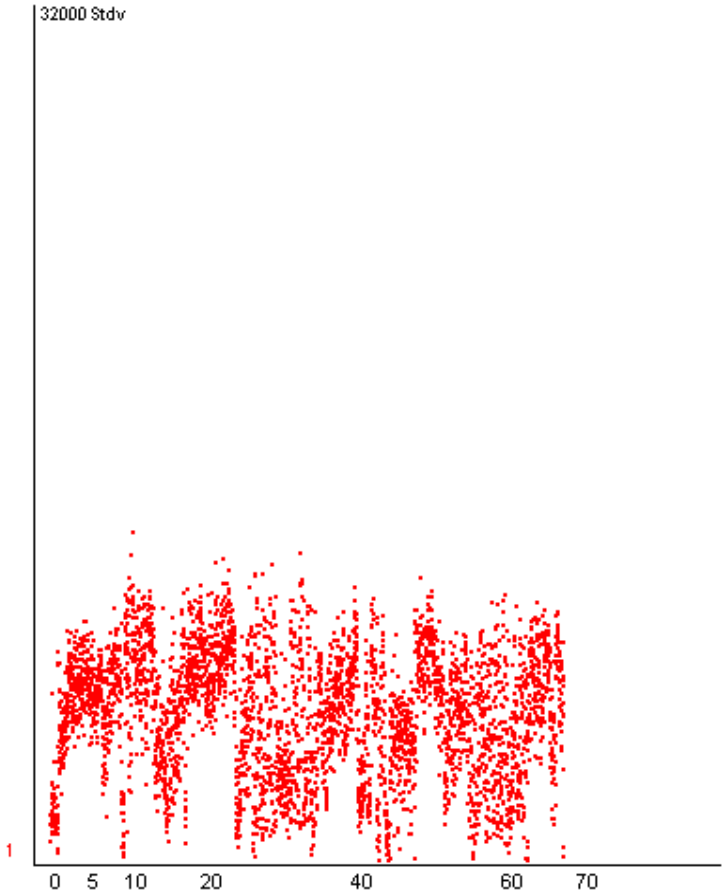
1 = ((((std\Acid Mothers Temple Boiler Room x NRMAL Live Set.wav plot



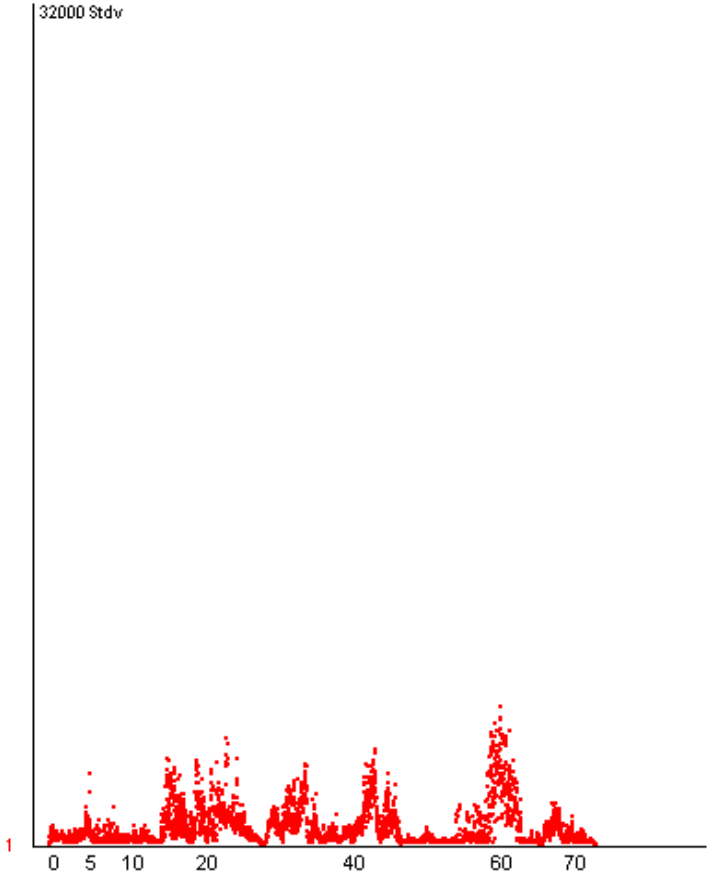
1 = ((((((((((((std\John Adams - Common tones in simple time.wav plot



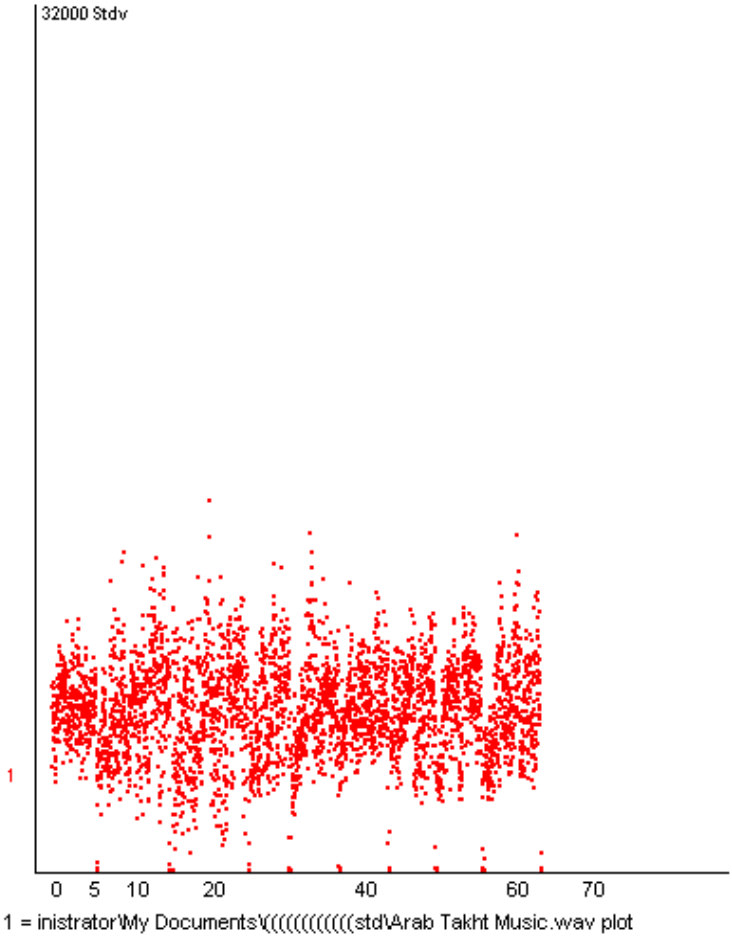
1 = istrator\My Documents\((((((((((((std\Adams-Shaker Loops.wav plot

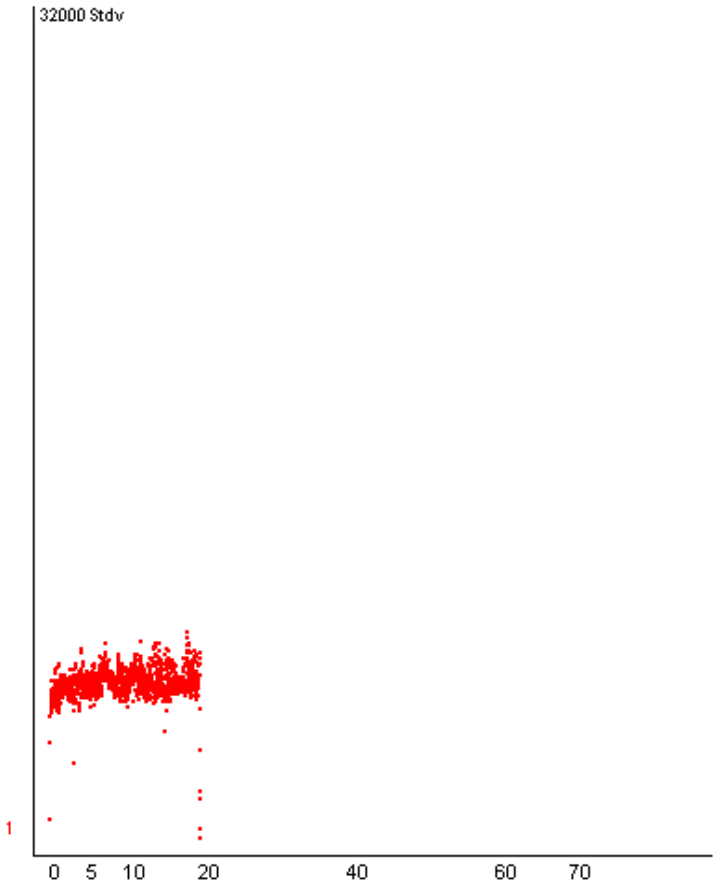


1 = (((((((((std\update\Albert Ayler - The Copenhagen tapes.wav plot

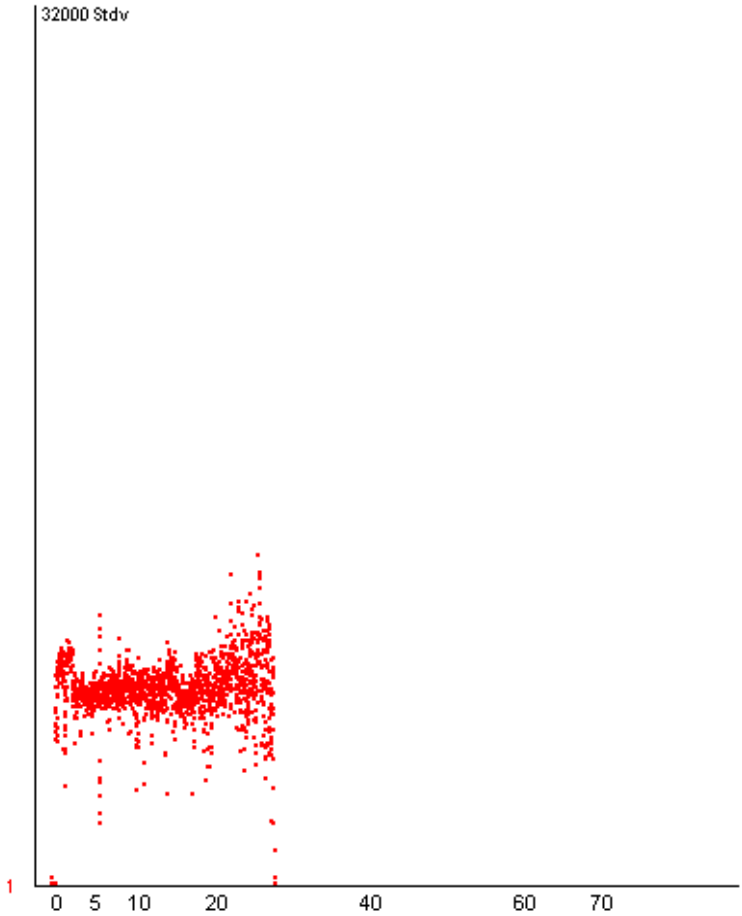


1 = uments\((((((((((((std\AMM - The Nameless Uncarved Block.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = inistrator\My Documents\((((((((((((std\Arab Takht Music.wav plot



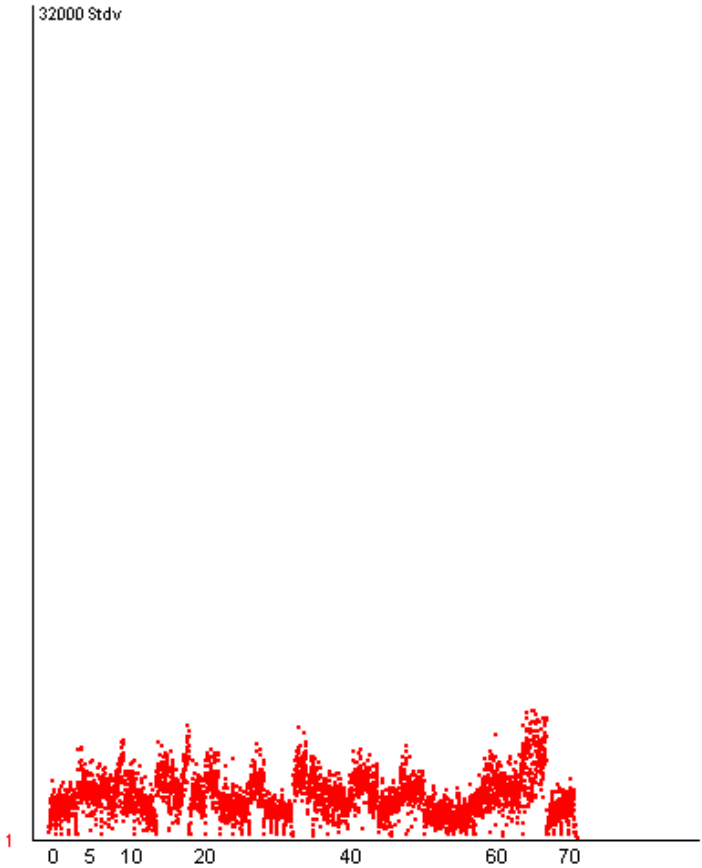
1 = E:\Noise\Atrax Morgue - Gas Chamber For Humanity.wav plot



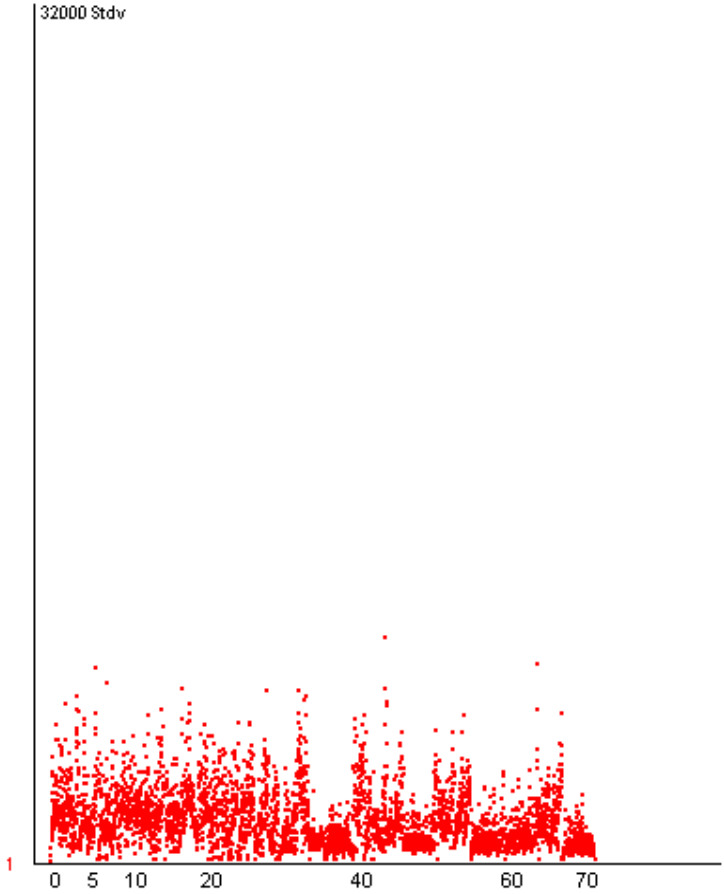
1 = E:\Noise\Atrax Morgue - Live Modina 1995.wav plot

1 = Documents\((((((((((((std\Milton Babbitt - Arie da Capo.wav plot

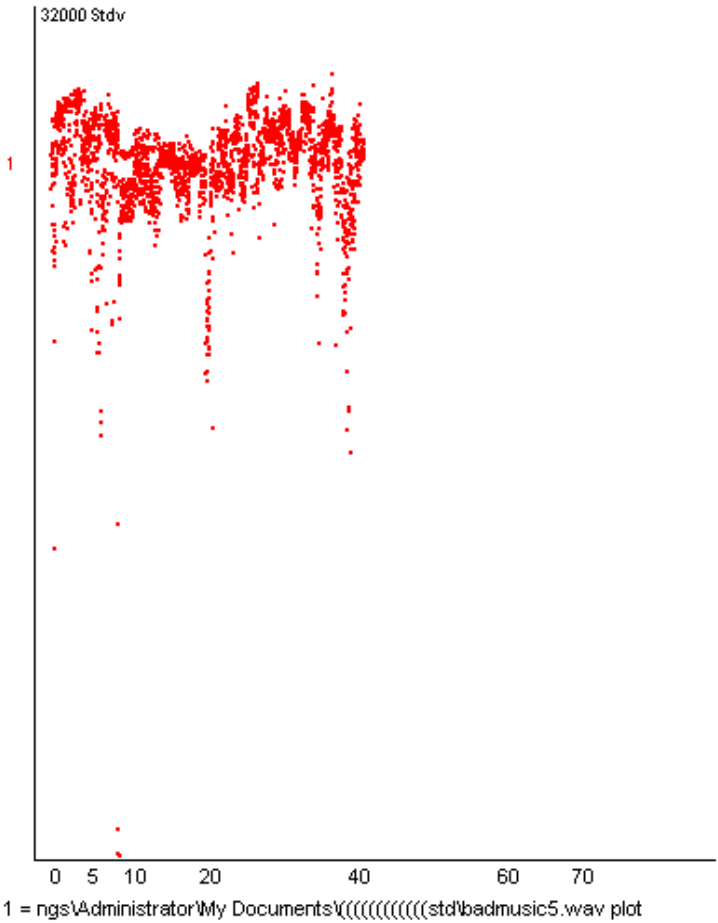
1 = (((((stdMilton Babbitt Concerto for Piano and Orchestra.wav plot

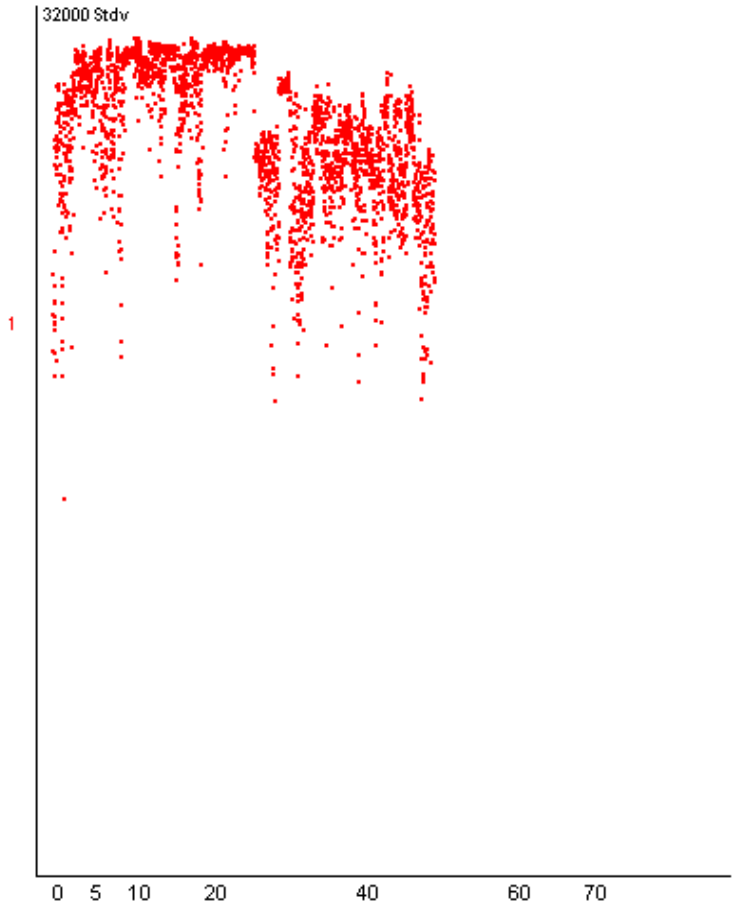


1 = Bach, Variaciones Goldberg BWV 988. András Schiff, piano.wav plot

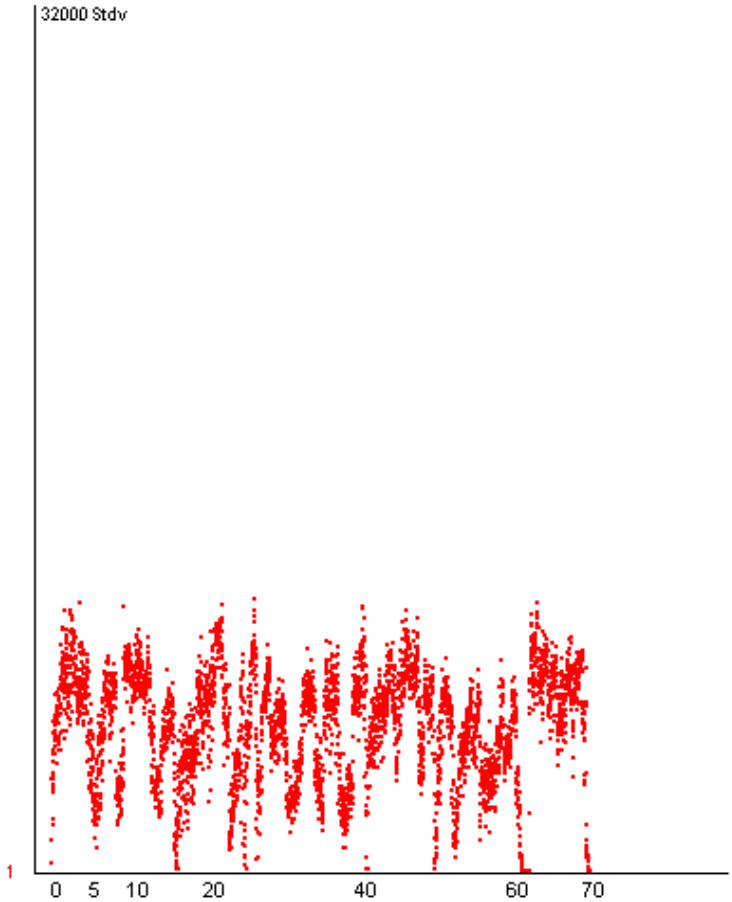


1 = inistrator\My Documents\((((((((((((std\J.S. Bach Motets.wav plot

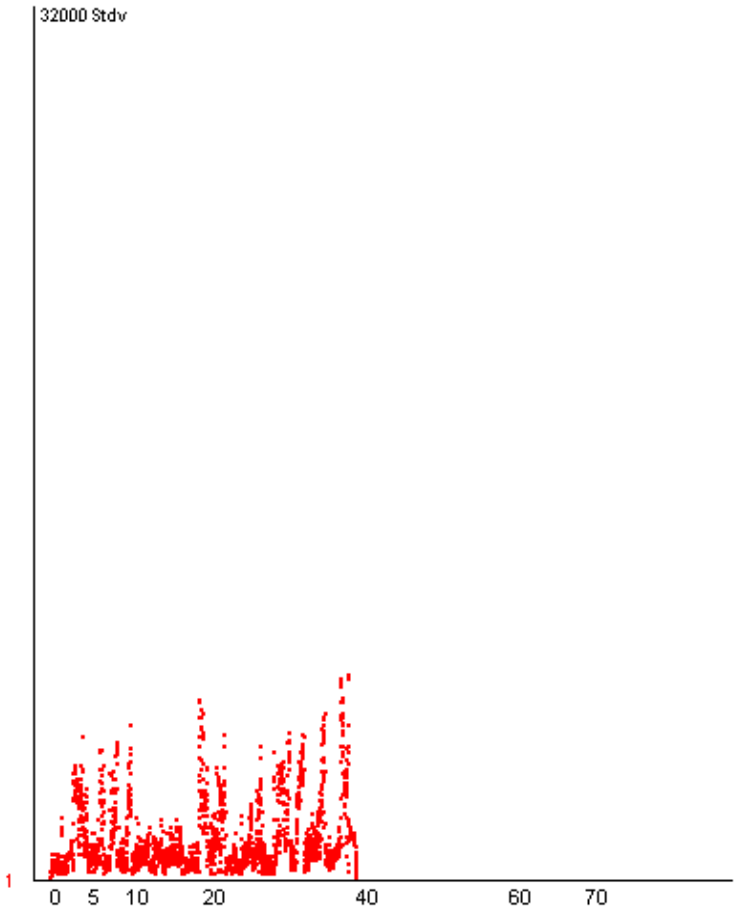

32000 Stdv
1
0
5
10
20
40
60
70
1 = ngs\Administrator\My Documents\((((((((((((std\badmusic5.wav plot



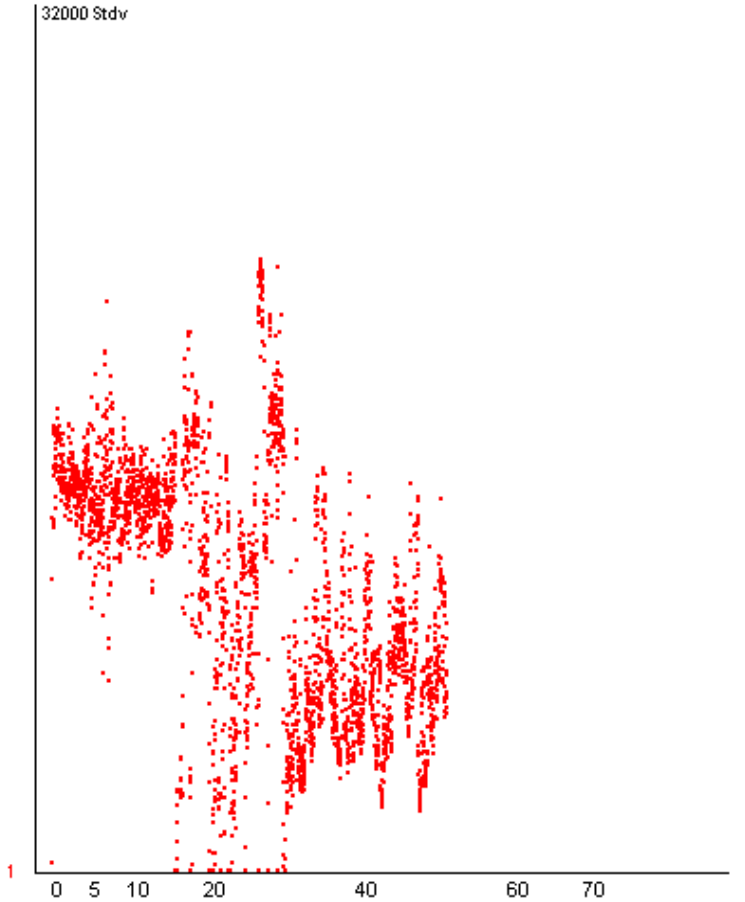
1 = ngs\Administrator\My Documents\((((((((((((std\Badmusic2.wav plot



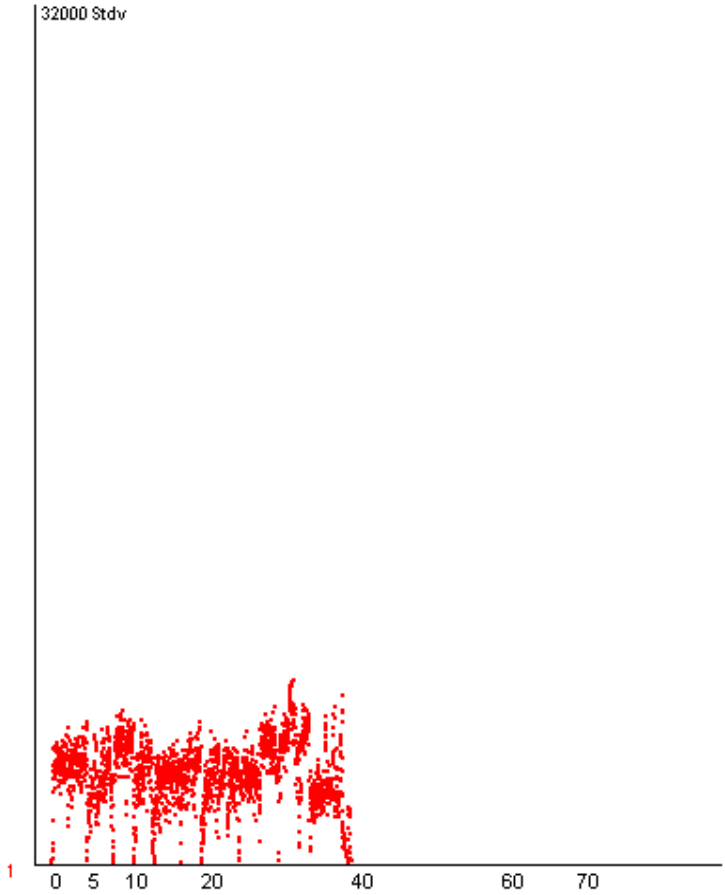
1 = y feat. Jamaaladeen Tacuma _Grant Calvin Weston Mirakle.wav plot



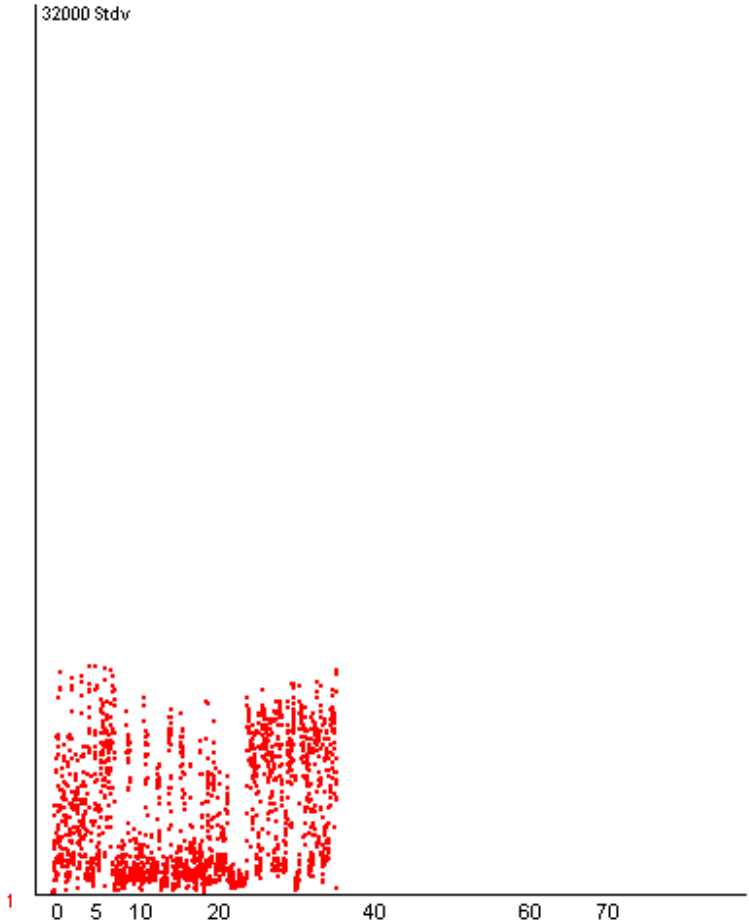
1 = ((std\Béla Bartók - Concerto For Orchestra (1943) (Full).wav plot



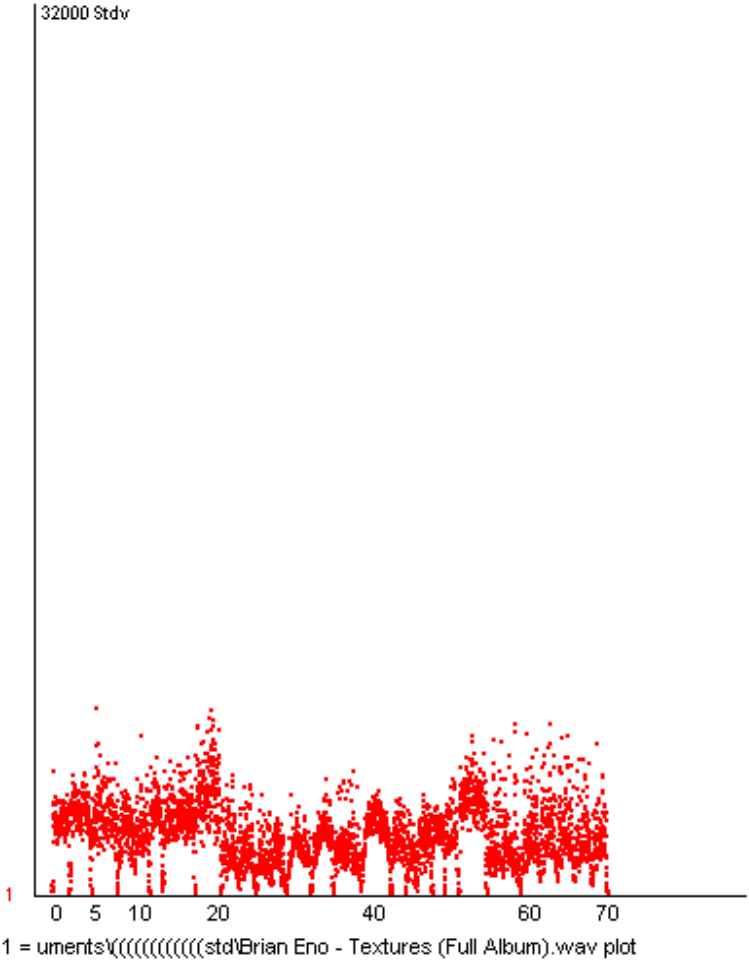
1 = E:\Noise\Bastard Noise - Hospital - Split (Full Album).wav plot

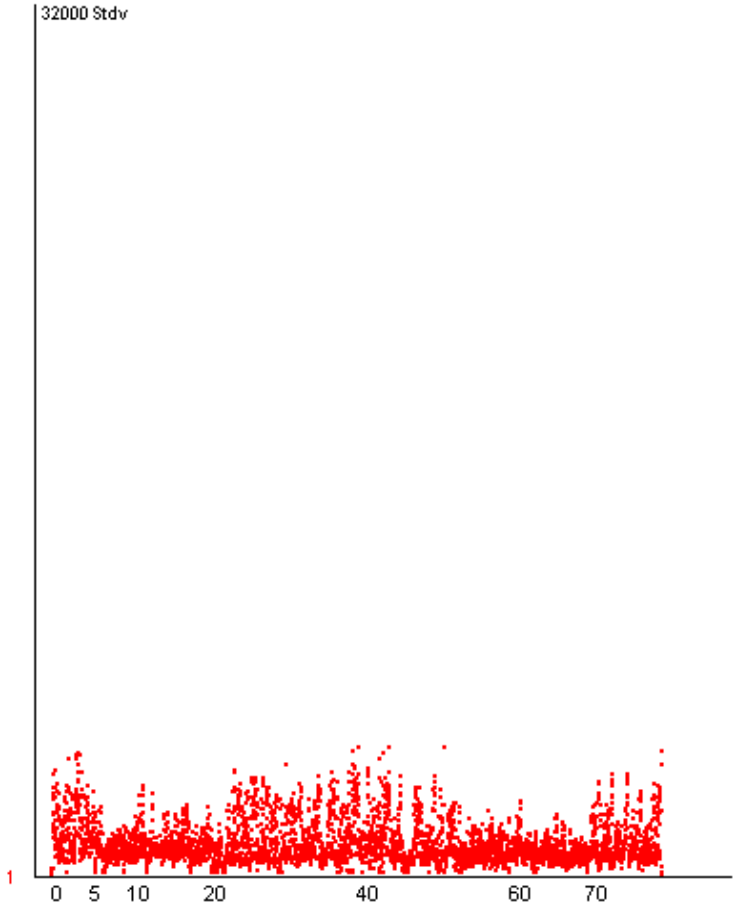


1 = ((((std\The Beatles Sgt. Peppers Lonely Hearts Club Band.wav plot

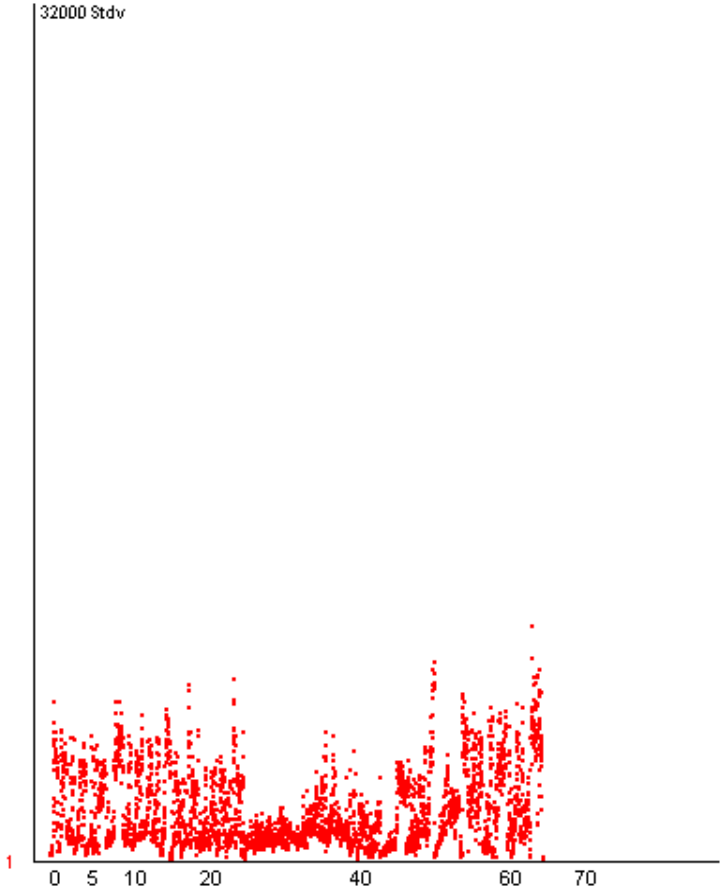


1 = gs\Administrator\My Documents\((((((((((((std\beet5-cmin.wav plot

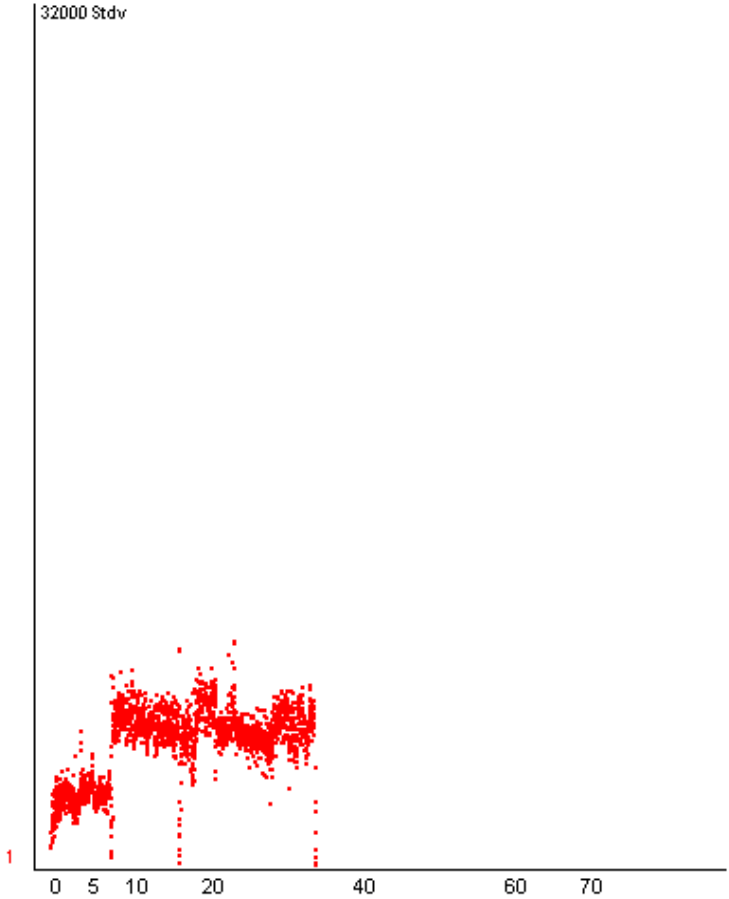

32000 Stdv
1
0
5
10
20
40
60
70
1 = uments\((((((((((((std\Brian Eno - Textures (Full Album).wav plot



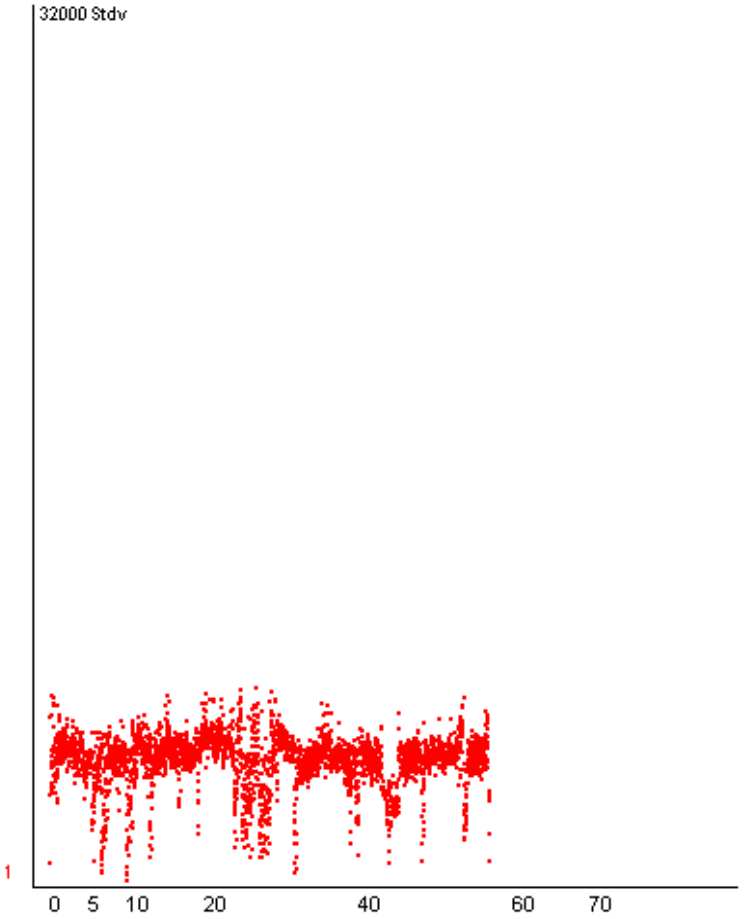
1 = odin Quartet plays Beethoven String Quartet Op.127 _130.wav plot



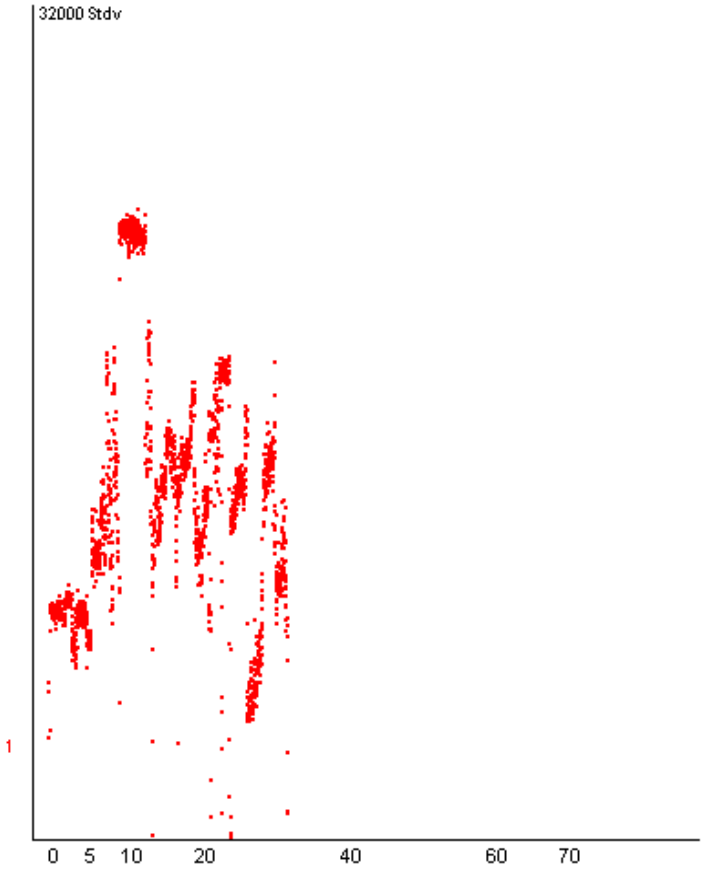
1 = \My Documents\((((((((((((std\Symphony No. 9 ~ Beethoven.wav plot



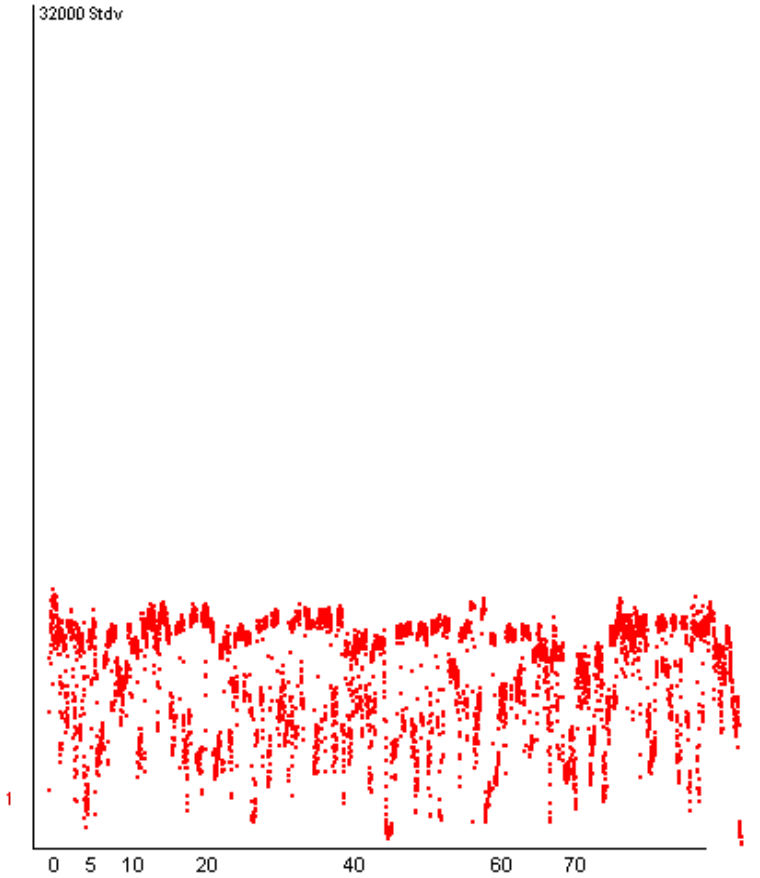
1 = My Documents\((((((((((((std\Berber music (NORTH AFRICA).wav plot



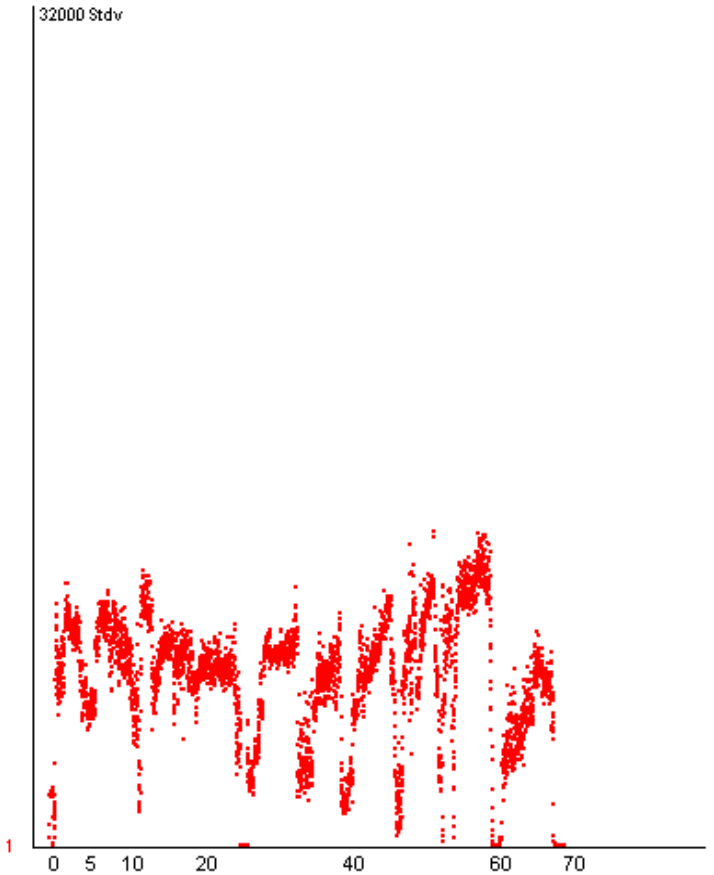
1 = (((std\Black Sabbath - Never Say Die 1978 (Full Concert).wav plot



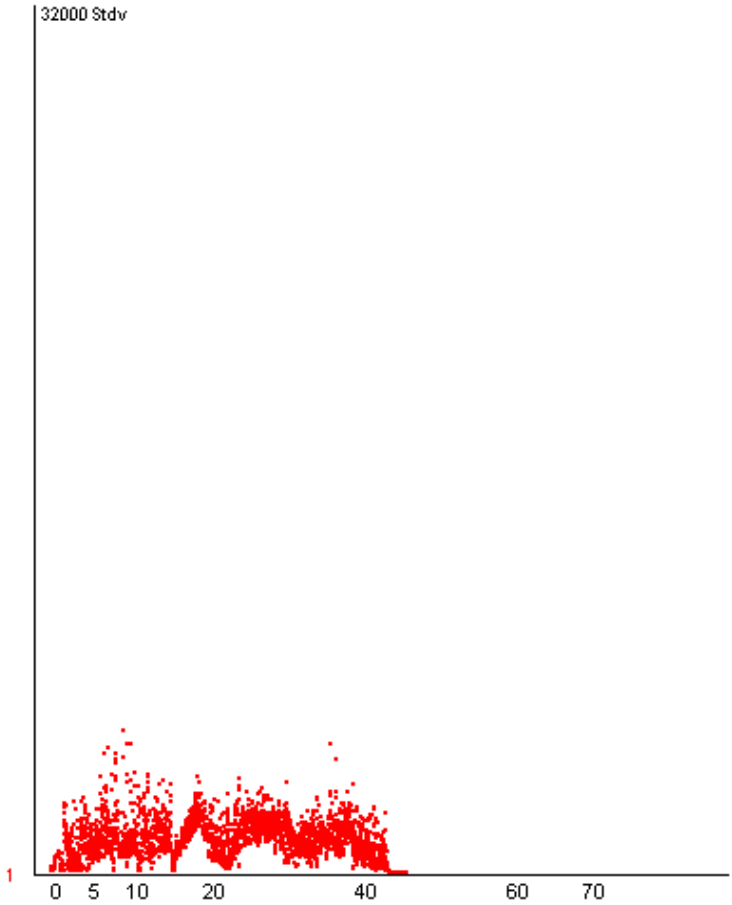
1 = bath Black Fiji - Mistake Of A Small Bird (Full 2010 LP).wav plot



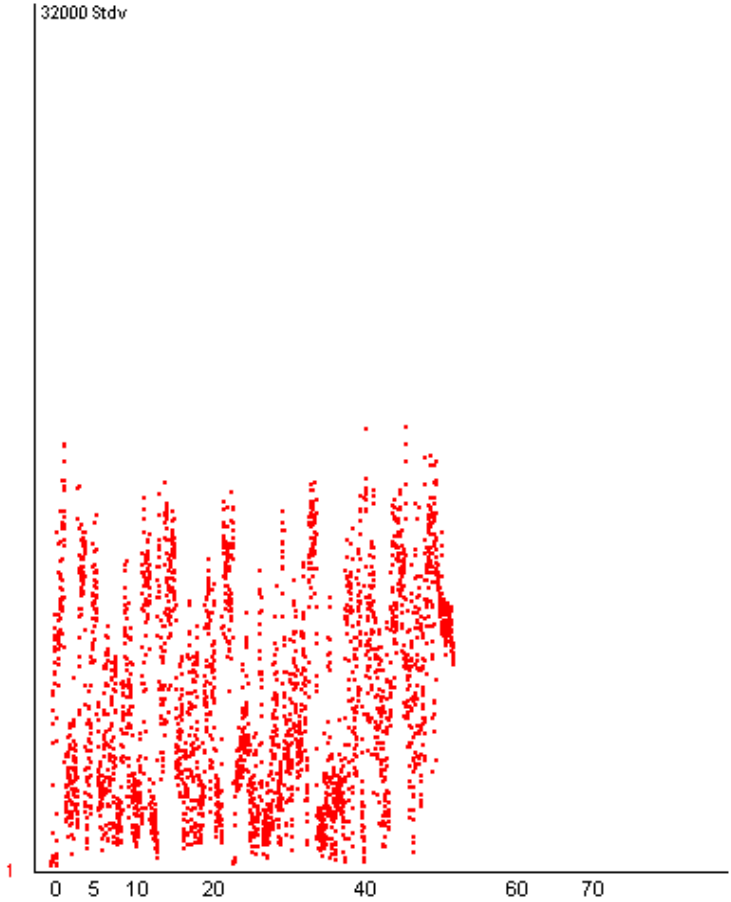
1 = F:\Stephan Bodzin Live @ NGHTDVSN ADE 2015 Full Set.wav plot



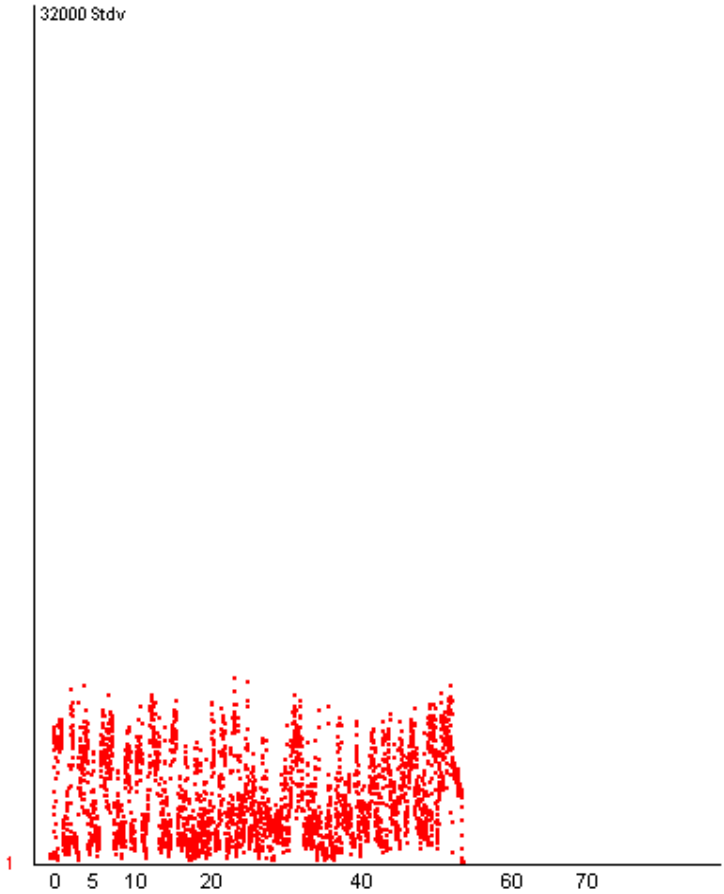
1 = E:\Noise\Boredoms - Vision Creation Newsun (full).wav plot



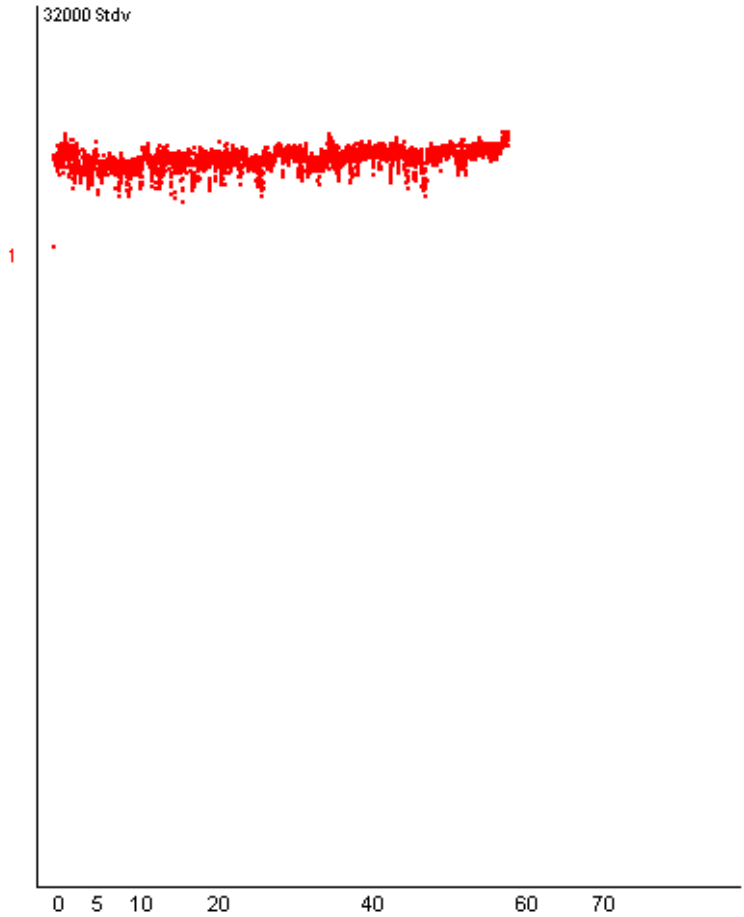
1 = Répons - Ensemble intercontemporain - Matthias Pintscher.wav plot



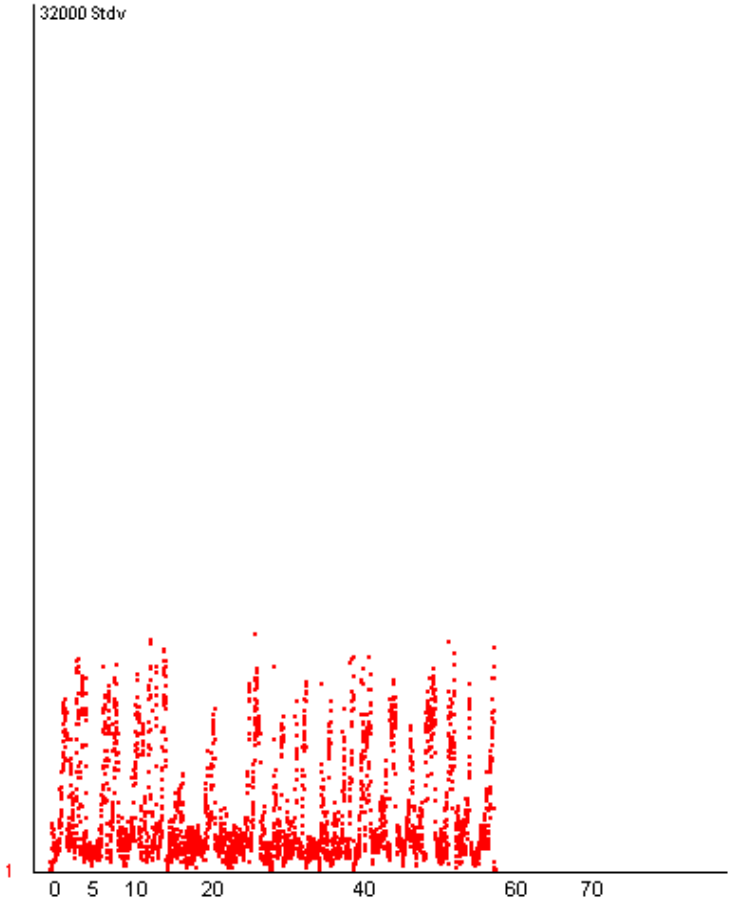
1 = (std\Brahms - Piano Concerto No. 1 - Hélène Grimaud [HD].wav plot



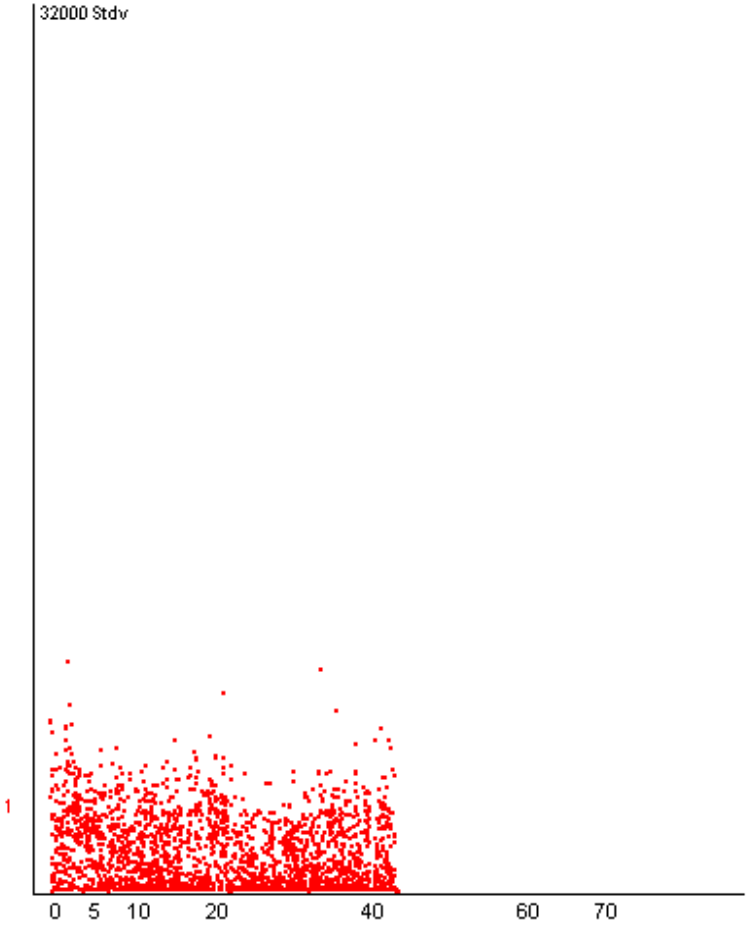
1 = (((((((((std\Brahms - Symphony No. 1 (Leonard Bernstein).wav plot



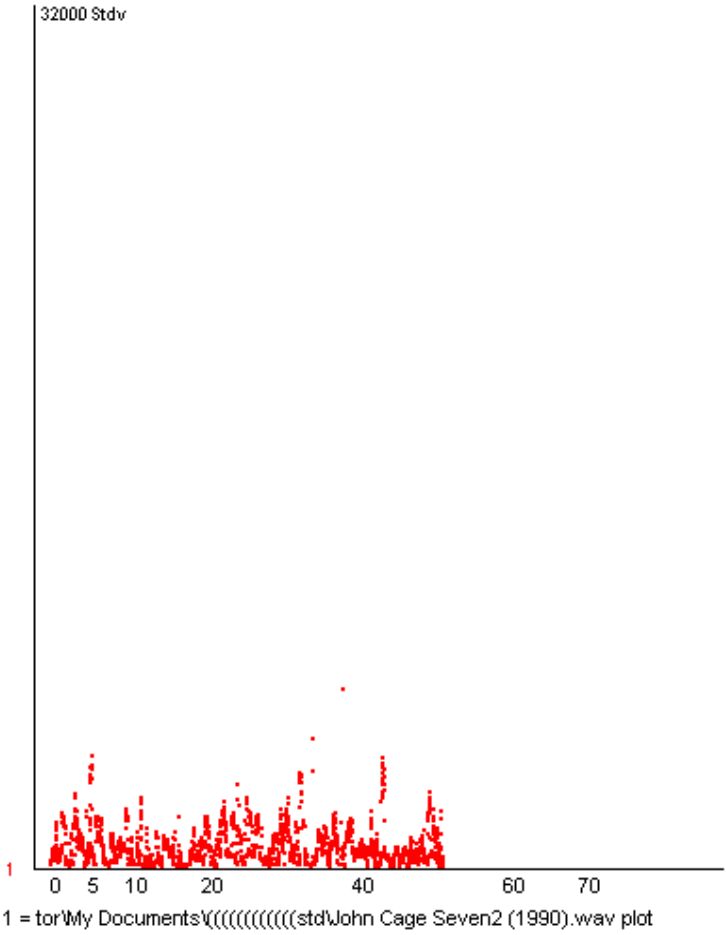
1 = cuments\(((((((((((std\The Rita - Thousand Of Dead Gods.wav plot



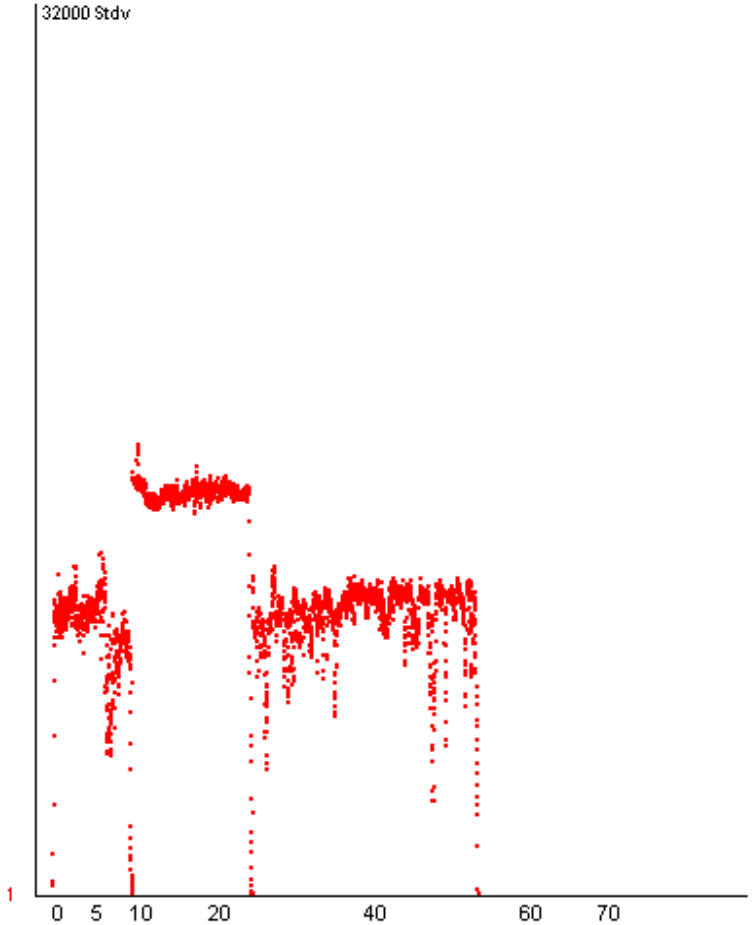
1 = ((((((((((std\Anton Bruckner - Symphony No. 4 'Romantic'.wav plot



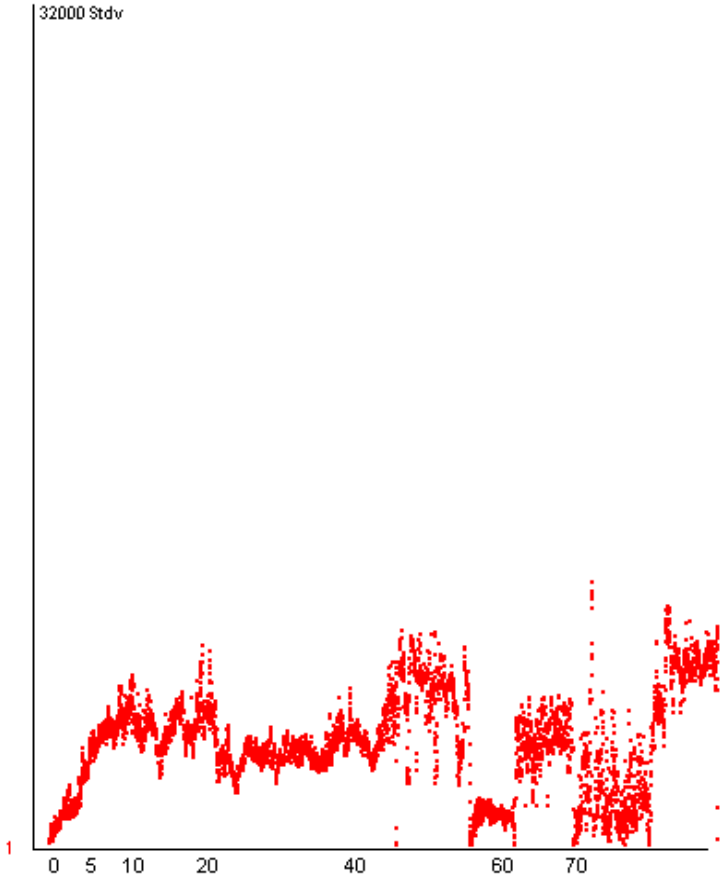
1 = uments\((((((((((((std\John Cage Music of Changes (1951).wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = tor\My Documents\((((((((((((std\John Cage Seven2 (1990).wav plot

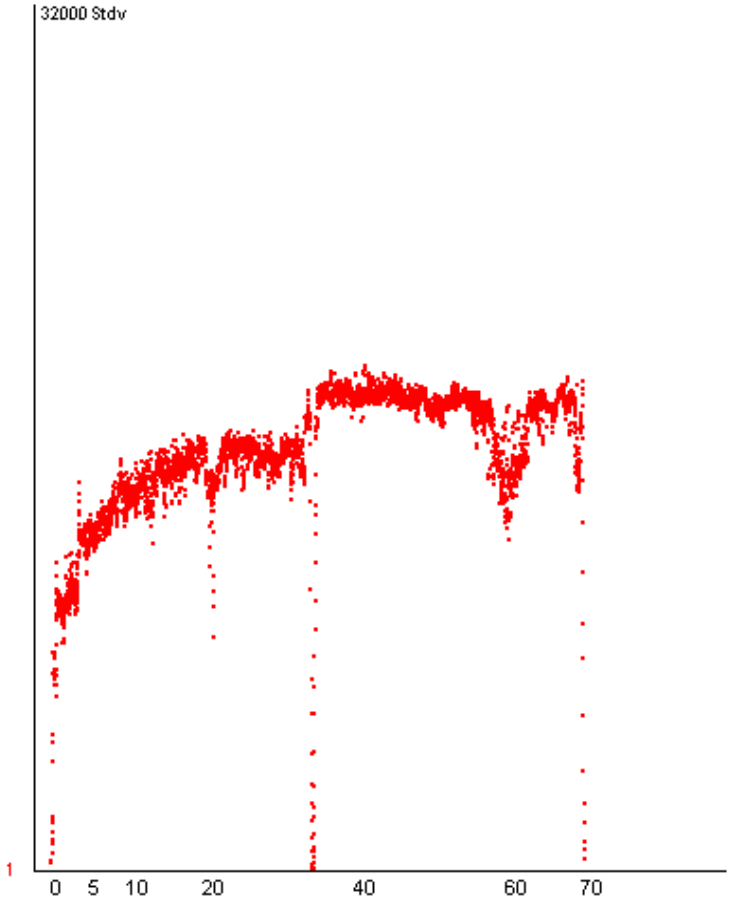
1 = td\John Cage - Sonatas and Interludes (Thomas Nicholson).wav plot



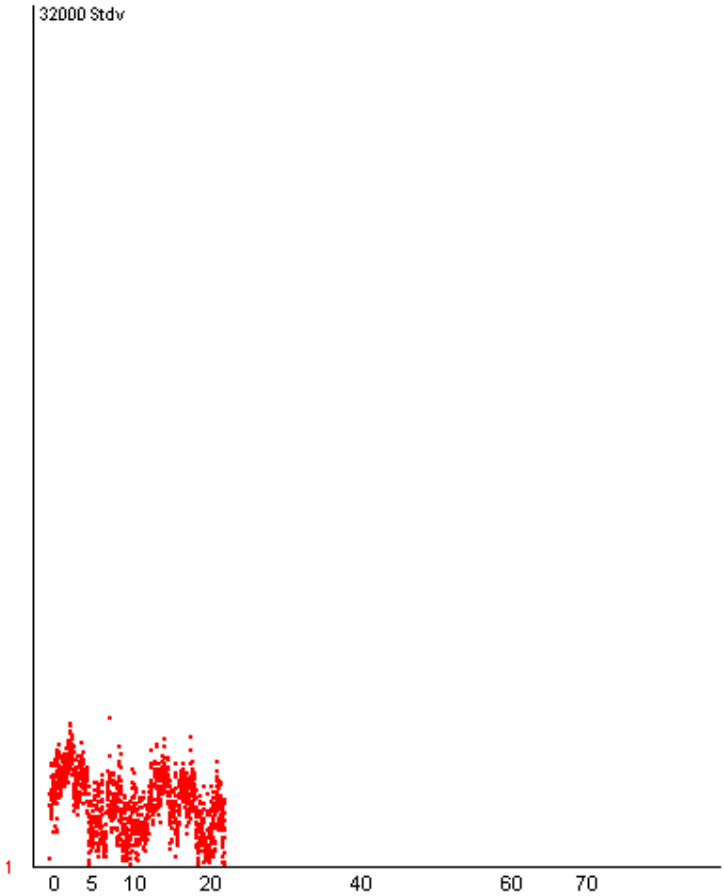
1 = tor\My Documents\((((((((((((std\C.C.C.C. - Love _Noise.wav plot



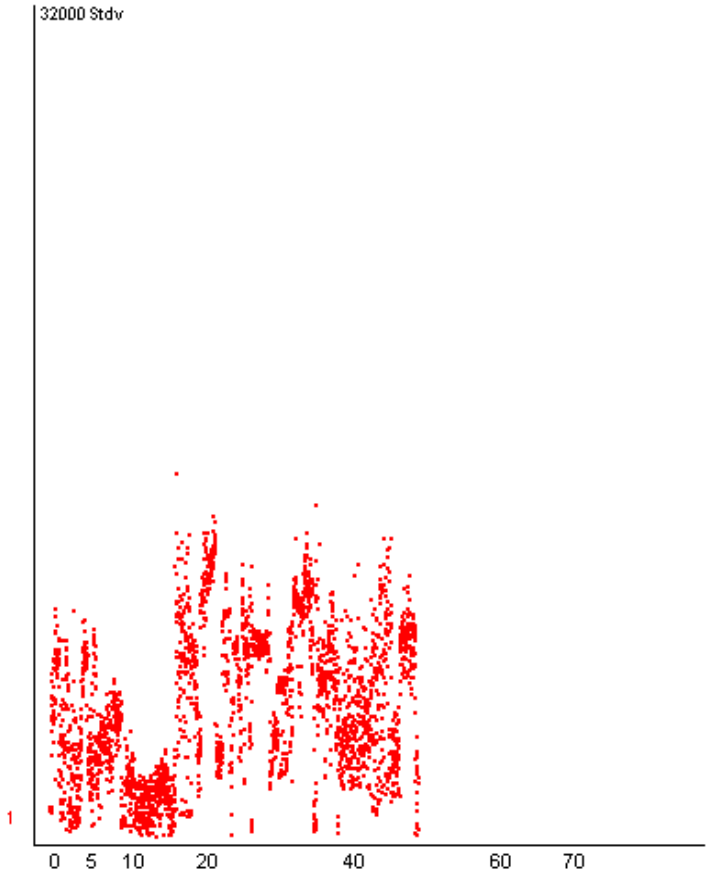
1 = E:\Noise\C.C.C.C. - Amplified Crystal II (Full Album).wav plot



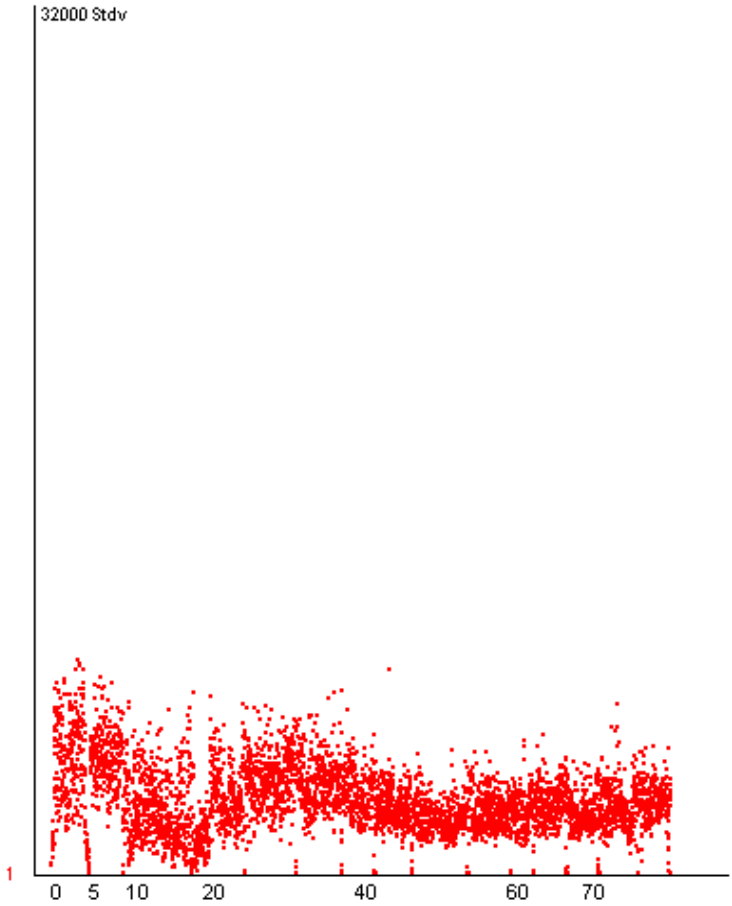
1 = \C.C.C.C. - Loud Sounds Dopa-Live In U.S.A. (Full Album).wav plot



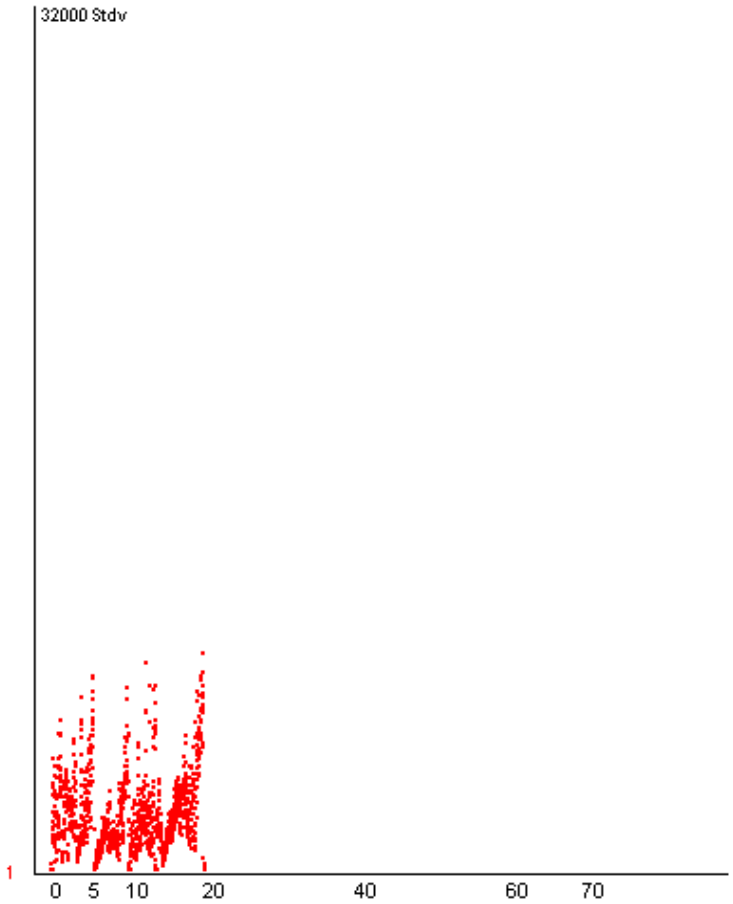
1 = ((std\update\Cecil Taylor - Unit Structures (Full Album).wav plot



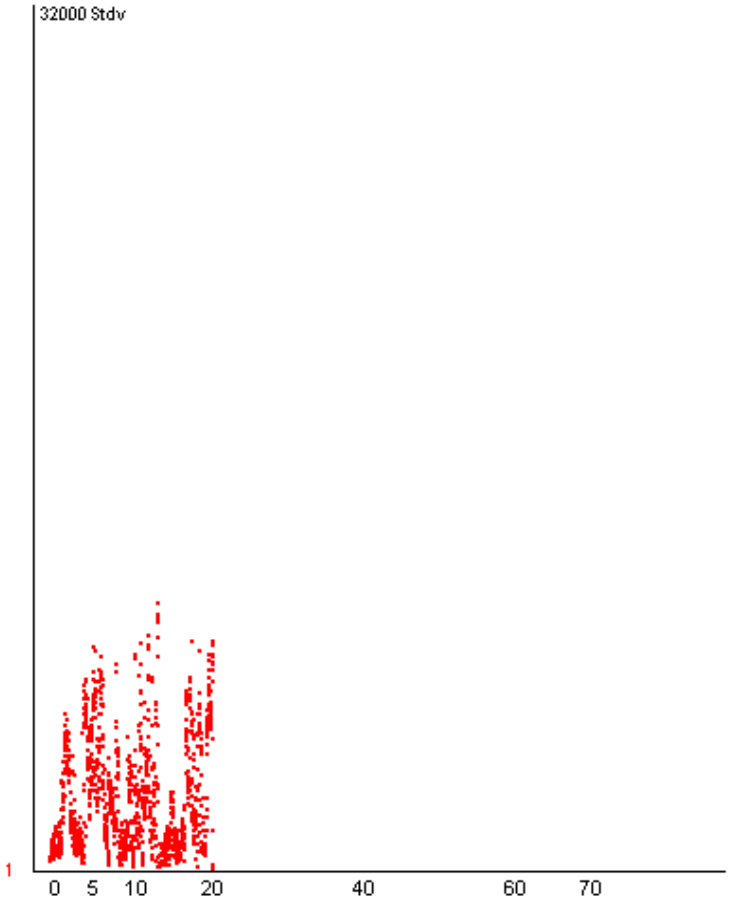
1 = E:\Noise\Cock E.S.P. - We Mean It This Time (full album).wav plot



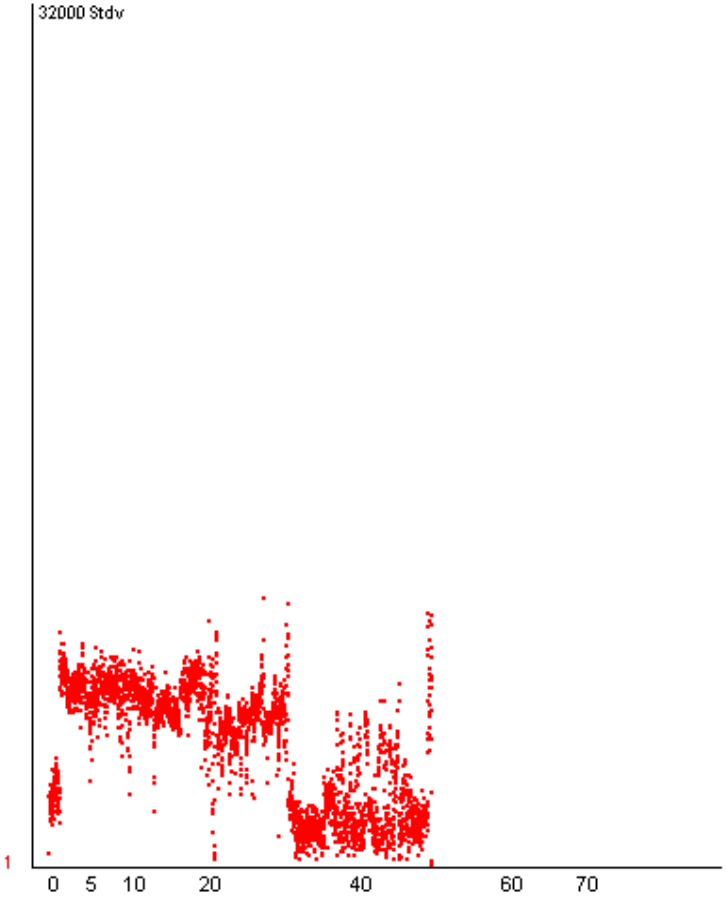
1 = ((((((((std\Ornette Coleman - The Shape Of Jazz To Come .wav plot



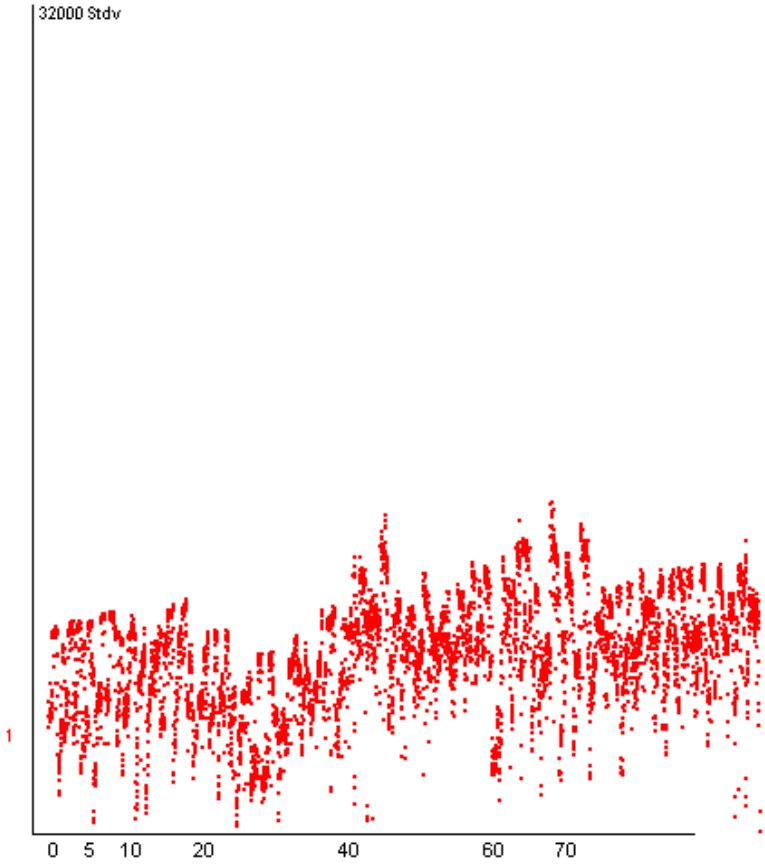
1 = ((std\Henry Cowell Symphony No.4 'Short Symphony' (1946).wav plot



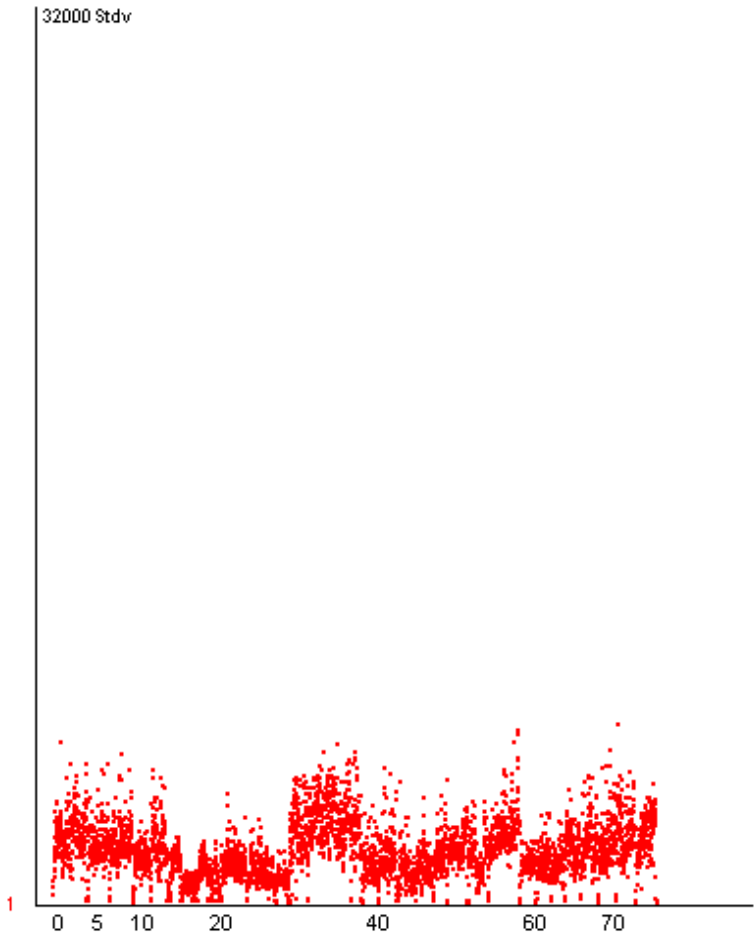
1 = ry Cowell Symphony No.11 'Seven Rituals of Music' (1954).wav plot



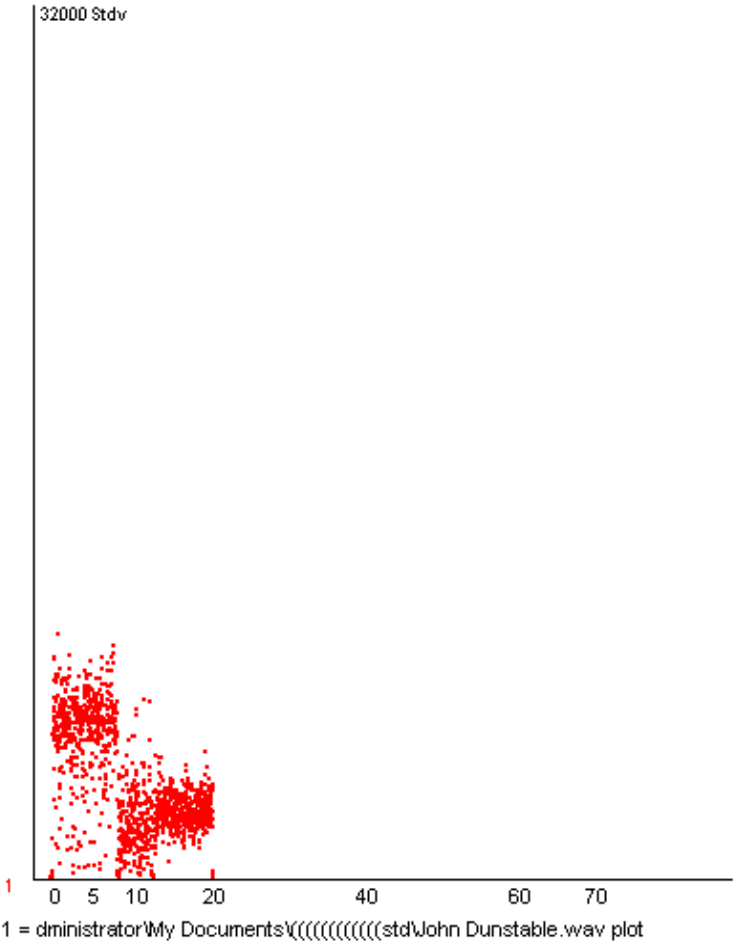
1 = ream - 1968 Farewell Concert -BBC-TV - Royal Albert Hall.wav plot

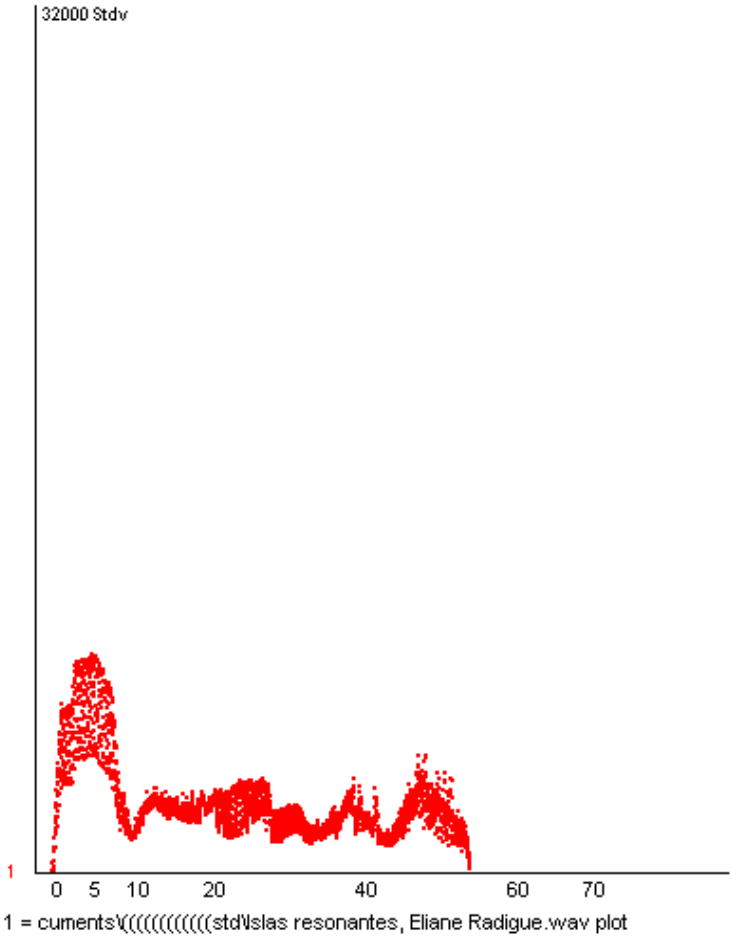


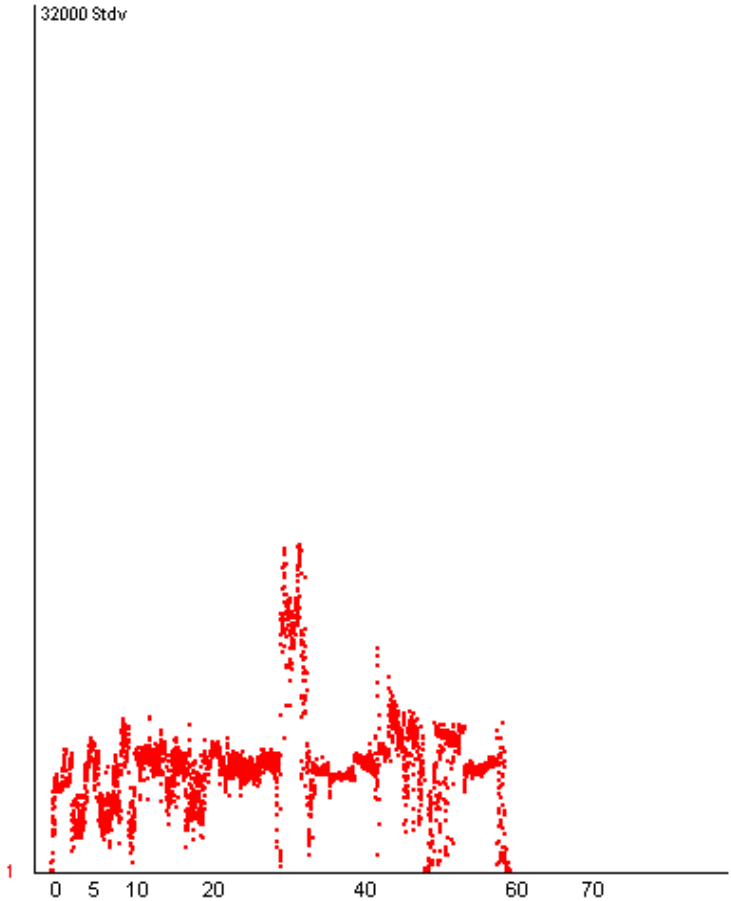
1 = ideo Mix - New EDM Dance Charts Songs - Club Music Remix.wav plot



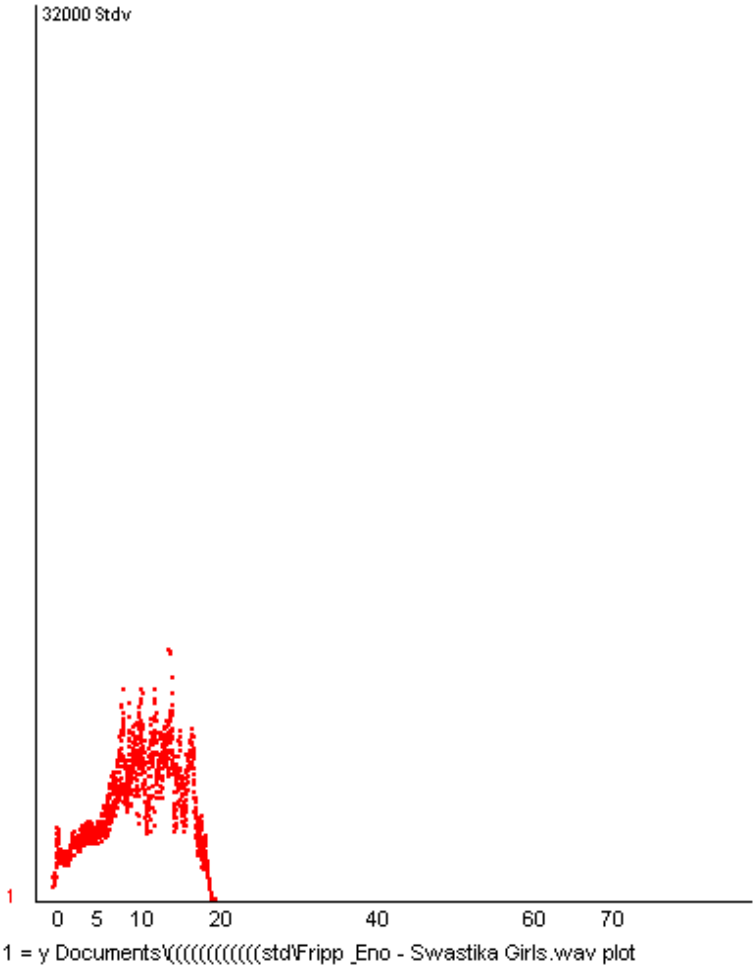
1 = nts\((((((((((((std\Philippus De Monte and The Habsburgs.wav plot

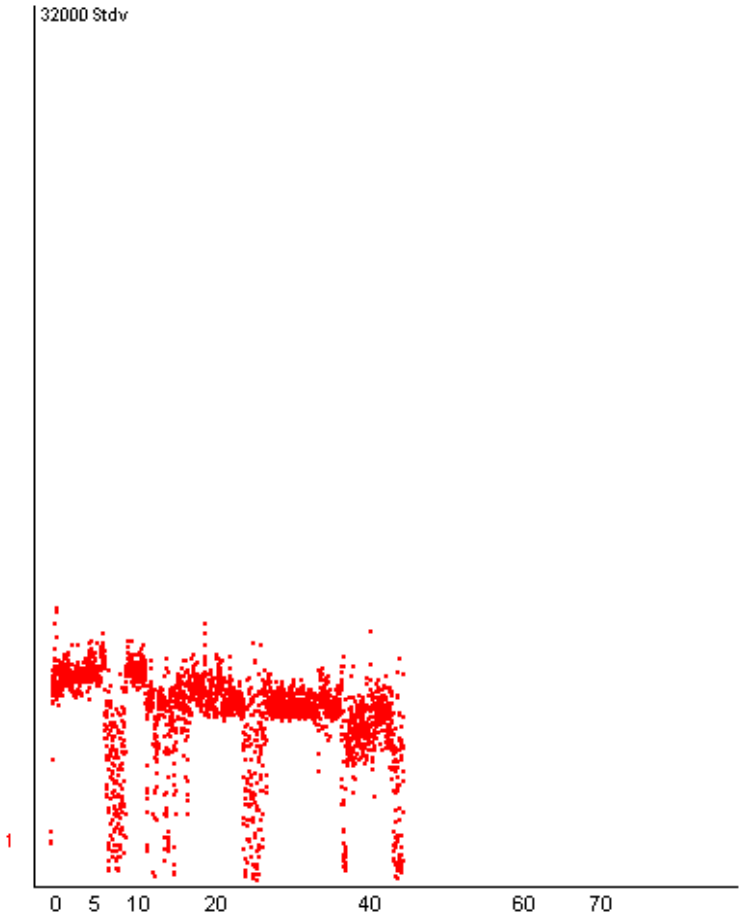

32000 Stdv
1
0
5
10
20
40
60
70
1 = dministrator\My Documents\((((((((((((std\John Dunstable.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = cuments\((((((((((((std\slas resonantes, Eliane Radigue.wav plot

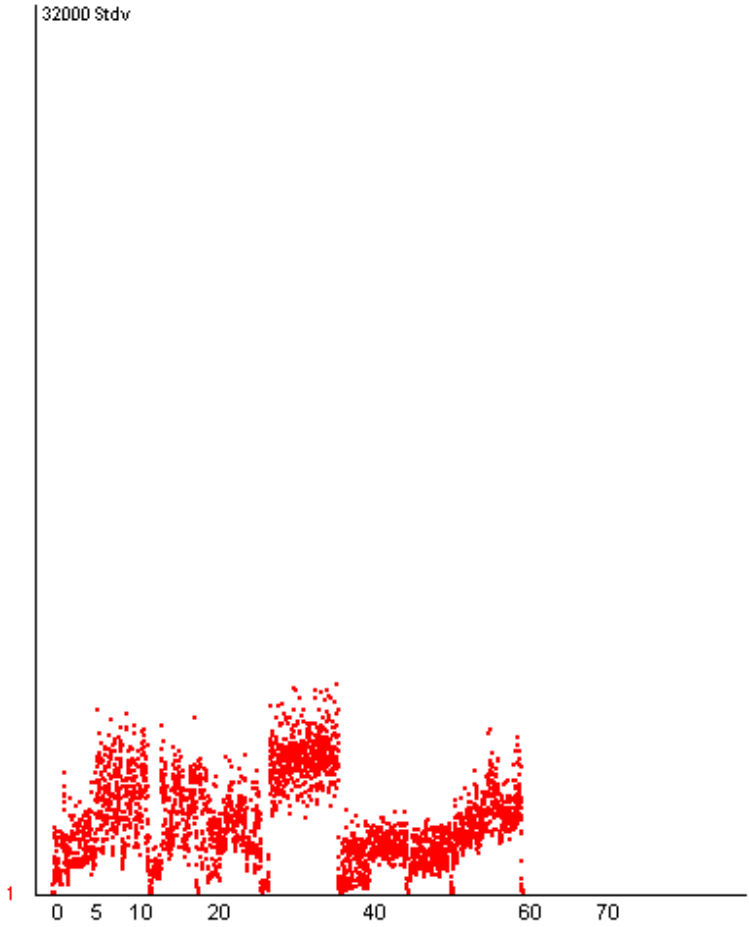


1 = se\Emil Beaulieau America's Greatest Living Noise Artist.wav plot

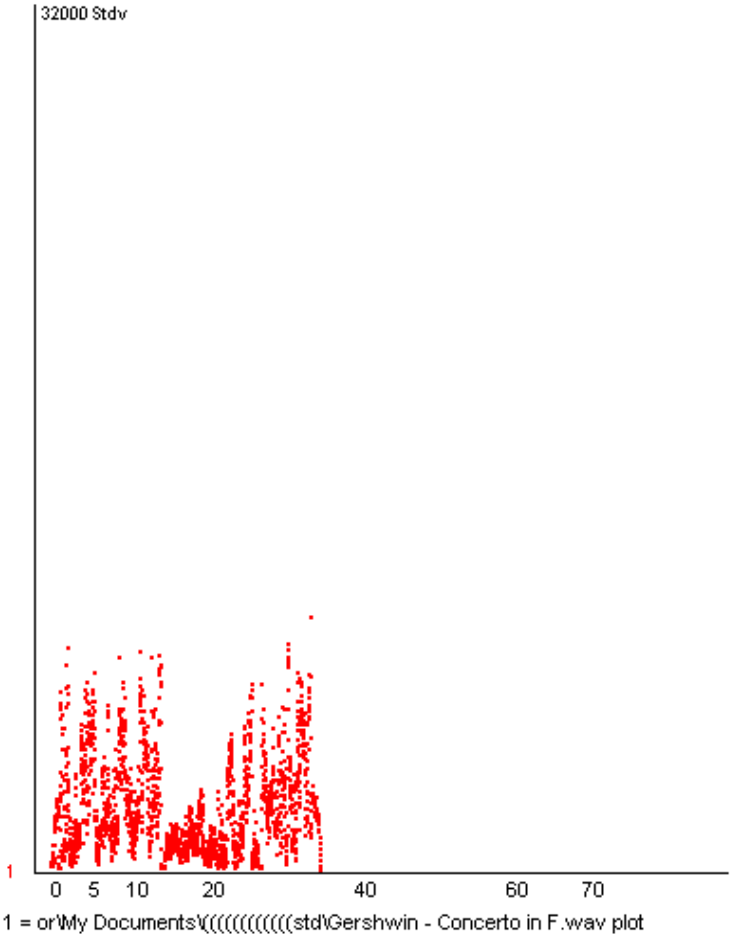

32000 Stdv
1
0
5
10
20
40
60
70
1 = y Documents\((((((((((((std\Fripp _Eno - Swastika Girls.wav plot

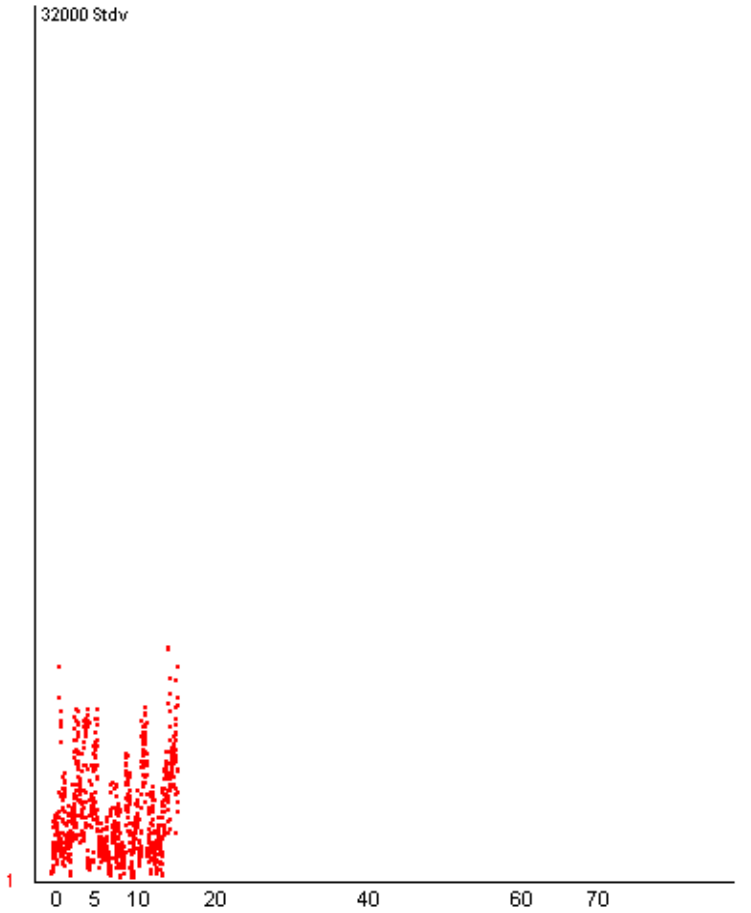


1 = E:\Noise\Fushitsush@ St John-at-Hackney 5-10-2012.wav plot

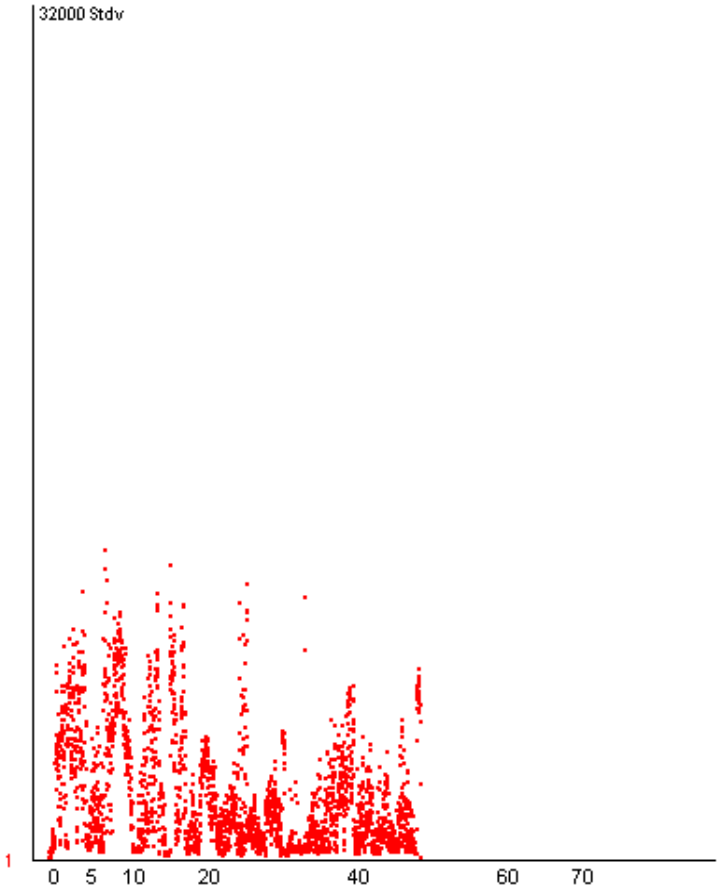


1 = Gamelan Music Of Bali (World Music Library) [Full Album].wav plot

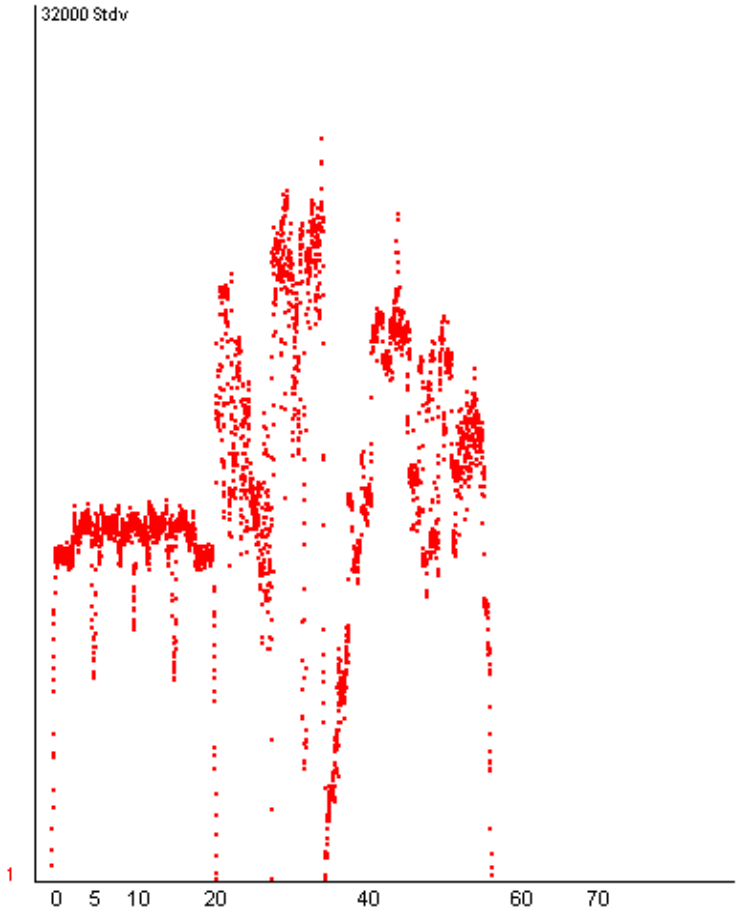

32000 Stdv
1
0
5
10
20
40
60
70
1 = or\My Documents\((((((((((((std\Gershwin - Concerto in F.wav plot



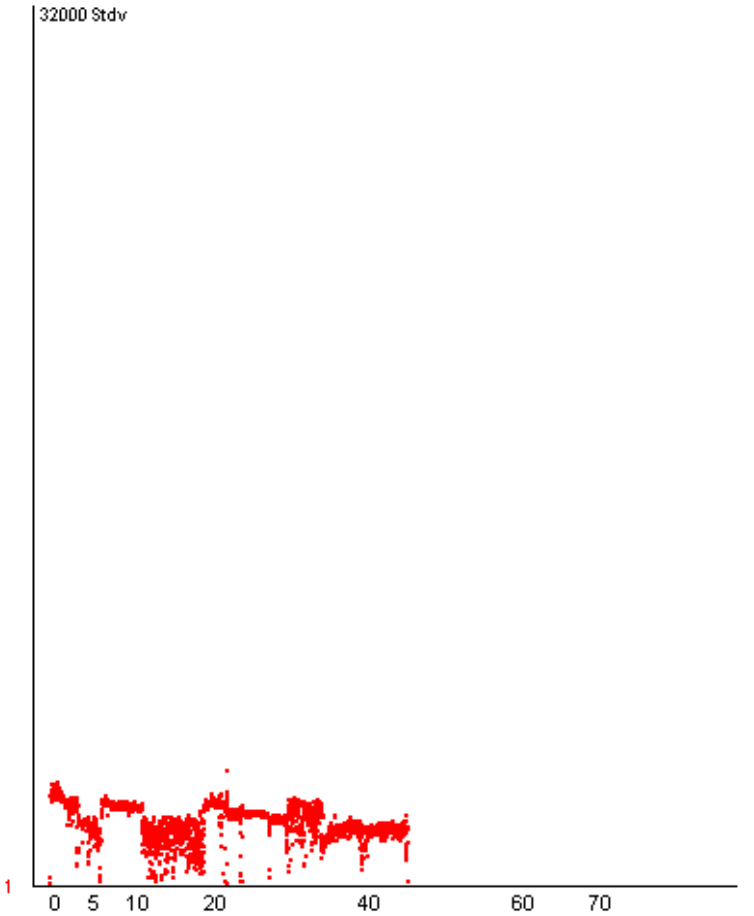
1 = ments\((((((((((((std\Gershwin Rhapsody In Blue Gershwin.wav plot



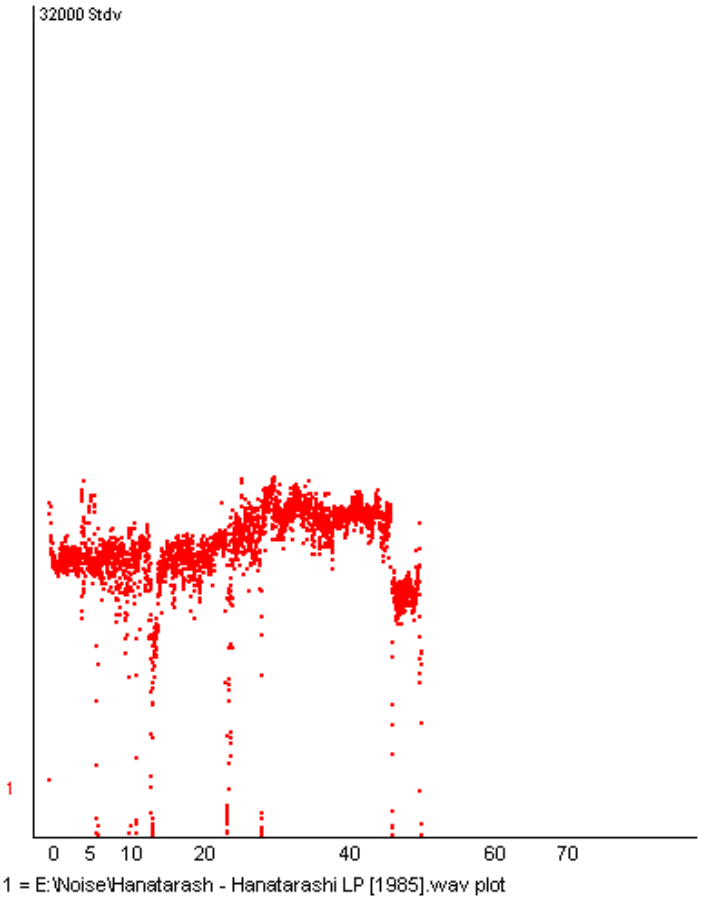
1 = ot Musical Improvisation in the Marlene Dumas Exhibition.wav plot

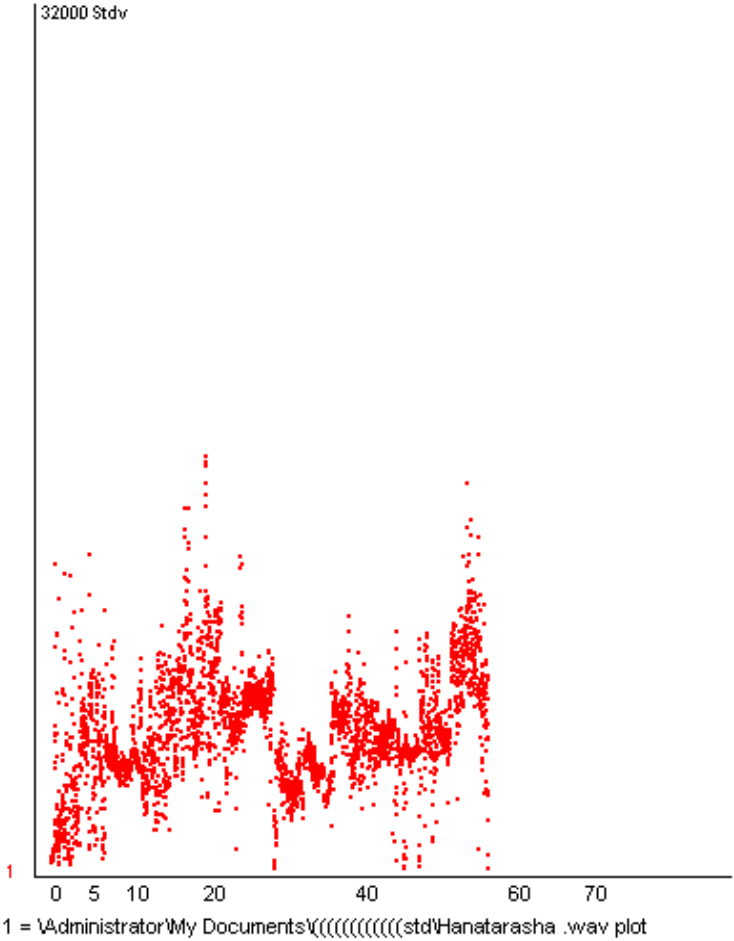


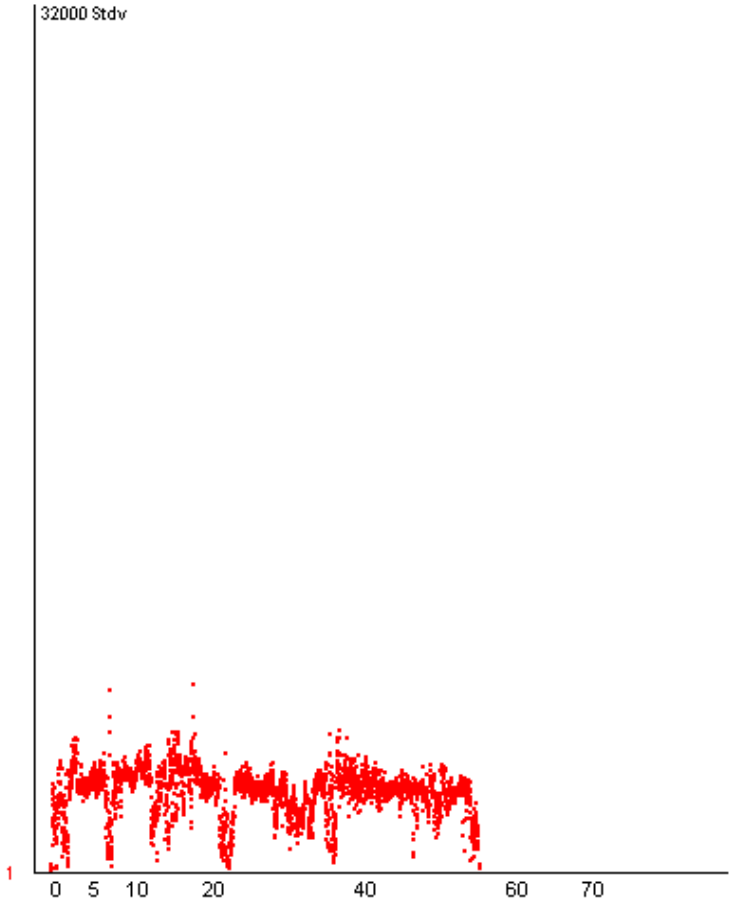
1 = GX Jupitter Larsen - God PussyUntitled [FULL TAPE ALBUM].wav plot



1 = E:\Noise\Hanatarash - Hanatarash 3 (Full).wav plot

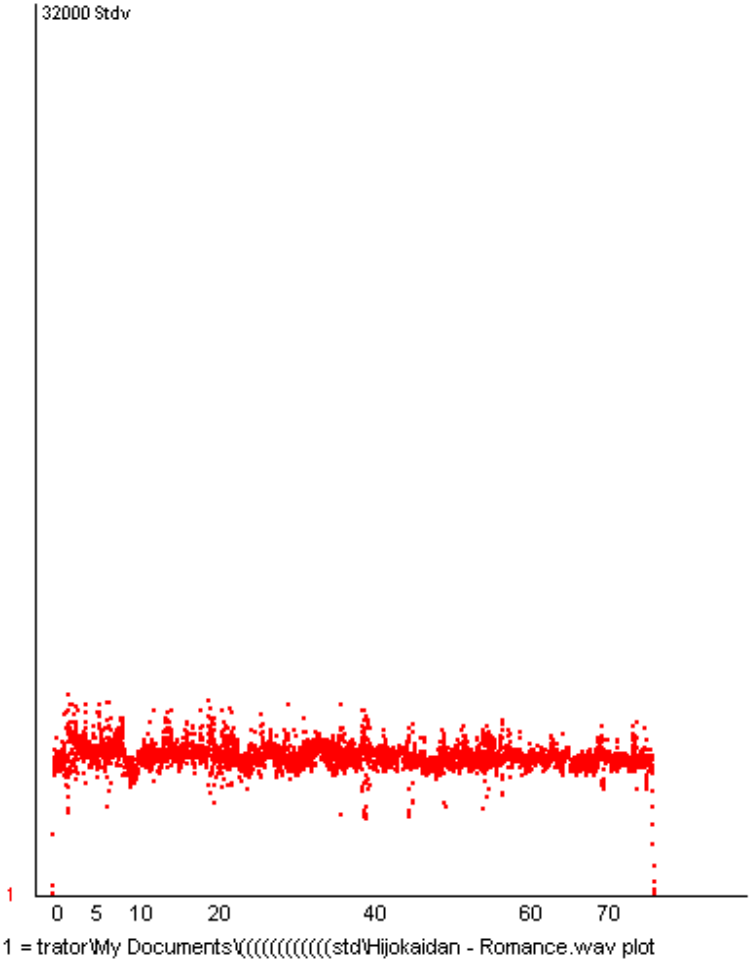

32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\Hanatarash - Hanatarashi LP [1985].wav plot

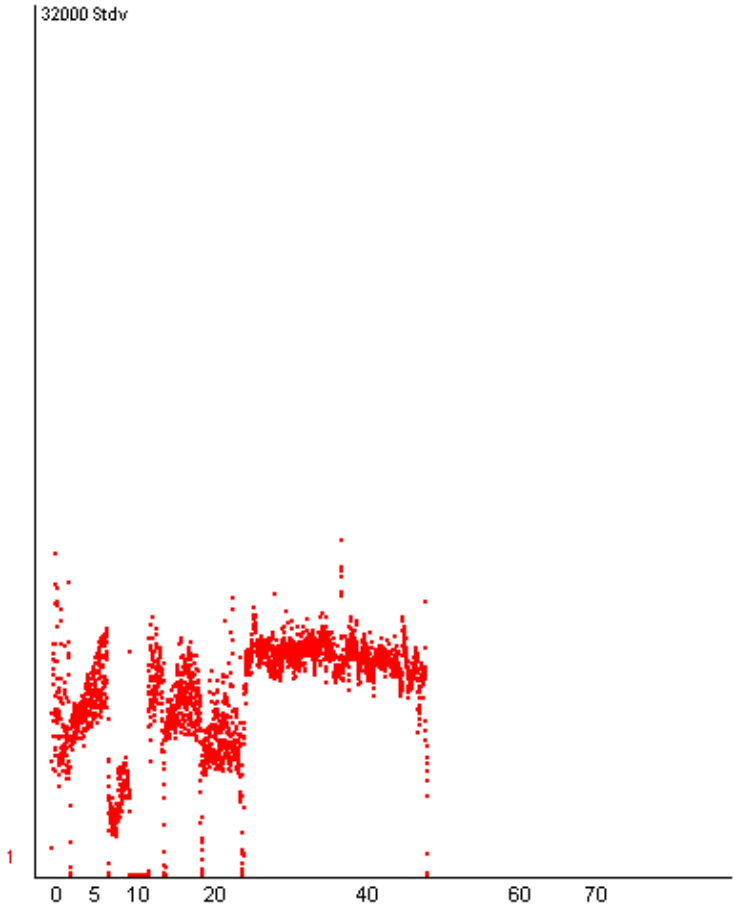

32000 Stdv
1
0
5
10
20
40
60
70
1 = \Administrator\My Documents\((((((((((((std\Hanatarasha .wav plot



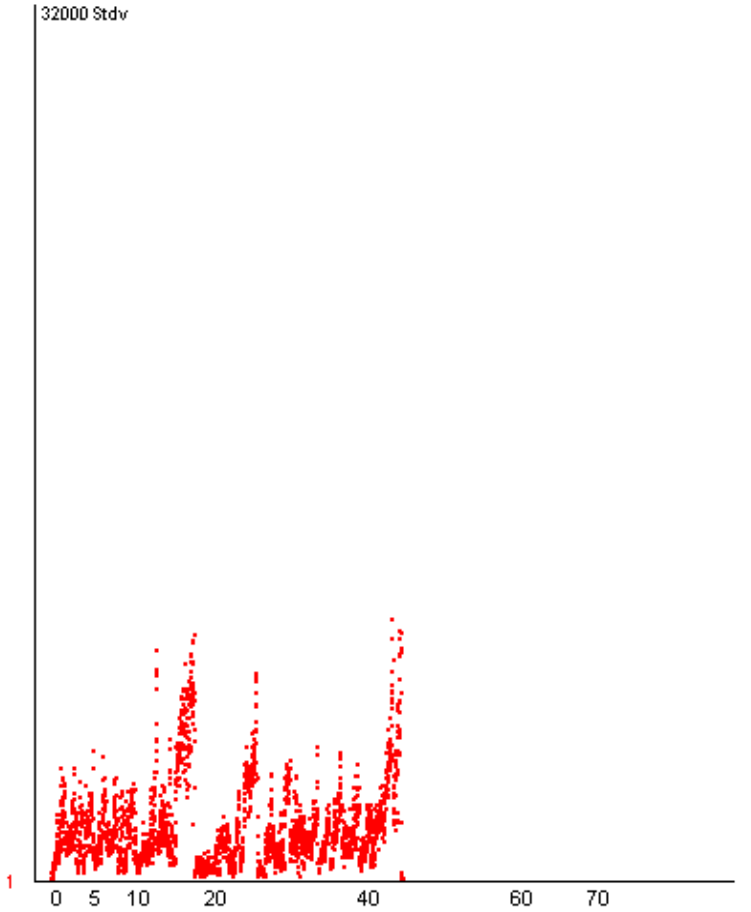
1 = cuments\(((((((((((std\Jimi Hendrix Live Stockholm 1969.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = \My Documents\((((((((((((std\Hijokaidan King of Noise.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = trator\My Documents\((((((((((((std\Hijokaidan - Romance.wav plot

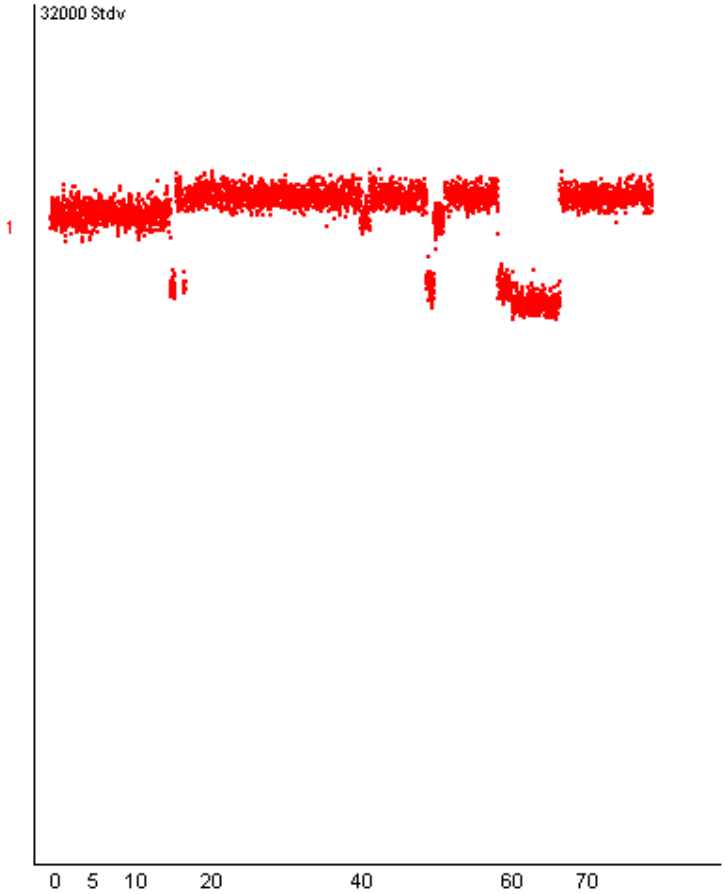


1 = E:\Noise\Hijokaidan - Viva Angel [Full album].wav plot

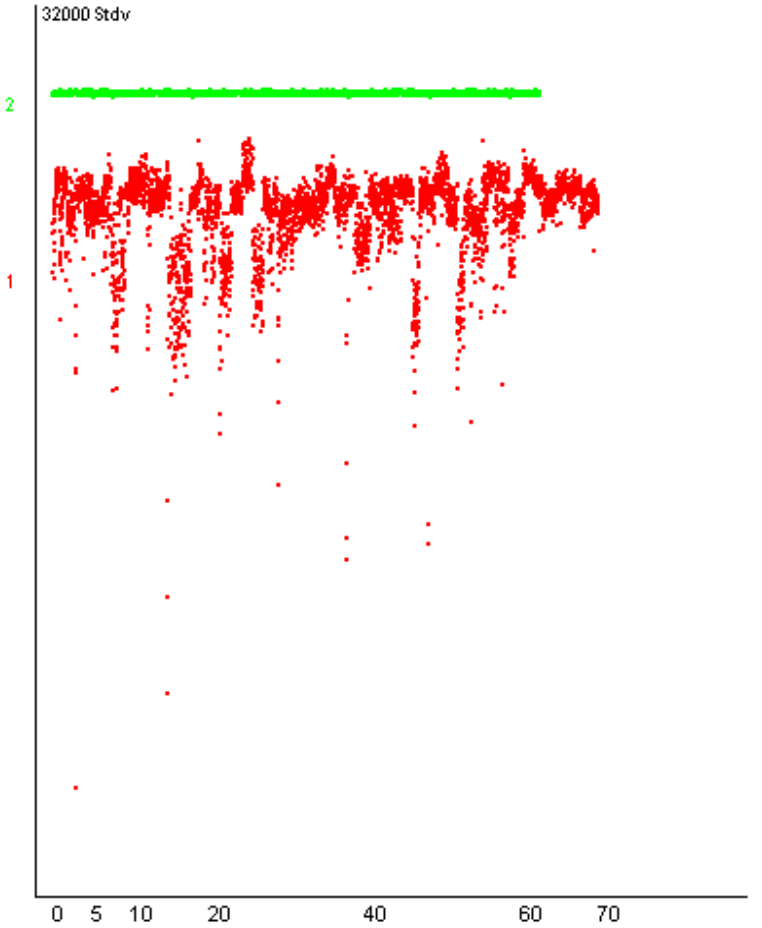


1 = (((((((((std\Charles Ives 'Symphony No 1' James Sinclair.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = tor\My Documents\((((((((((((std\28_05_2013_09_49_52-HNV.wav plot



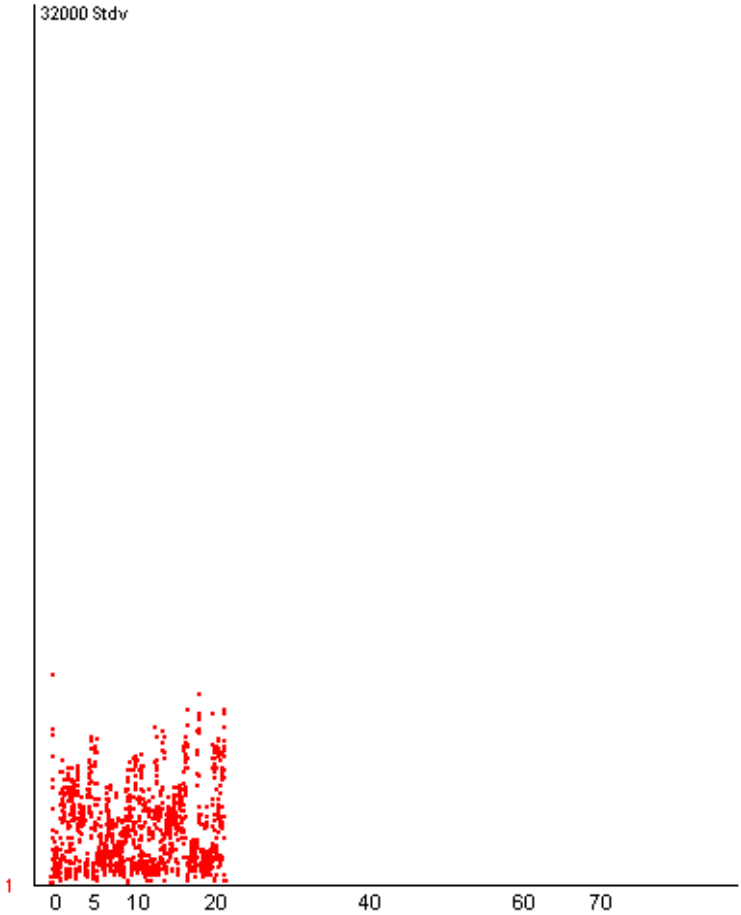
1 = My Documents\((((((((((((std\msx27x04x2014x08x14x22x1-HN.wav plot



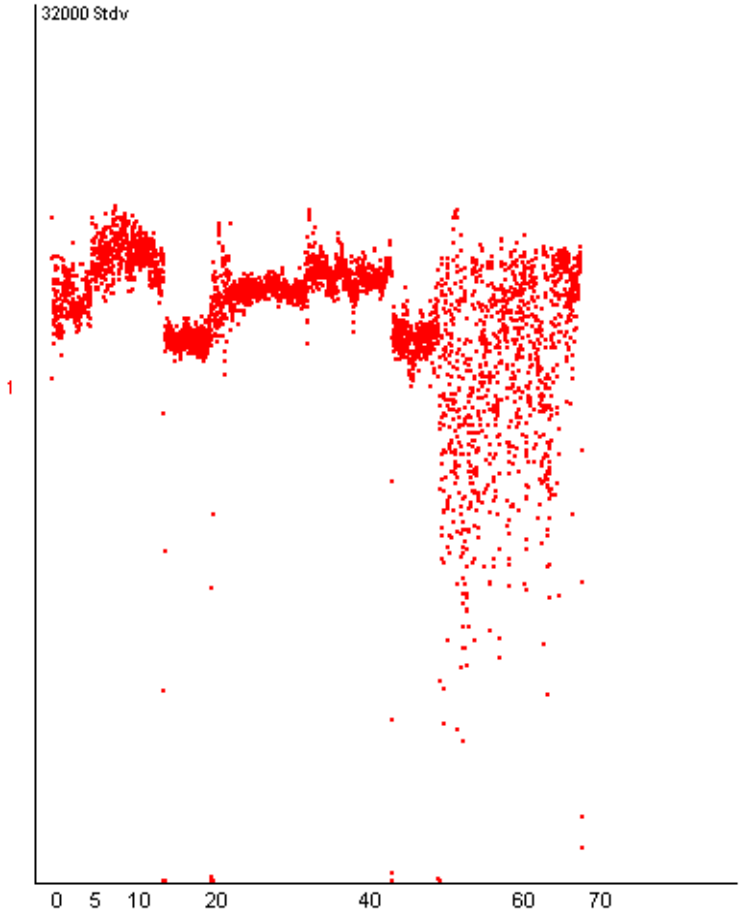
1 = ings\Administrator\My Documents\((((((((((((std\badmusic.wav plot

2 = ings\Administrator\My Documents\((((((((((((std\hnwjava1.wav plot

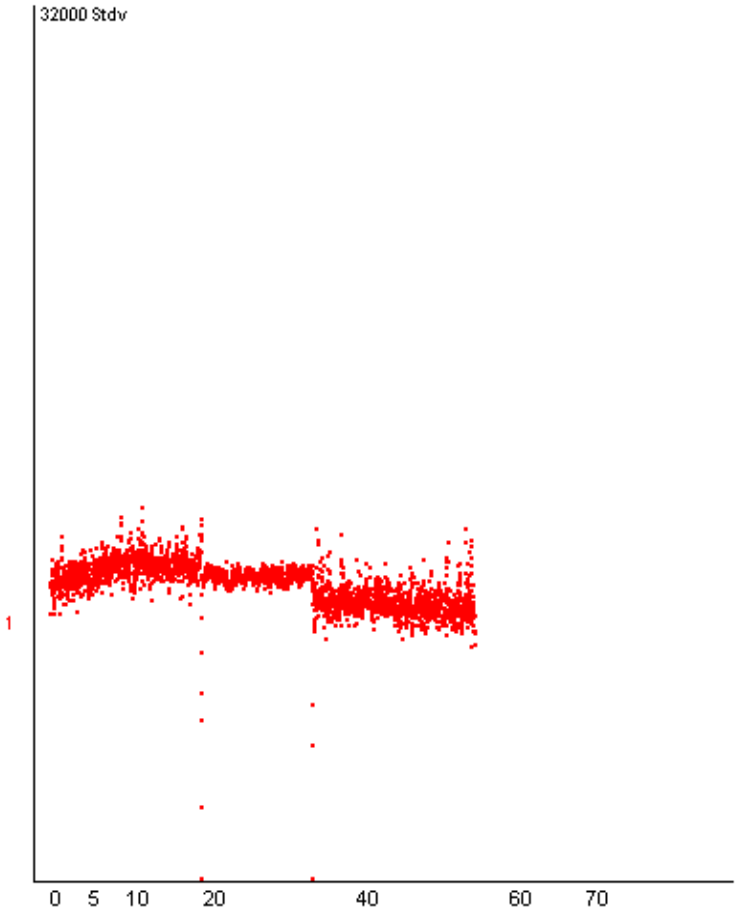
1 = ony No.6 in D major H.1 'Le Matin', Sir Neville Marriner.wav plot



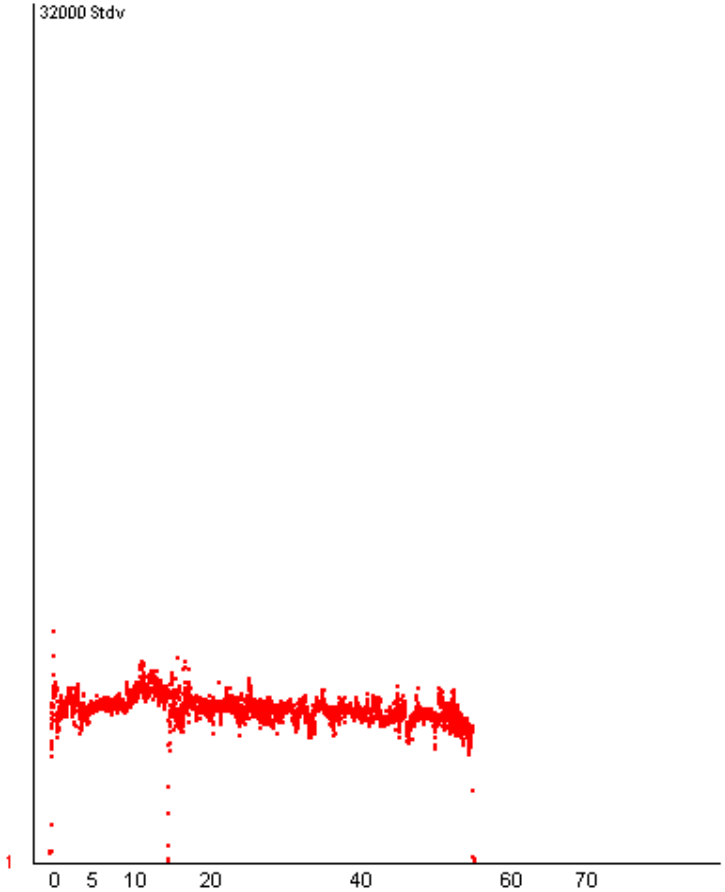
1 = o.60 in C major H.1 'Il distratto', Sir Neville Marriner.wav plot



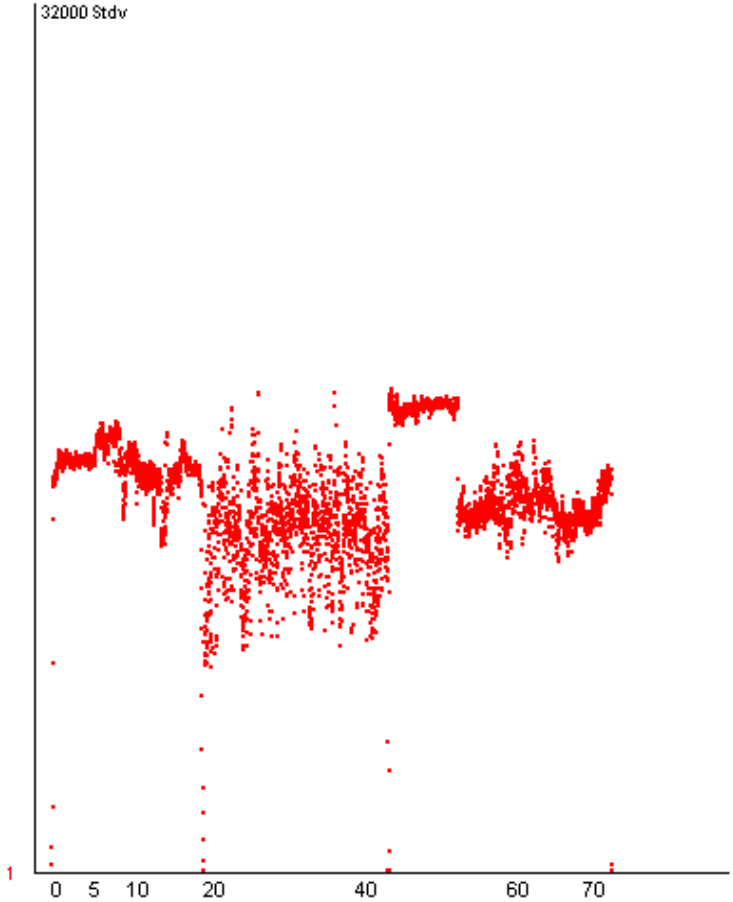
1 = E:\Noise\Incapacitants - Default Standard (Full Album).wav plot



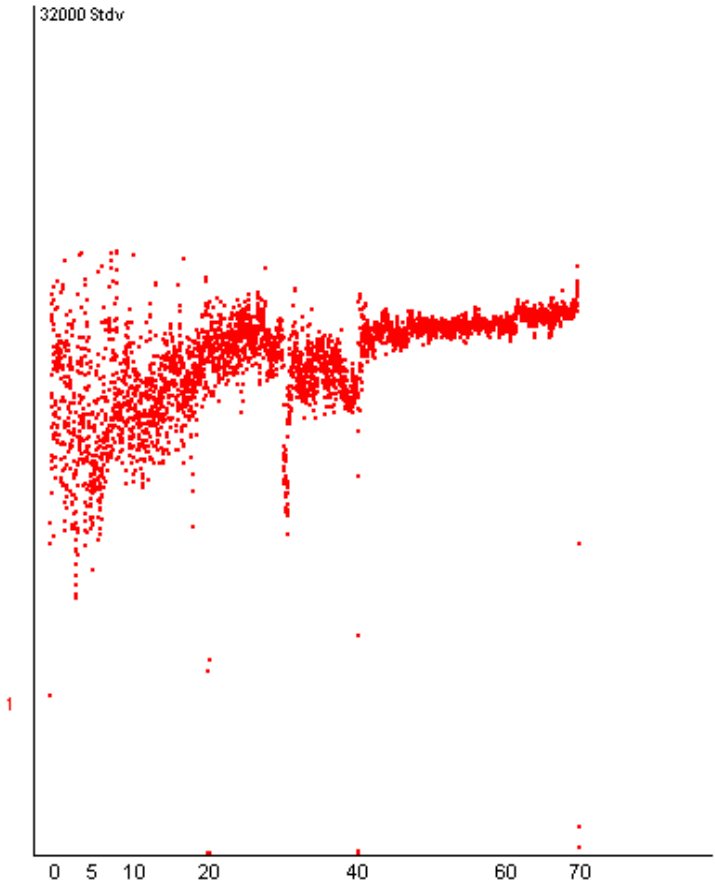
1 = se\Incapacitants - Eat! Meat!! Manifesto!!! (Full Album).wav plot



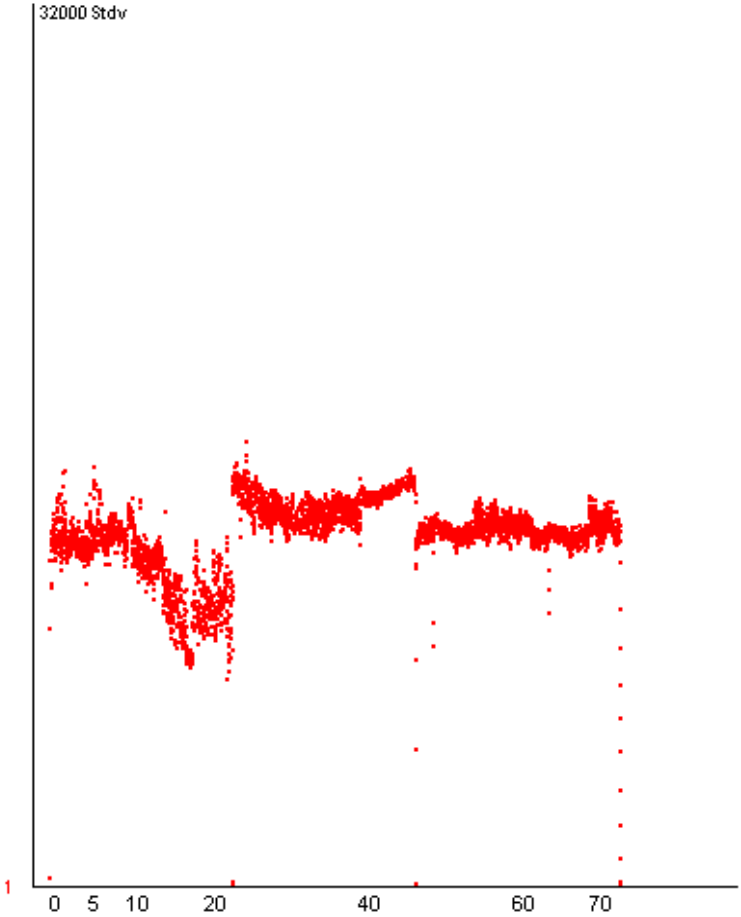
1 = E:\Noise\Incapacitants - Fabrication (Full Album).wav plot



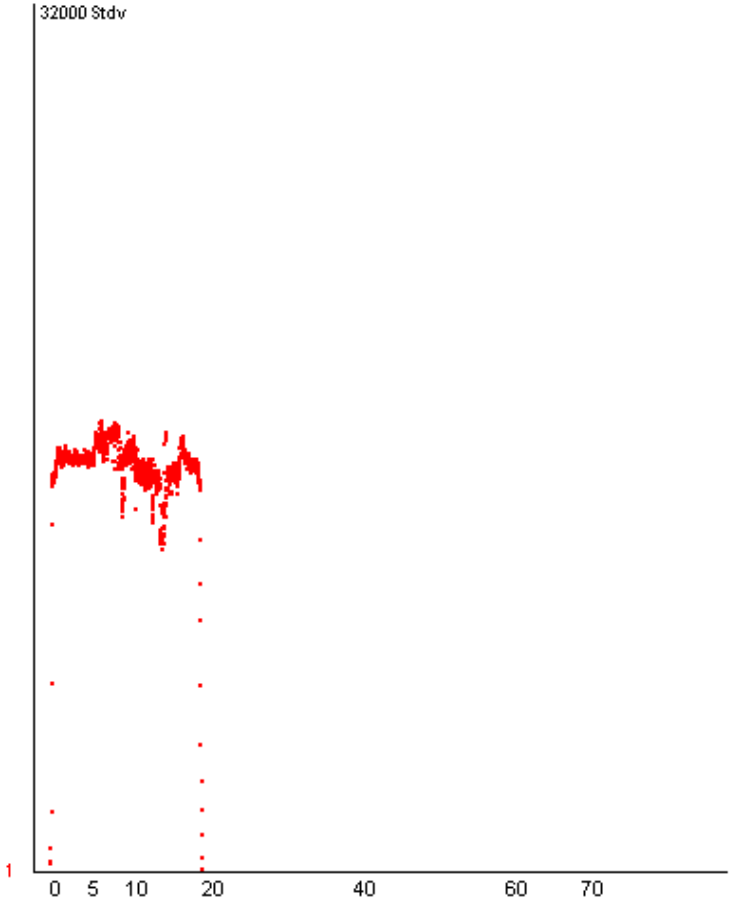
1 = ise\Incapacitants - Ministry Of Foolishness (Full Album).wav plot



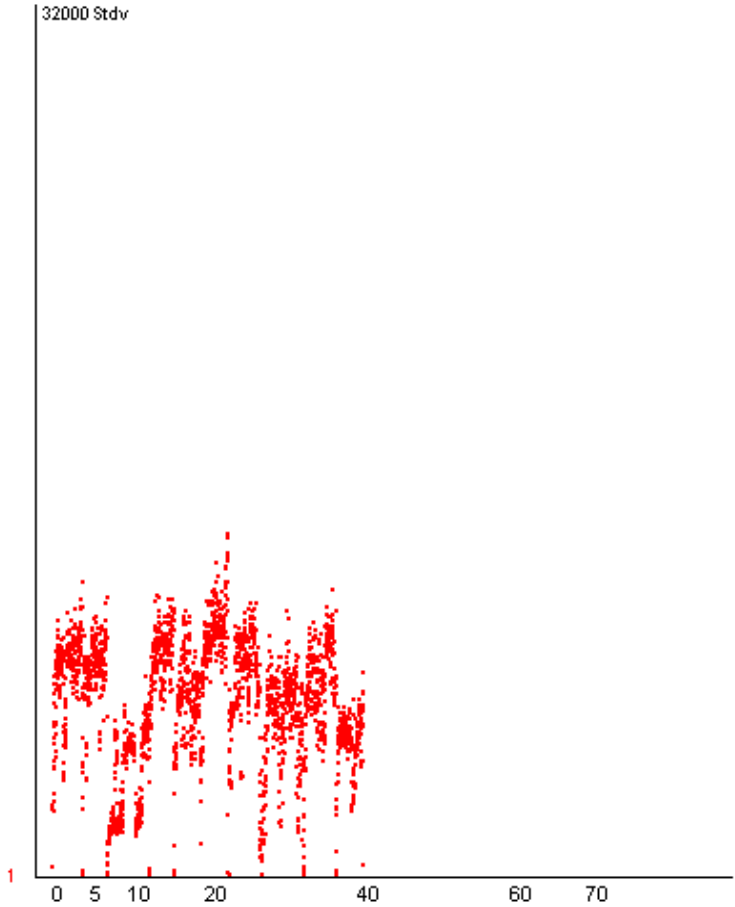
1 = Noise\Incapacitants - New Movements in CMPD (Full Album).wav plot



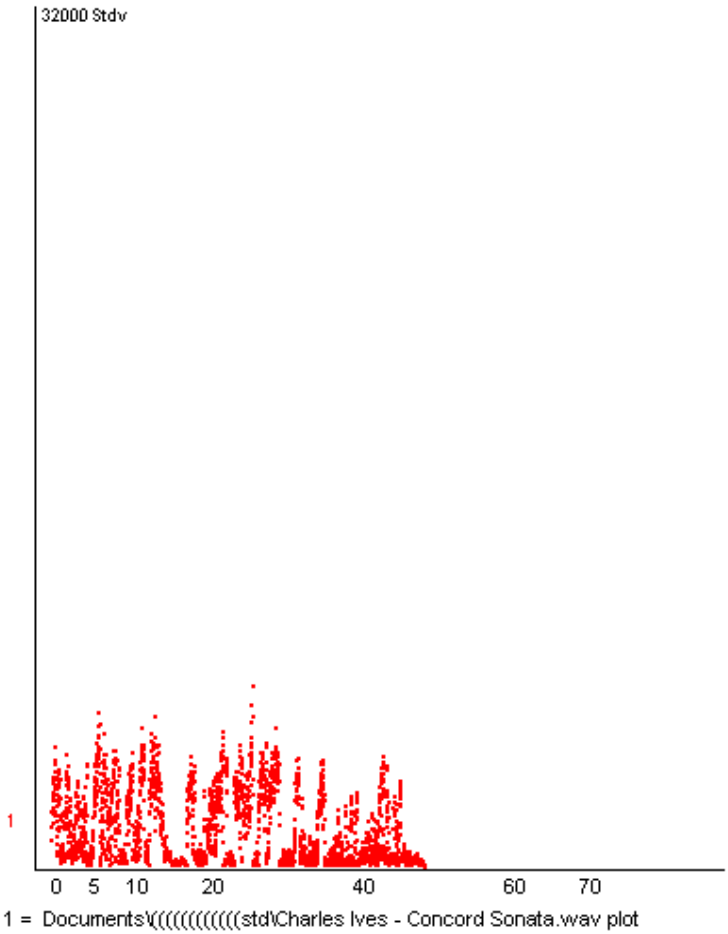
1 = trator\My Documents\((((((((((((std\Incapacitants - Repo.wav plot

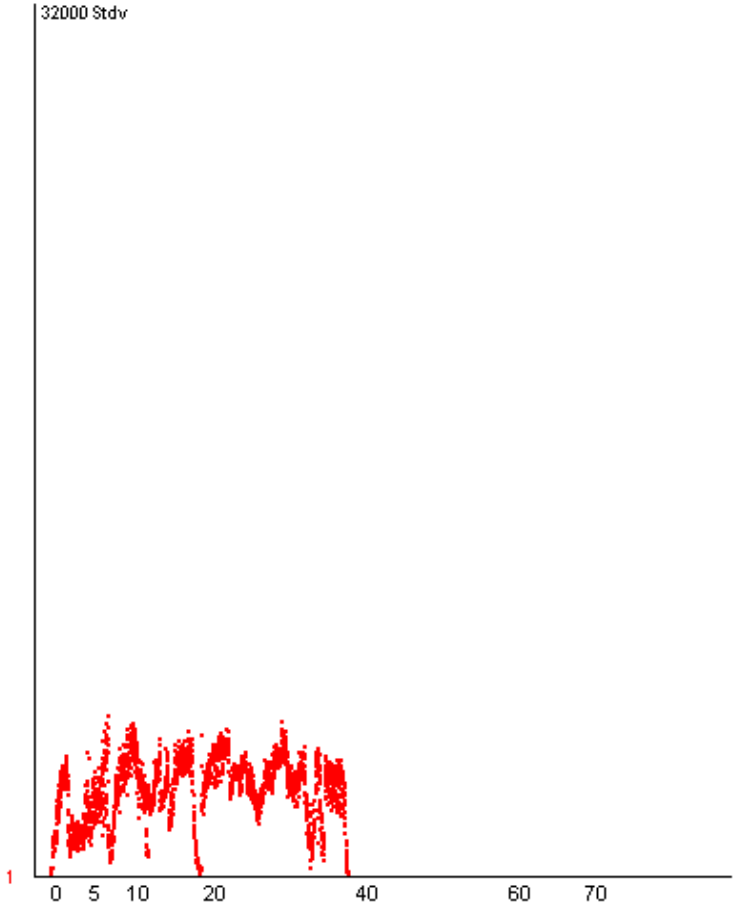


1 = E:\Noise\Incapacitants - Stone River.wav plot

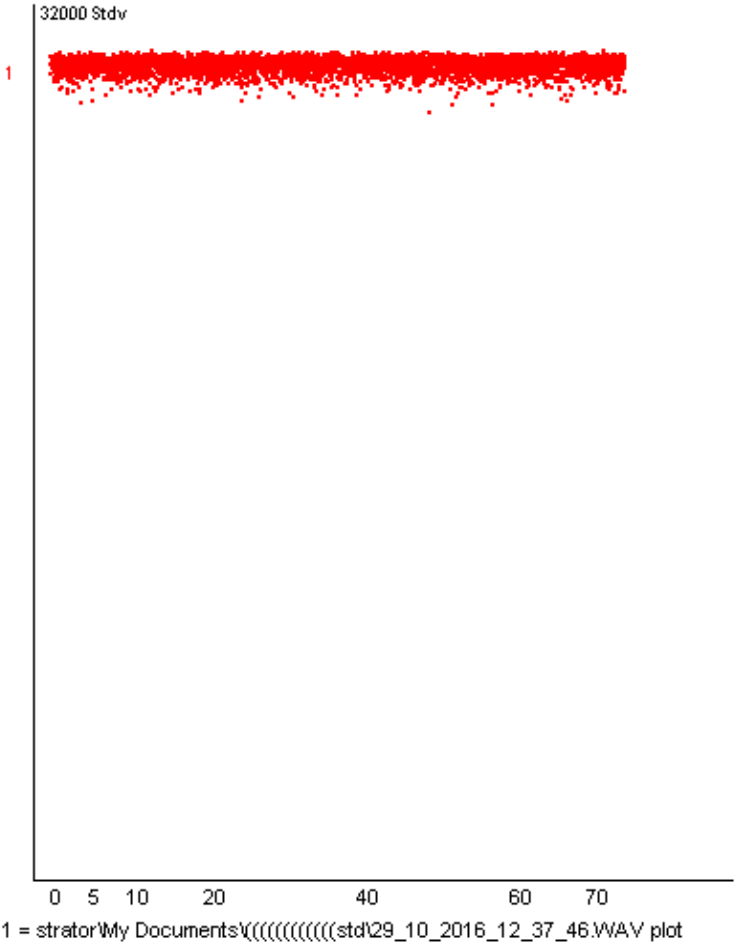


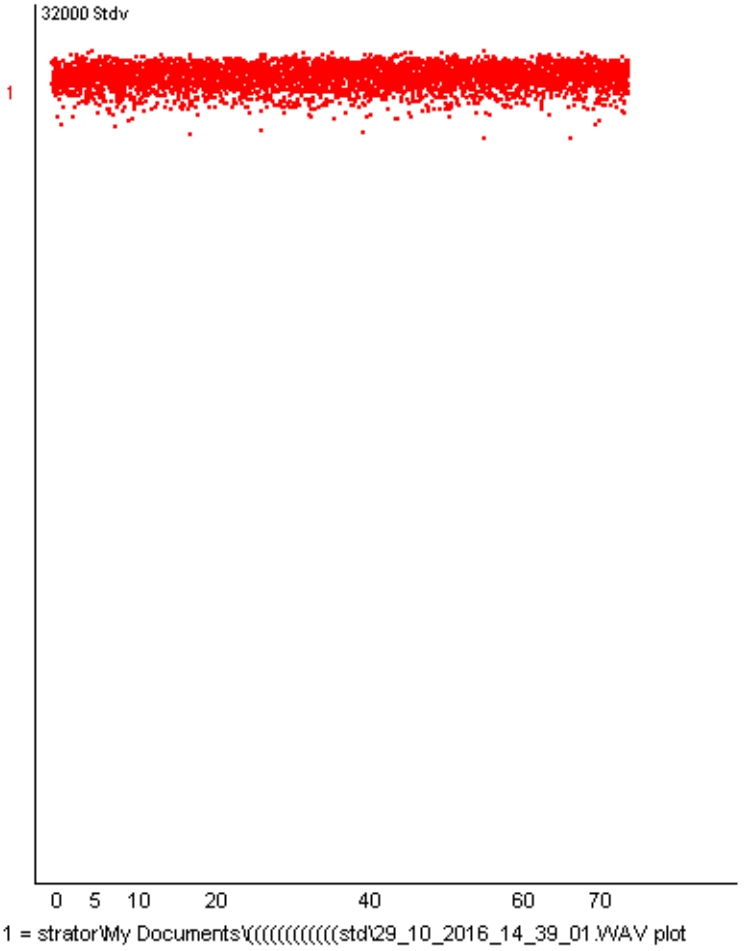
1 = ((((((((((std\Iron Maiden Iron Maiden (1980 Full Album).wav plot

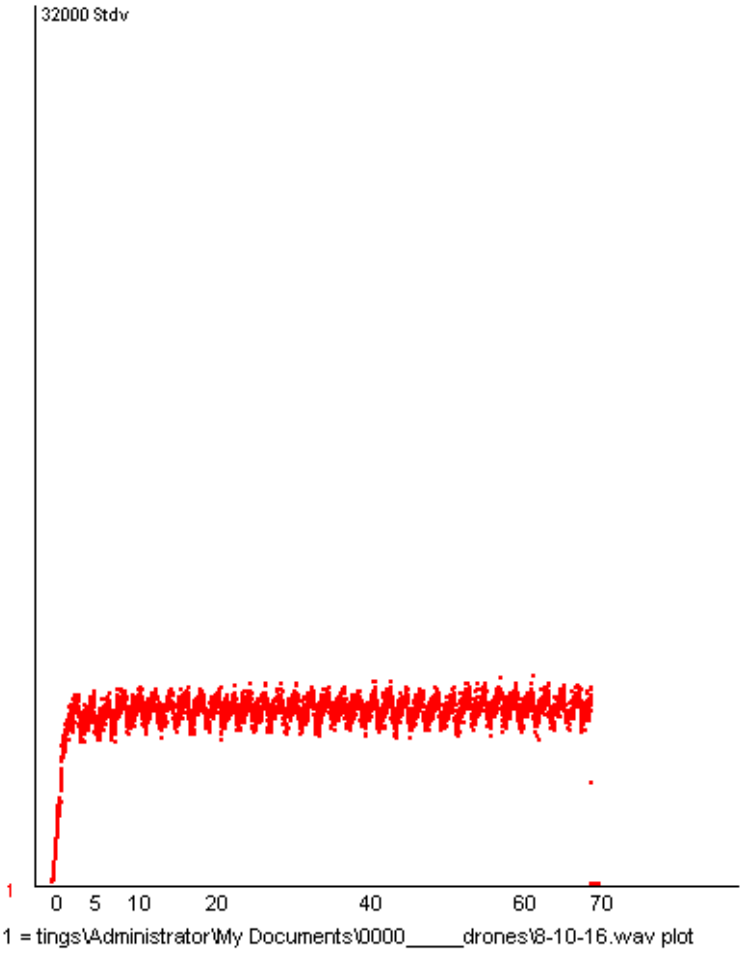

32000 Stdv
1
0
5
10
20
40
60
70
1 = Documents\((((((((((((std\Charles Ives - Concord Sonata.wav plot

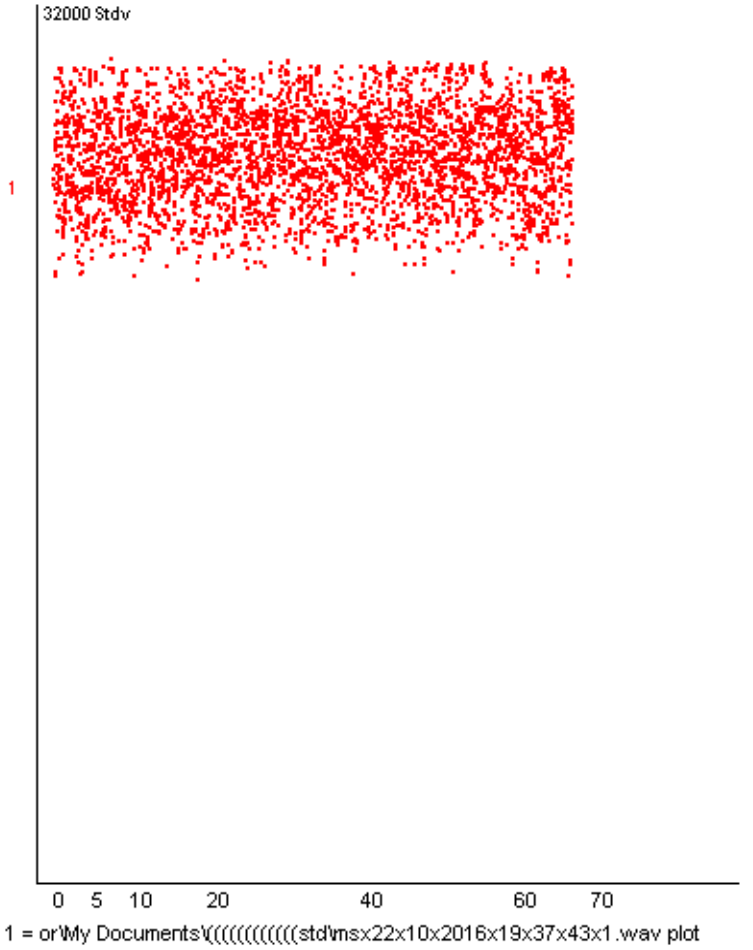


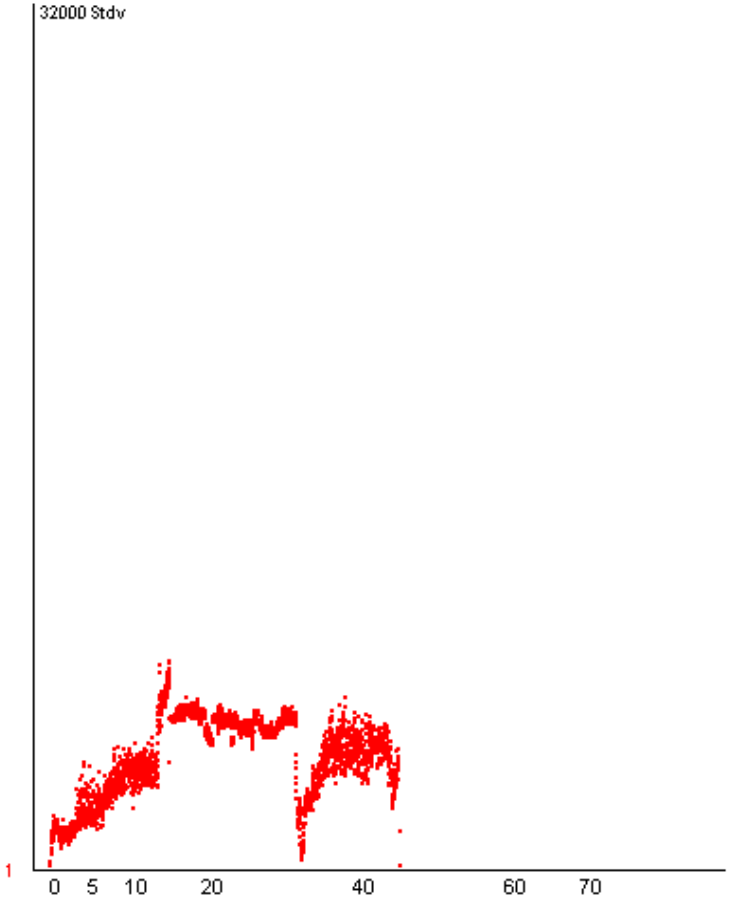
1 = y Documents\(((((((((((std\Jean Michel Jarre - Equinoxe.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = strator\My Documents\((((((((((((std\29_10_2016_12_37_46.WAV plot

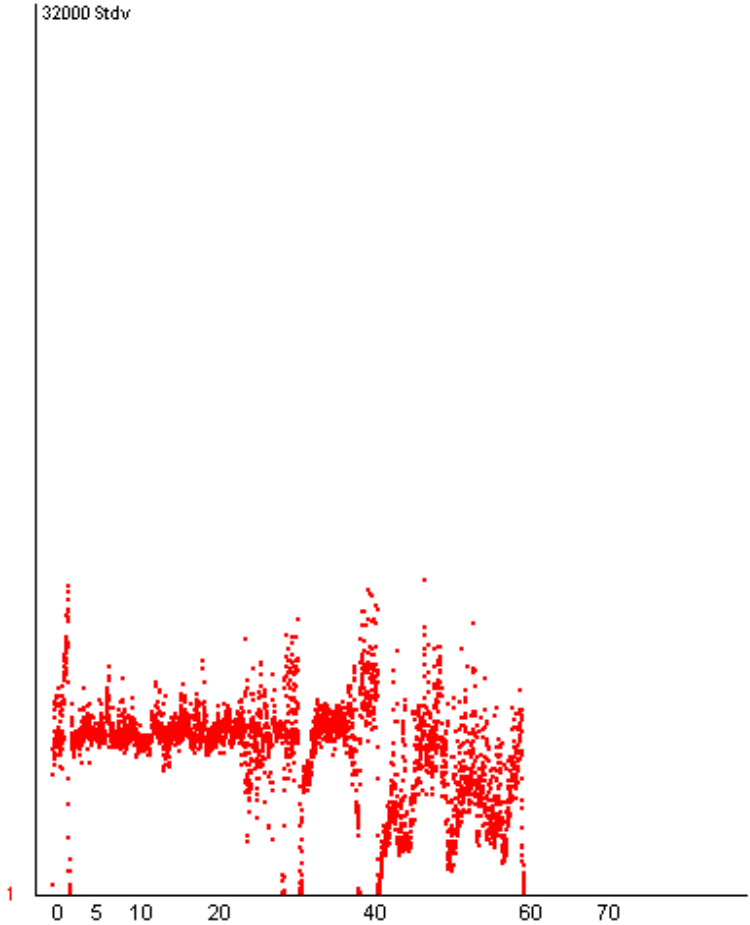

32000 Stdv
1
0
5
10
20
40
60
70
1 = strator\My Documents\((((((((((((std\29_10_2016_14_39_01.WAV plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = tings\Administrator\My Documents\0000_____drones\8-10-16.wav plot

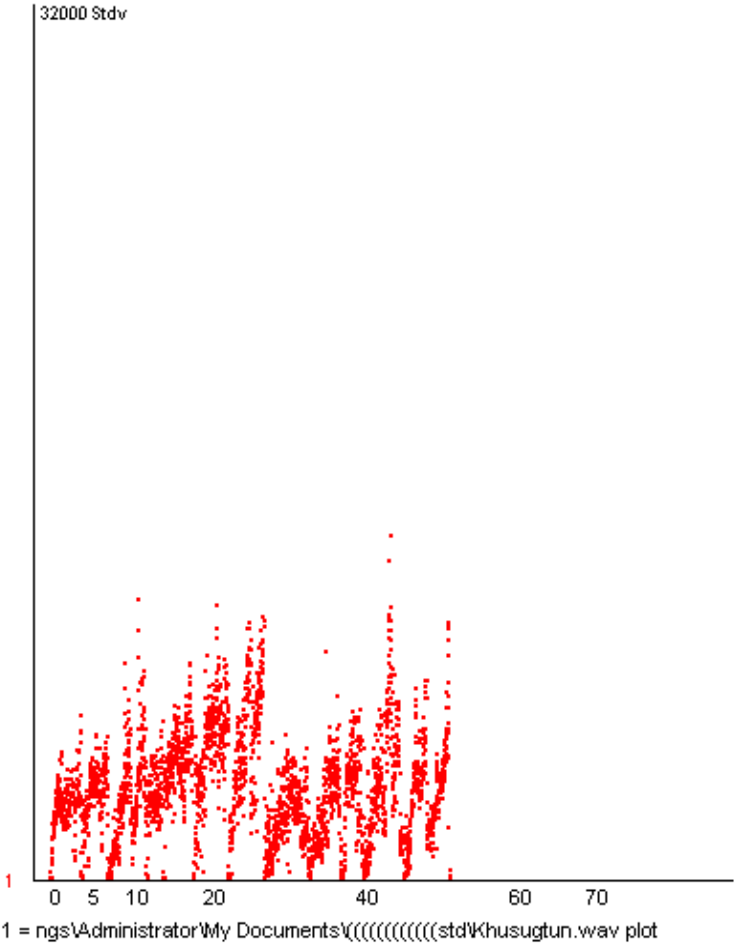

32000 Stdv
1
0
5
10
20
40
60
70
1 = or\My Documents\((((((((((((std\msx22x10x2016x19x37x43x1.wav plot

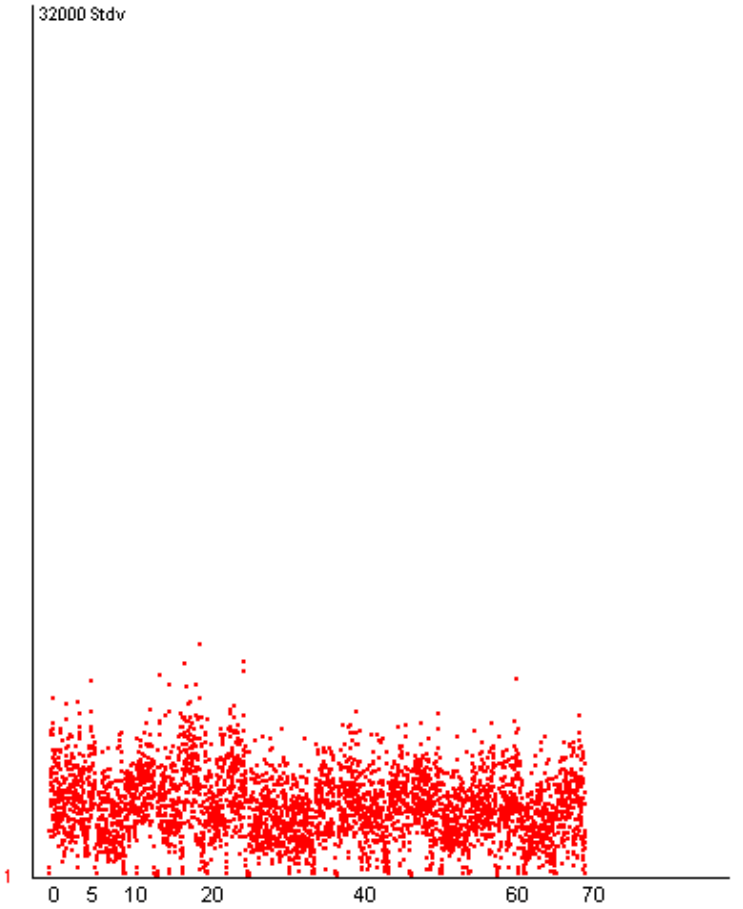


1 = E:\Noise\K.K. Null- Terminal Beach 1996.wav plot

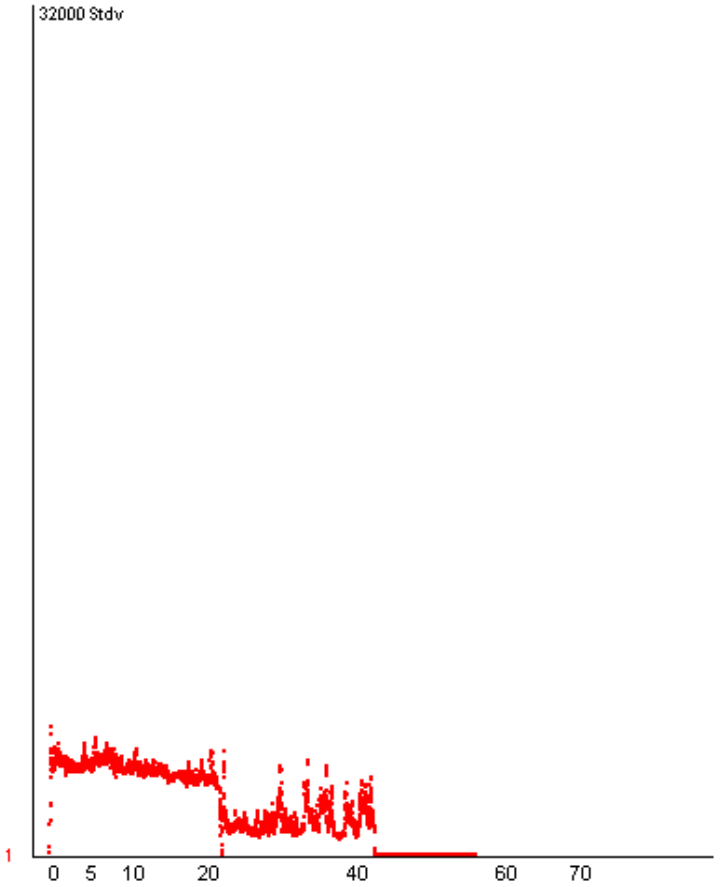


1 = Guy Lohnes, Ichiro Agata, Jim O'Rourke - Guitar Organism.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = ngs\Administrator\My Documents\((((((((((((std\Khusugtun.wav plot

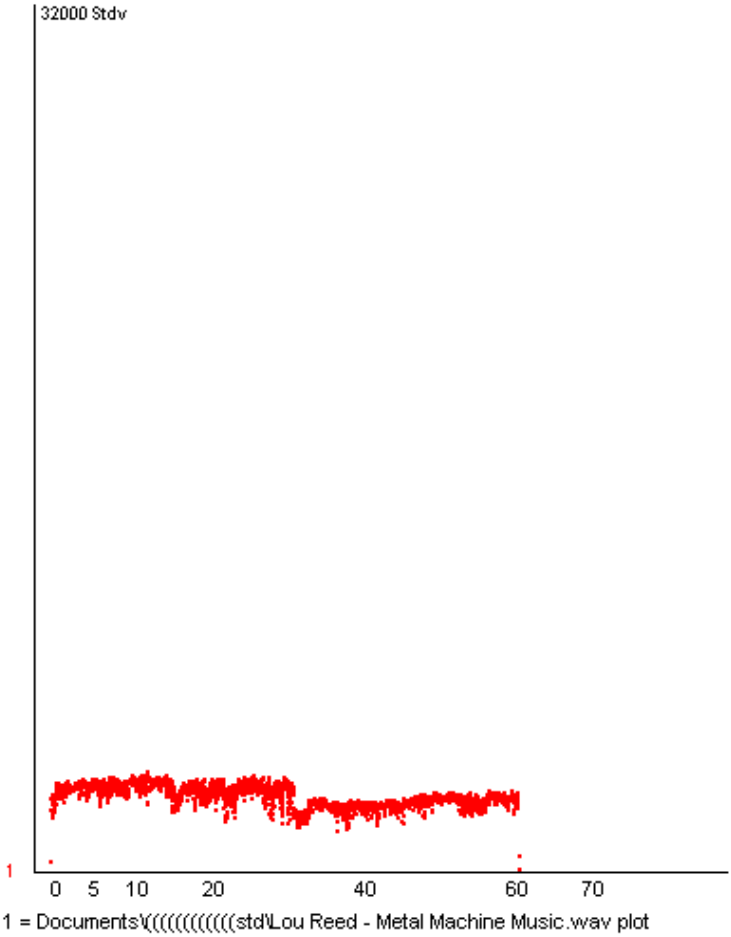


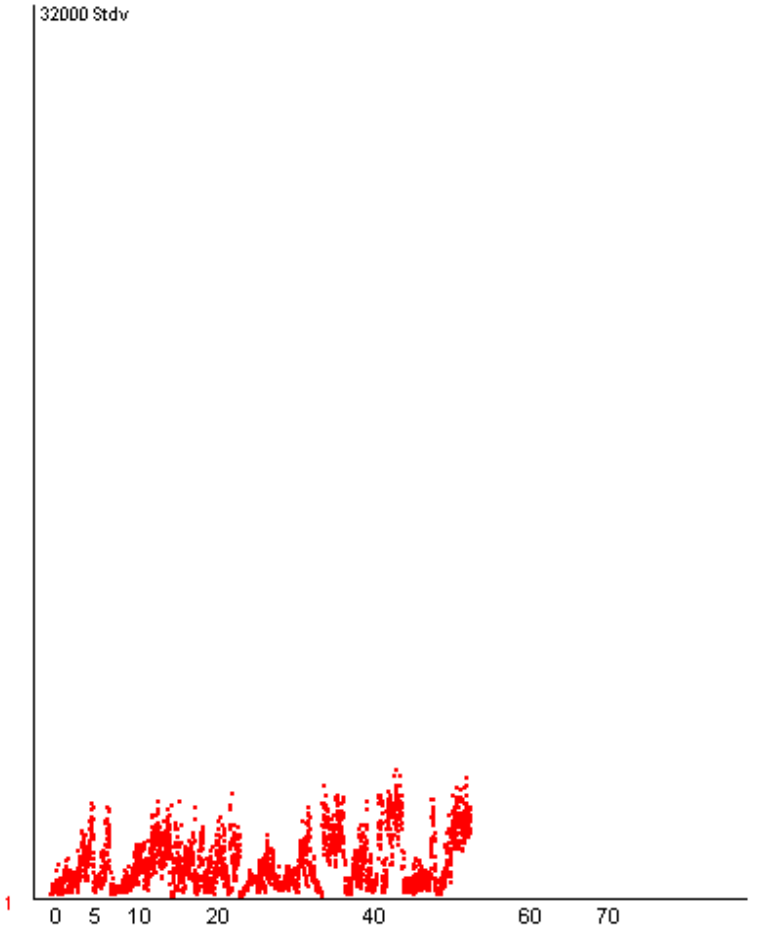
1 = (((((std\Jakob Klein (1688-1748) Sonatas for Violoncello.wav plot



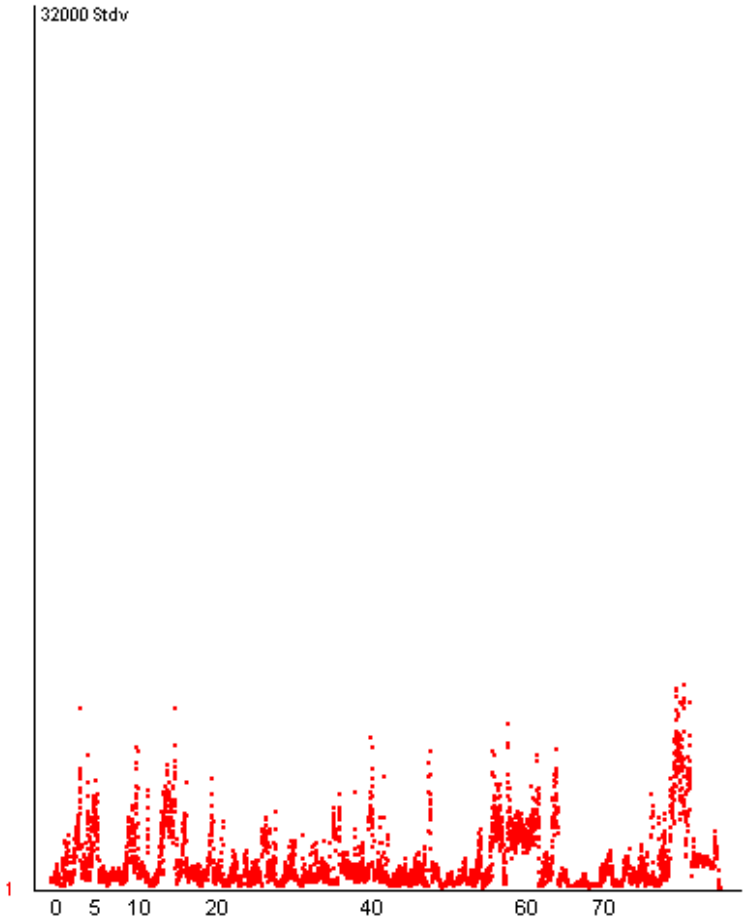
1 = (std\La Monte Young - Marian ZazeelaThe Black Album1969 .wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = \((((((((((((std\La Monte Young - Rag Bhairava (Excerpt).wav plot

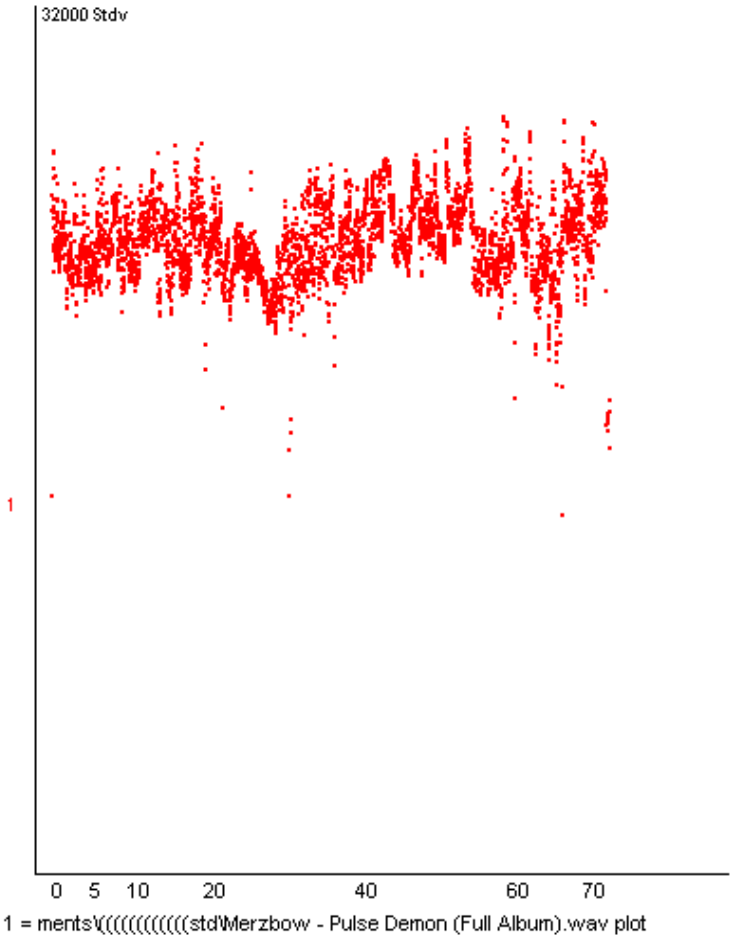

32000 Stdv
1
0
5
10
20
40
60
70
1 = Documents\((((((((((((std\Lou Reed - Metal Machine Music.wav plot

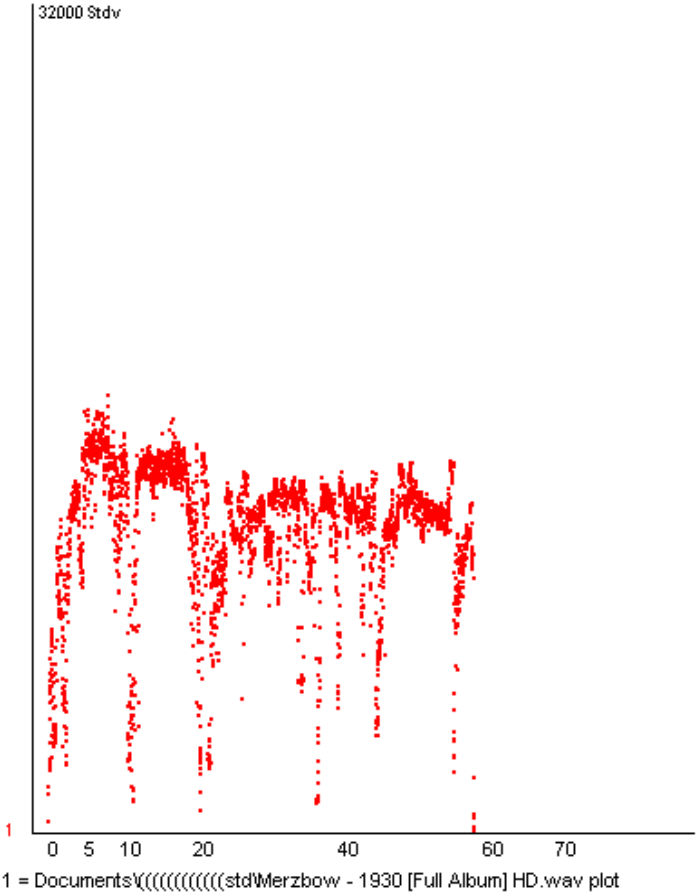


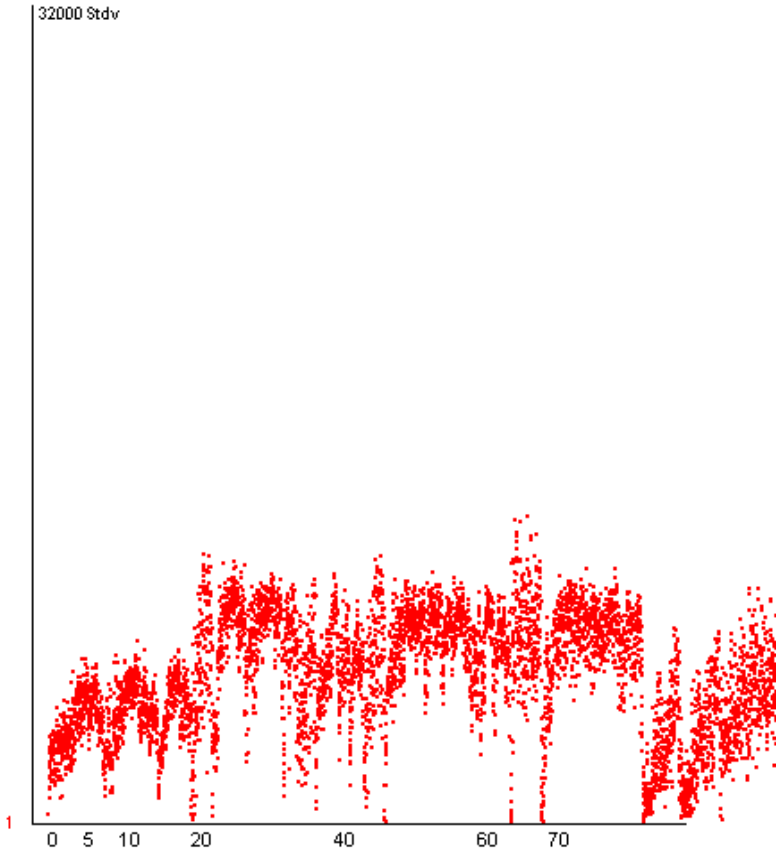
1 = 'The Titan' - Bernstein · Vienna Philharmonic Orchestra.wav plot



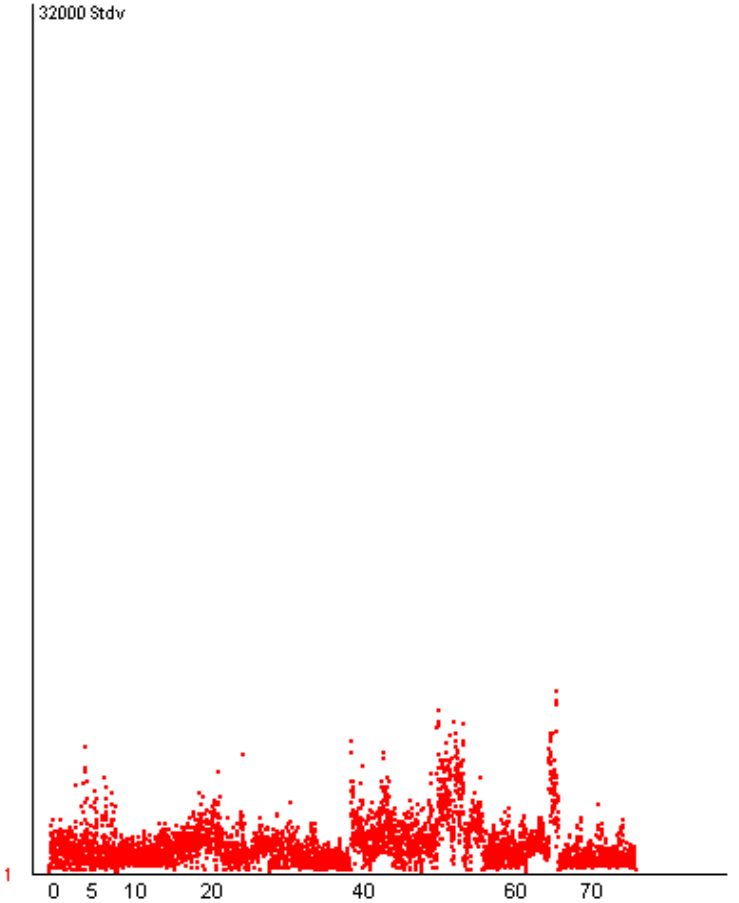
1 = surrection' (Lucerne Festival Orchestra, Claudio Abbado).wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = ments\((((((((((((std\Merzbow - Pulse Demon (Full Album).wav plot

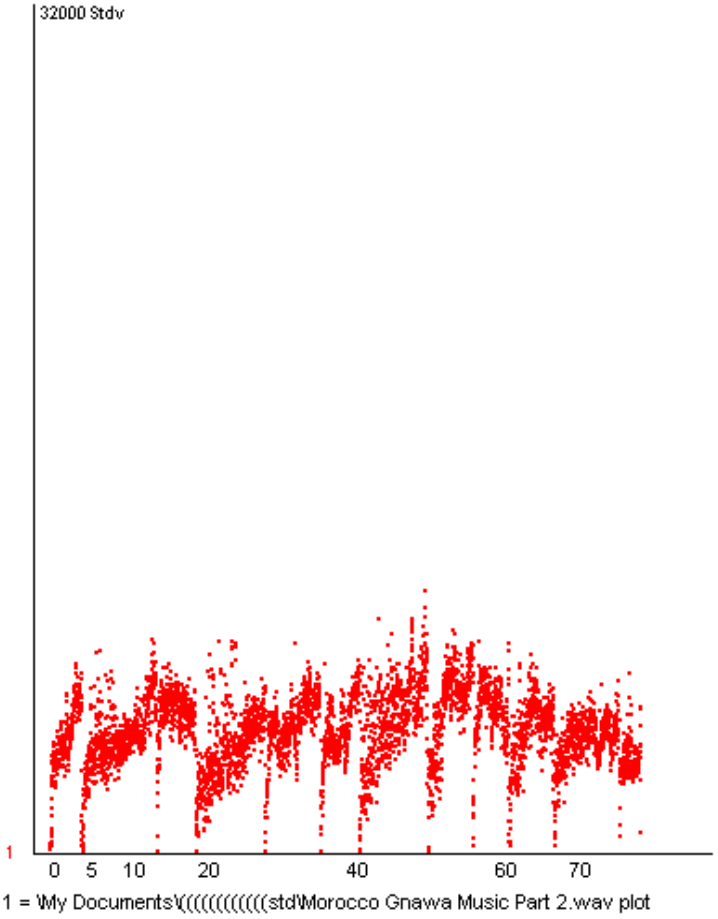

32000 Stdv
1
0
5
10
20
40
60
70
1 = Documents\((((((((((((std\Merzbow - 1930 [Full Album] HD.wav plot

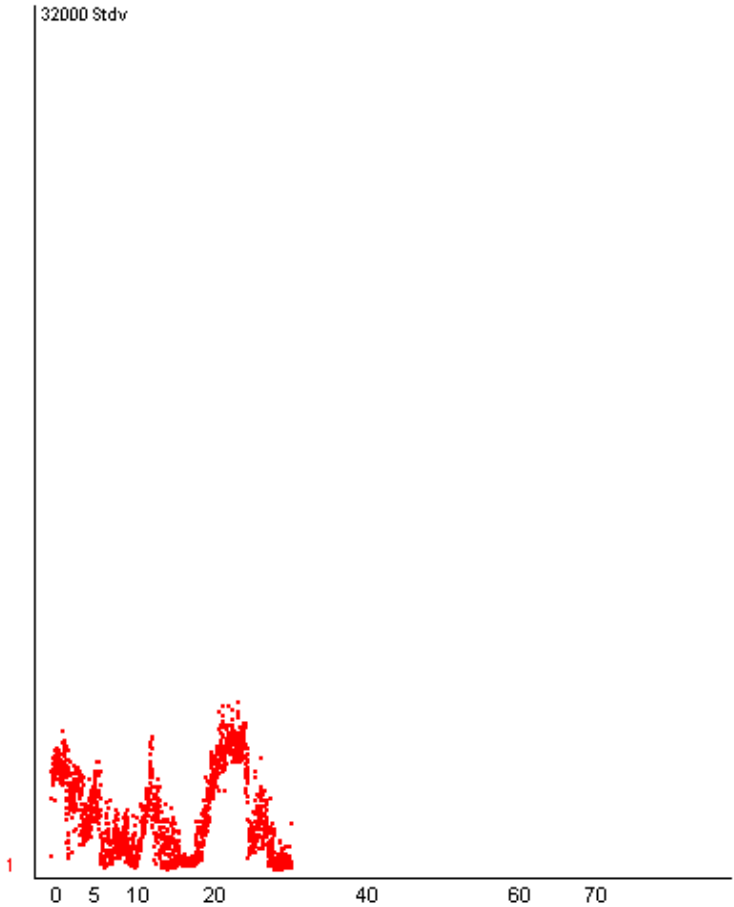


1 = td\update\Miles Davis - Bitches Brew (1970) - full album.wav plot

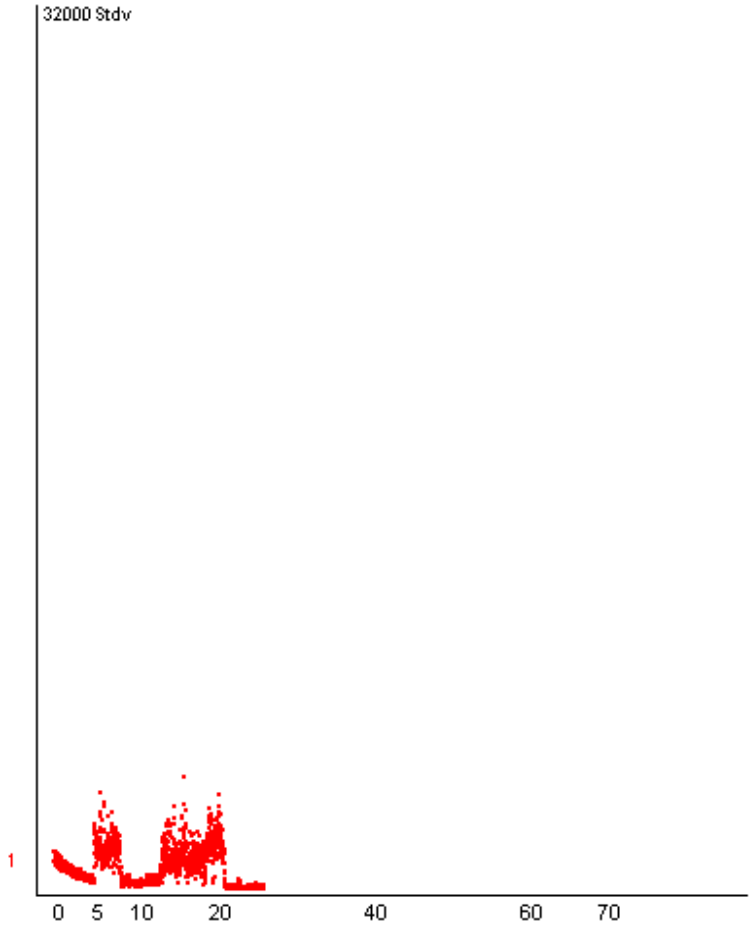


1 = (std\Monastic Chant 12th _13th C. European Sacred Music.wav plot

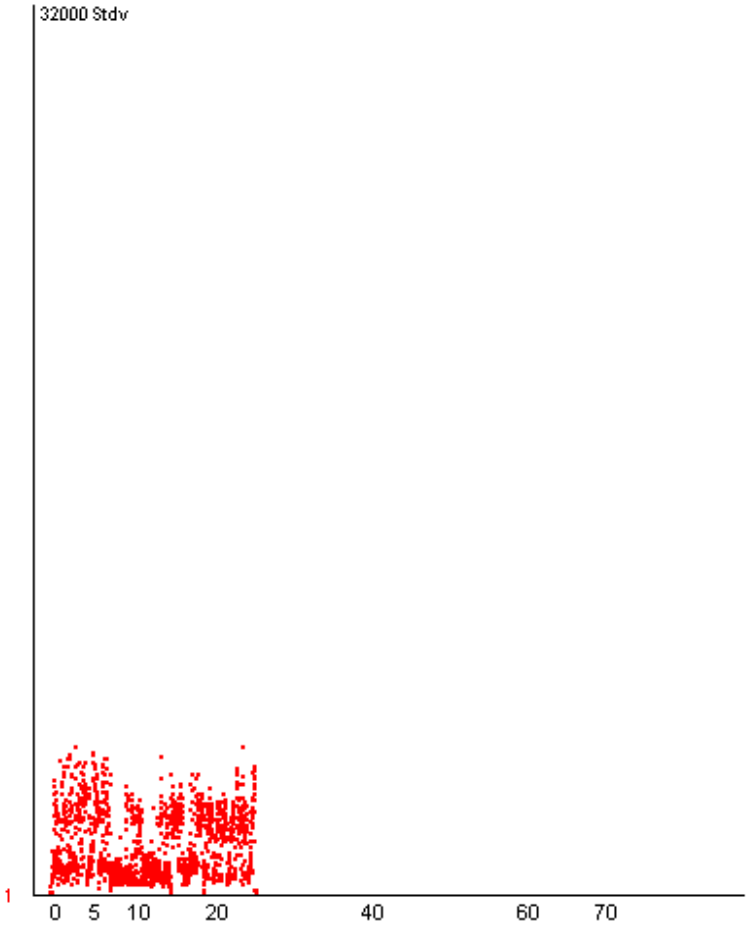

32000 Stdv
1
0
5
10
20
40
60
70
1 = \My Documents\((((((((((((std\Morocco Gnawa Music Part 2.wav plot



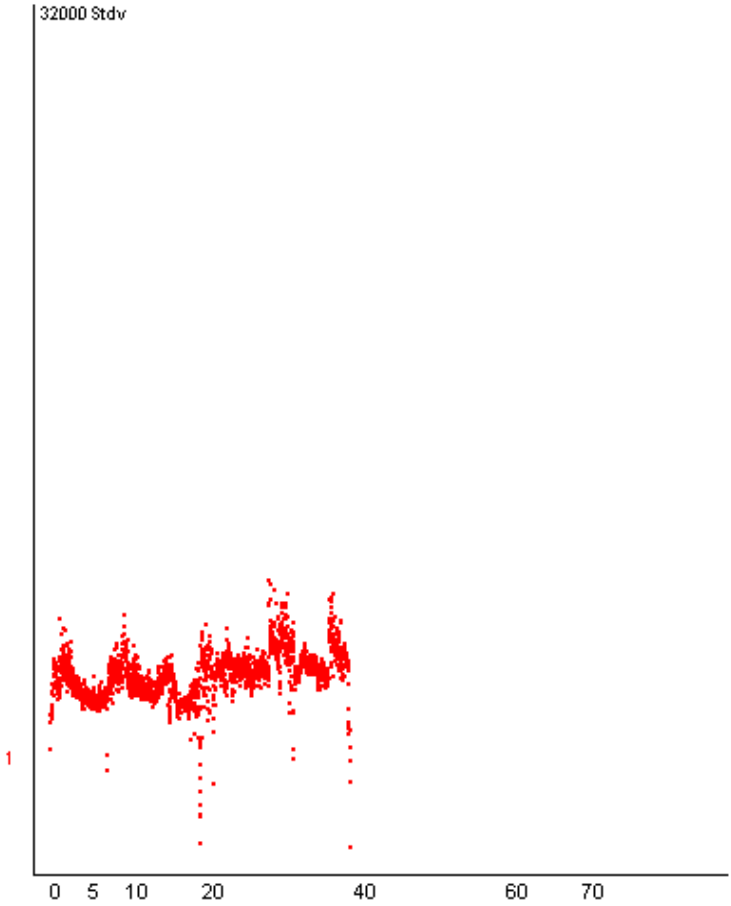
1 = ((((((((((std\Morton Subotnick Silver Apples Of The Moon.wav plot



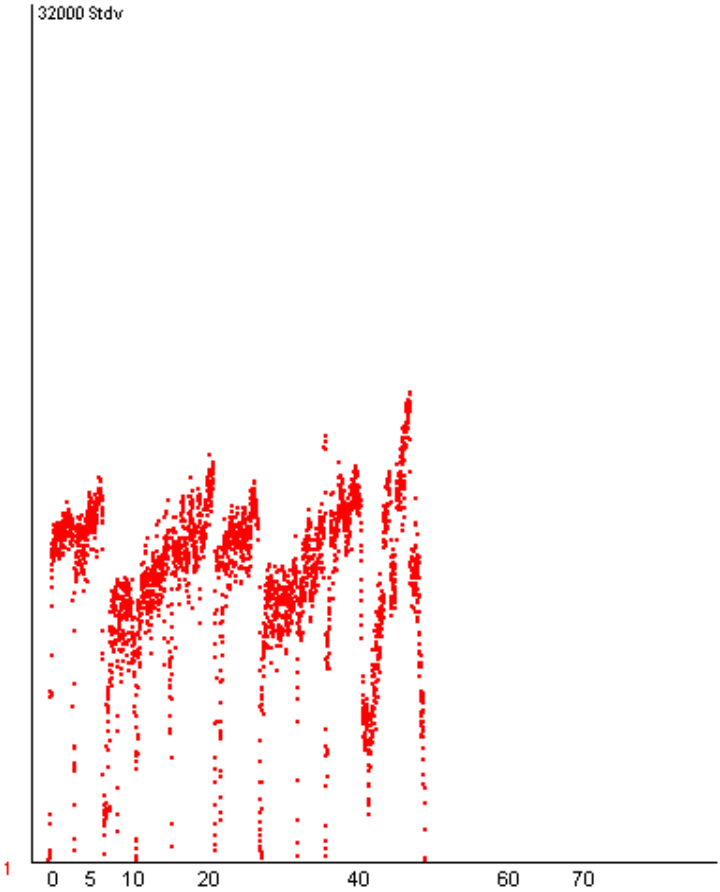
1 = \(((((((((((((std\Morton Subotnick The Wild Beasts (1978).wav plot



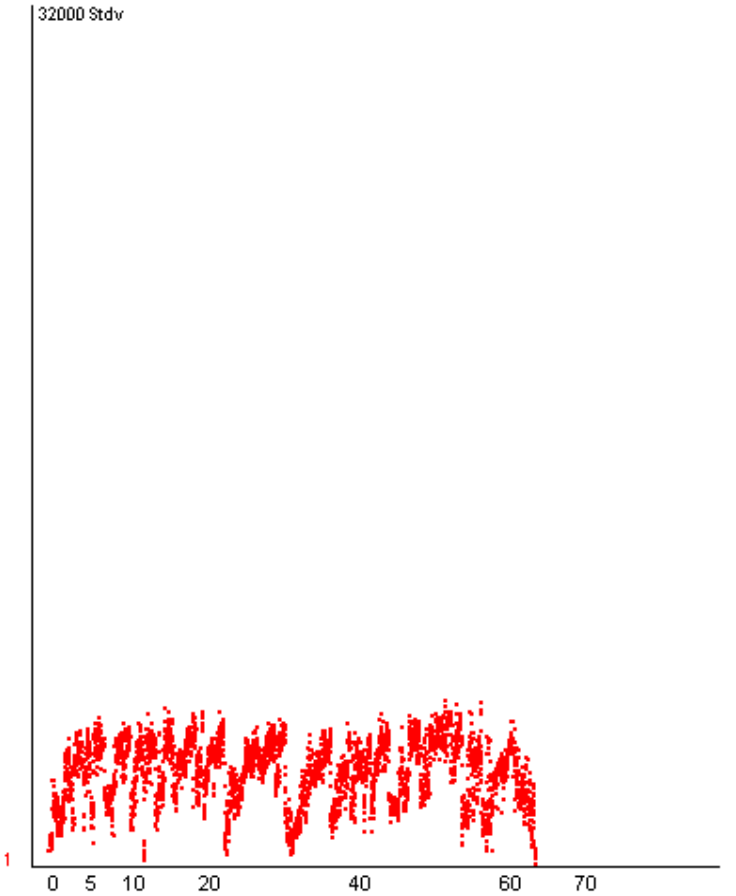
1 = d\Mozart - Symphony No. 40 in G minor, K. 550 [complete].wav plot



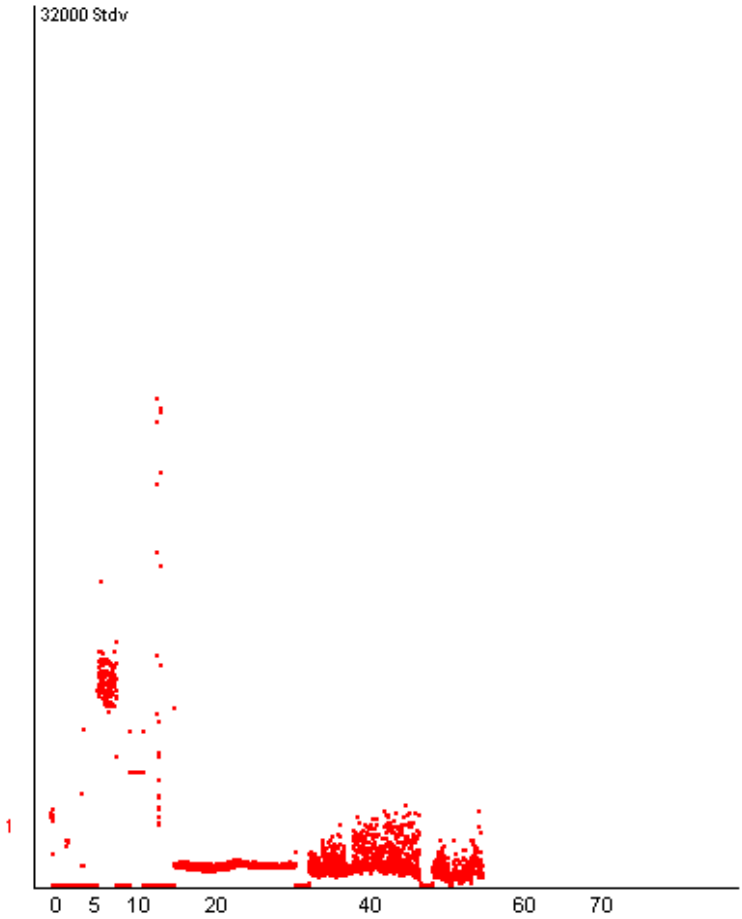
1 = ((((((((((std\The New Blockaders - Simphonie In X Major.wav plot



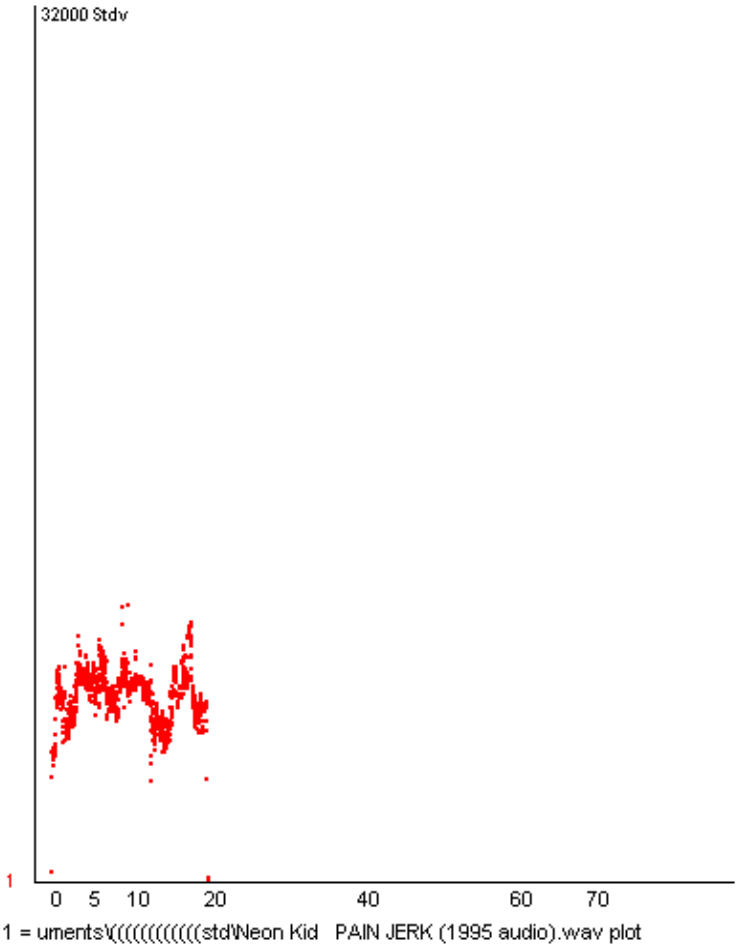
1 = ts\(((((((((((std\Oasis (whats the story) Morning Glory.wav plot

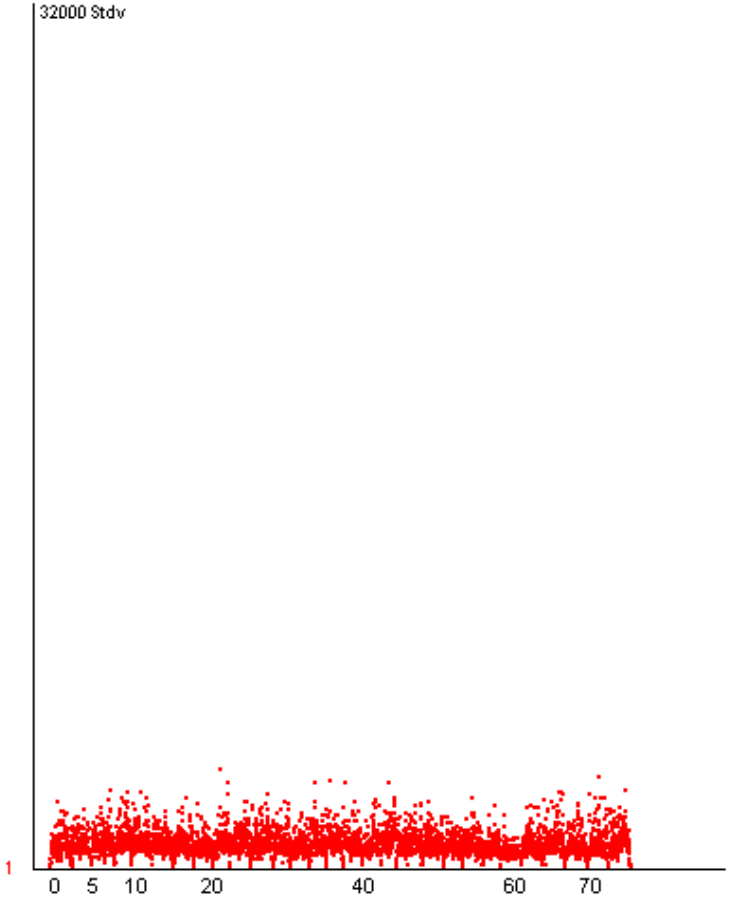


1 = field - Tubular bells II (Live in Edinburgh castle) 1992.wav plot

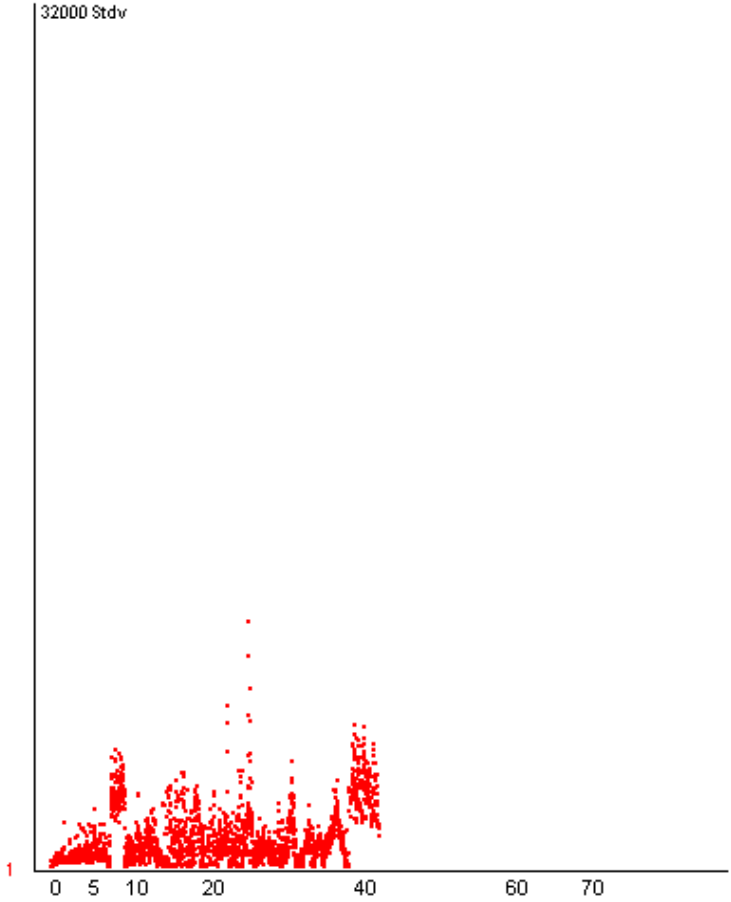


1 = Radst-SETI-Pimil-linein-millrandnos-cosmicBG-10milprimes.wav plot

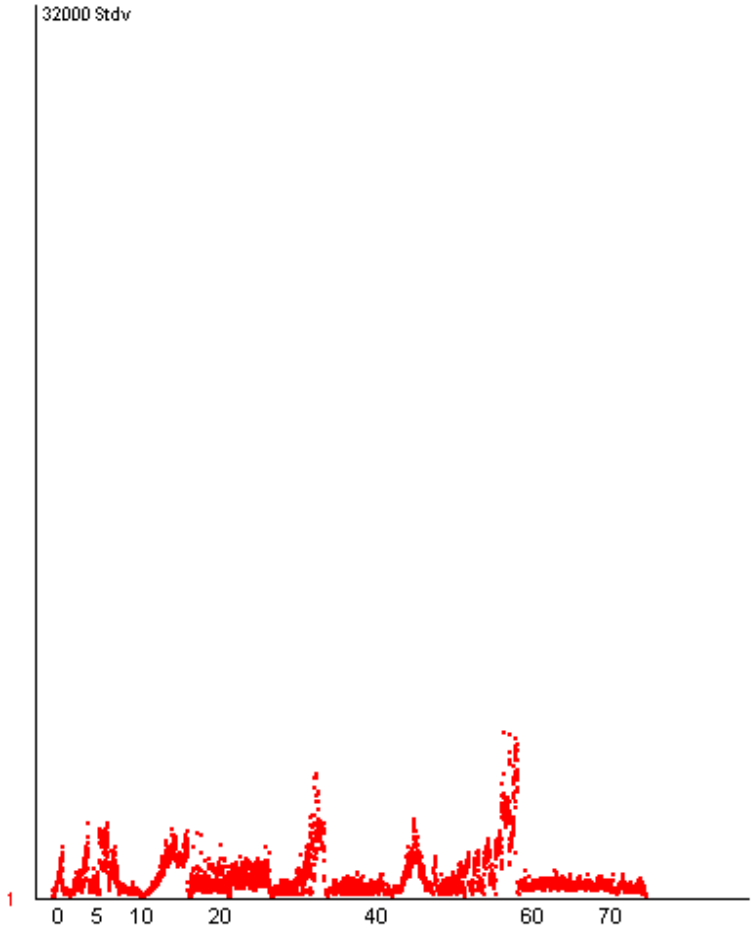

32000 Stdv
1
0
5
10
20
40
60
70
1 = uments\((((((((((((std\Neon Kid PAIN JERK (1995 audio).wav plot



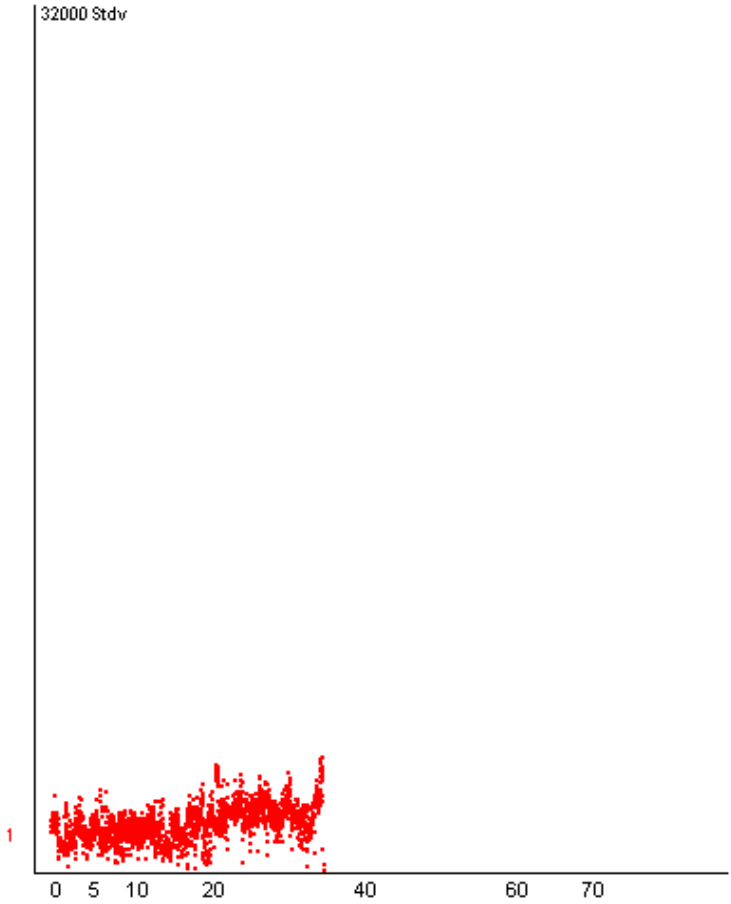
1 = ((((((((((std\G. P. da Palestrina - Motets for 5 voices.wav plot



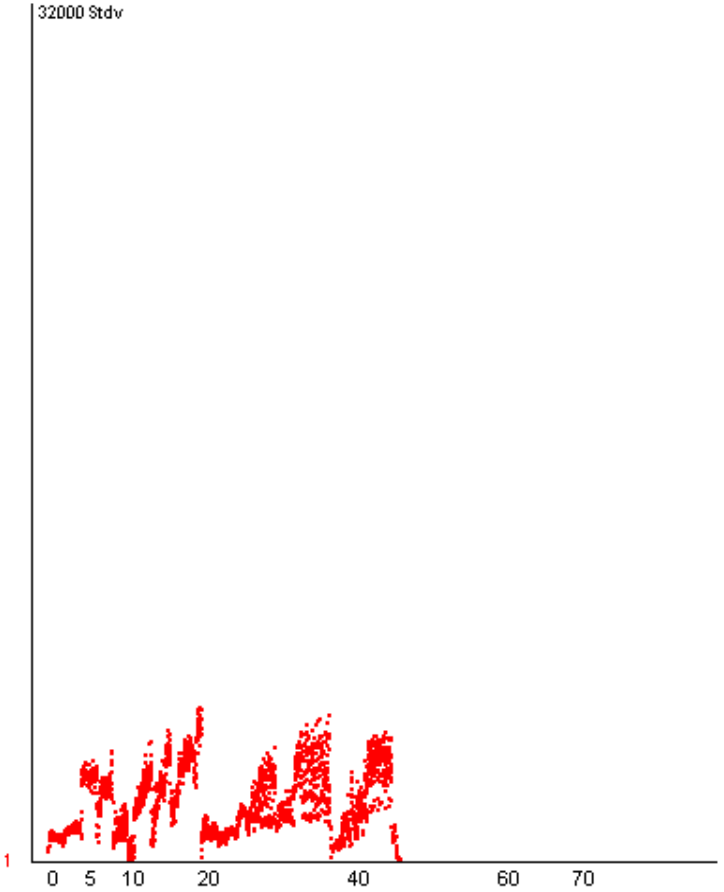
1 = — Symphony No. 4 'Los Angeles' (UK Première) Proms 2010.wav plot



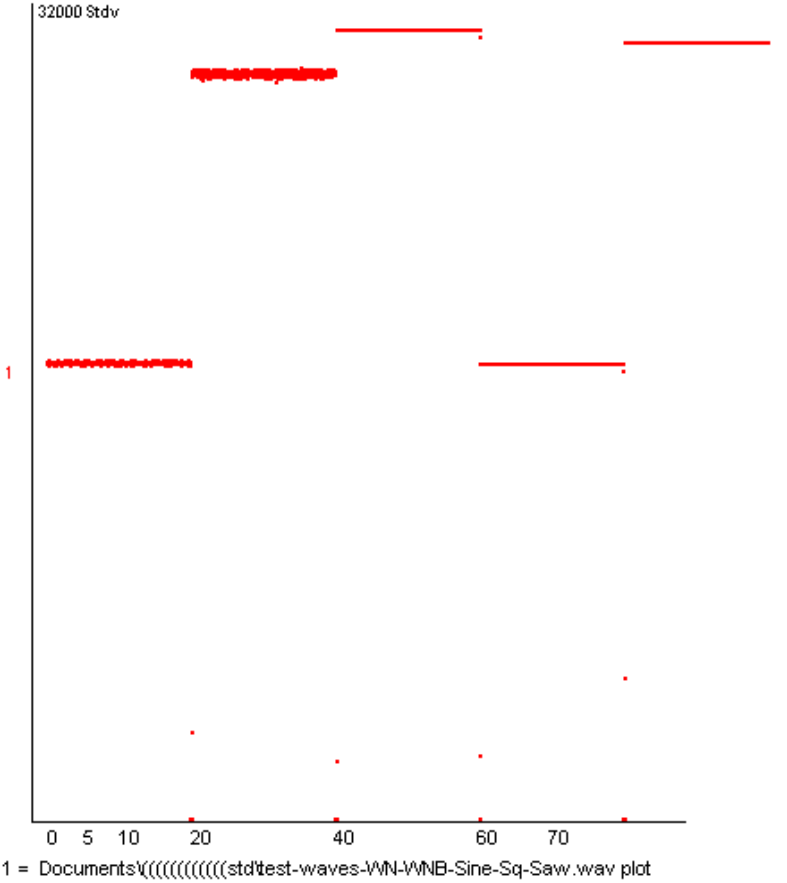
1 = tor\My Documents\((((((((((((std\Arvo Pärt - Tabula rasa.wav plot

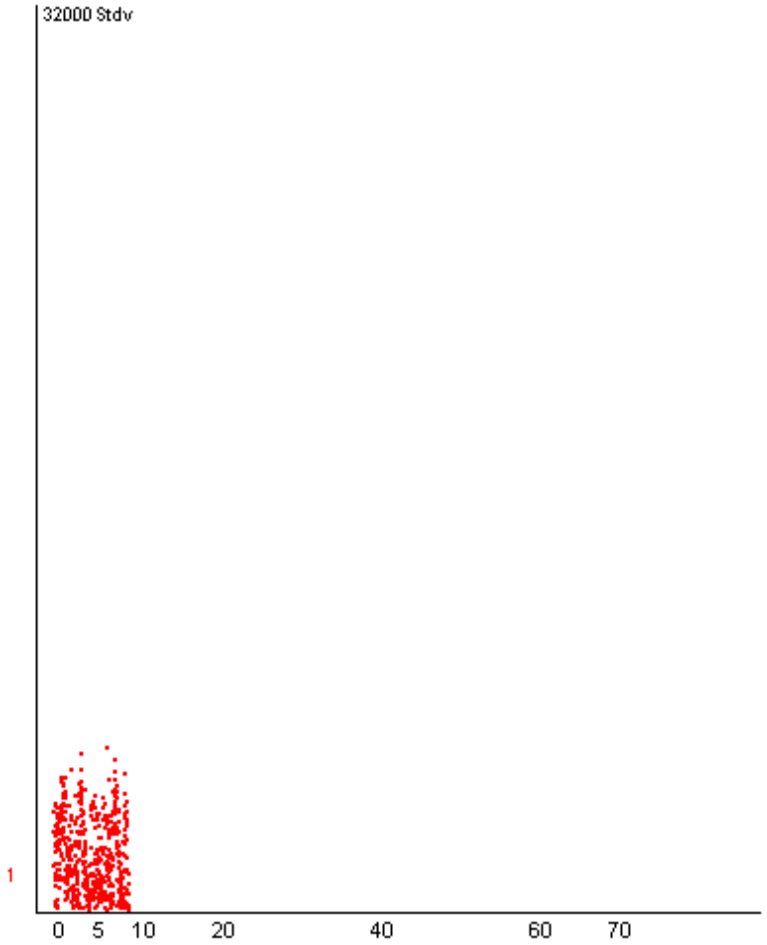


1 = ry Partch And On The Seventh Day Petals Fell In Petaluma.wav plot

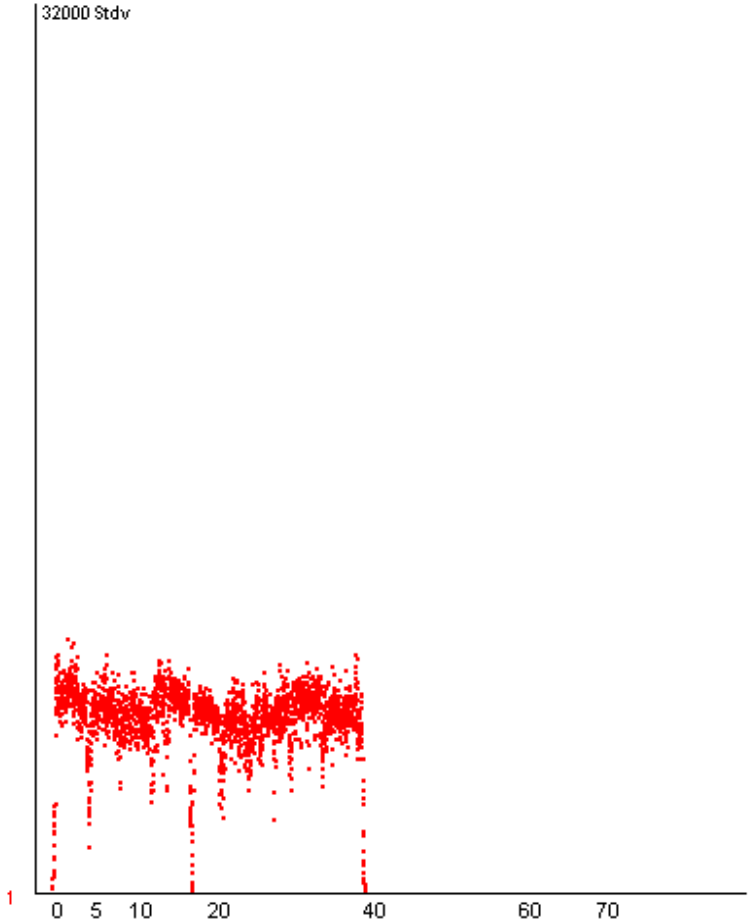


1 = (((((((std\Philip Glass Akhnaten (Complete) Act I (1-3).wav plot

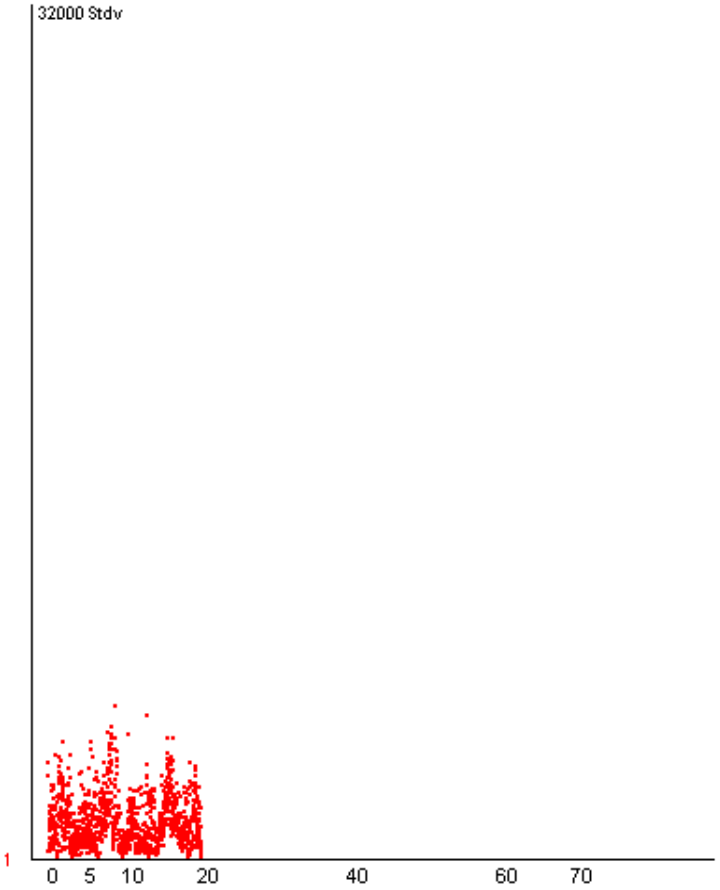

32000 Stdv
1
0
5
10
20
40
60
70
1 = Documents\(((((((((((((std\test-waves-WN-WNB-Sine-Sq-Saw.wav plot



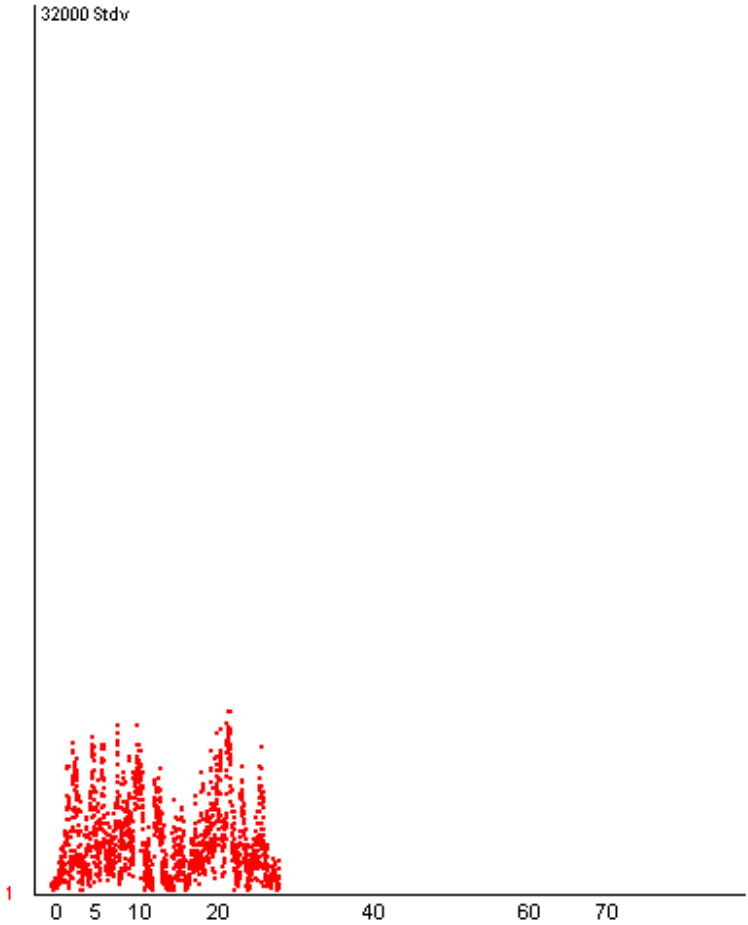
1 = gs\Administrator\My Documents\((((((((((((std\Webernop28.wav plot



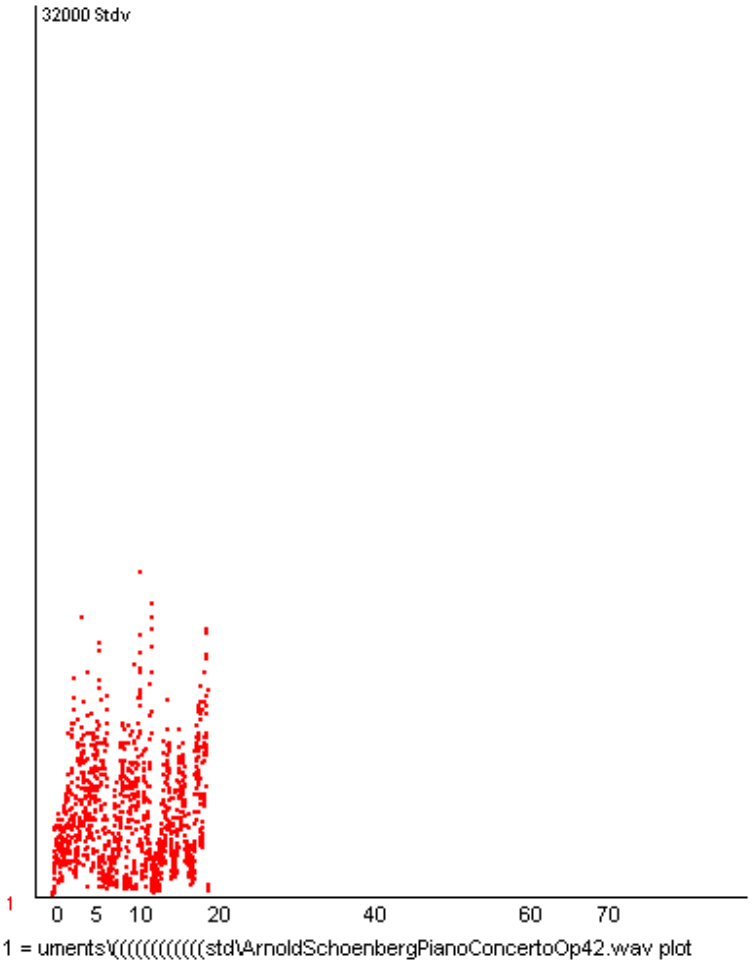
1 = (std\The Rolling Stones Still Life American Concert 1981.wav plot

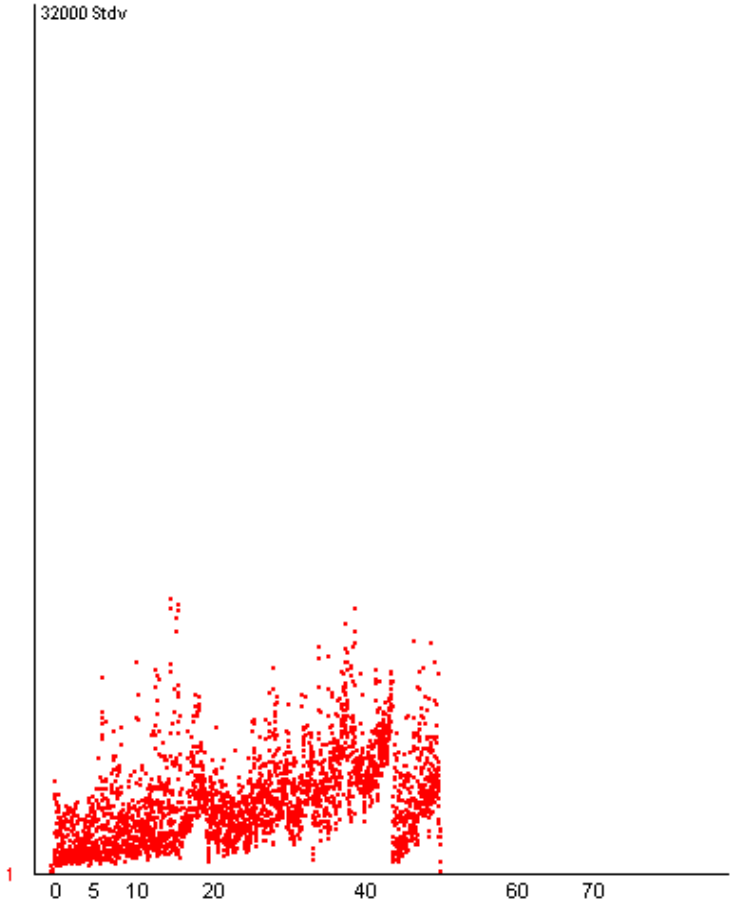


1 = ((((std\Pierre Schaeffer _Pierre Henry Orphée 53 (1953).wav plot



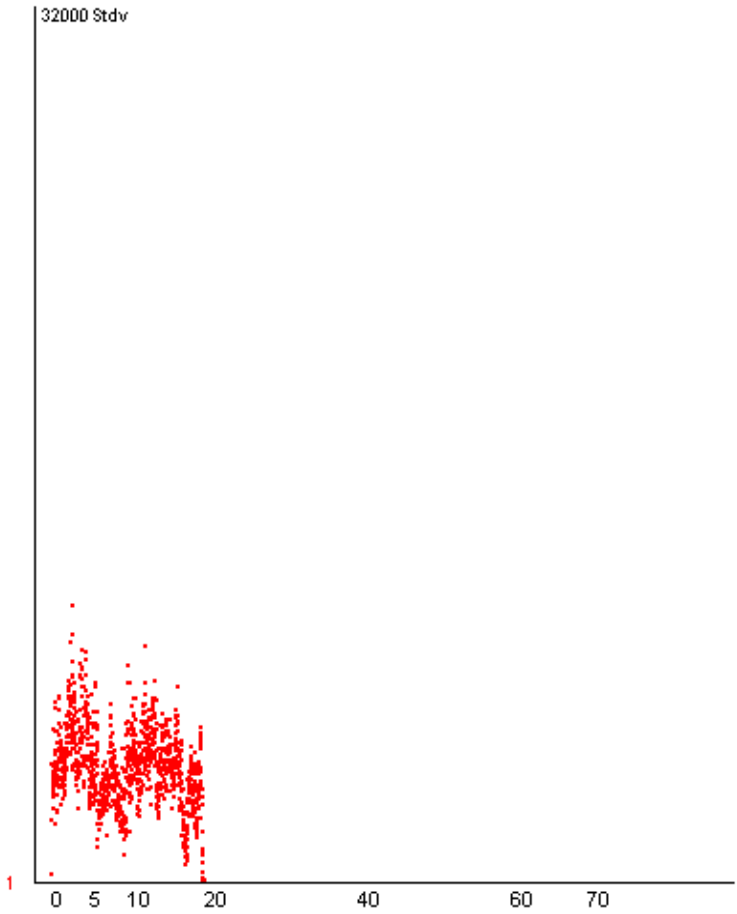
1 = y Documents\(((((((((((std\SchoenbergVerklärte NachtOp4.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = uments\((((((((((((std\ArnoldSchoenbergPianoConcertoOp42.wav plot

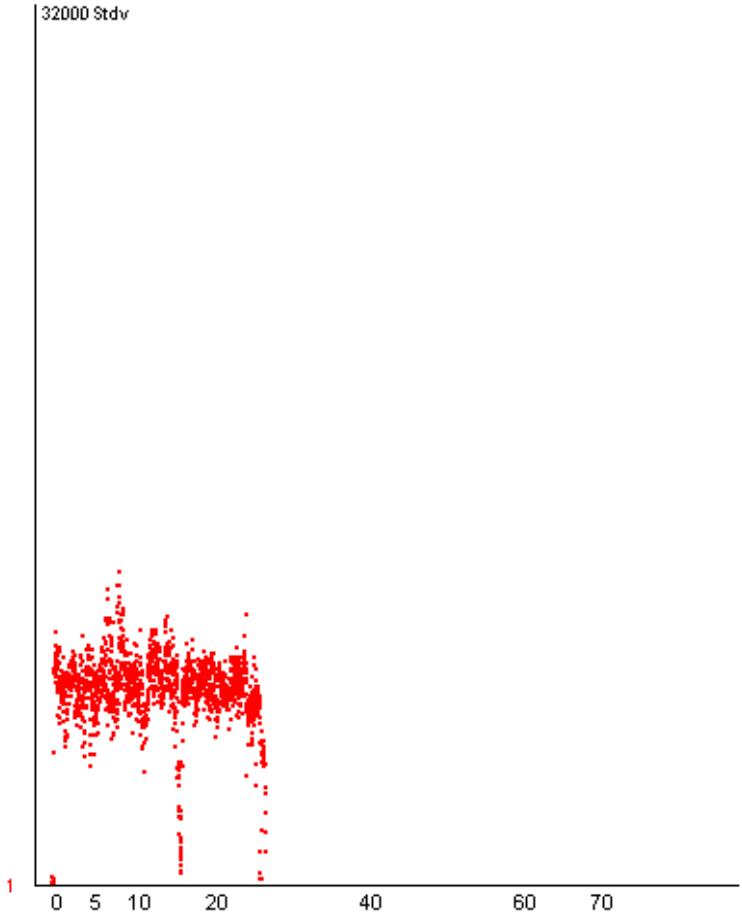


1 = (((((((std\Instrumental Music (Sitar) by Sri Partha Bose.wav plot

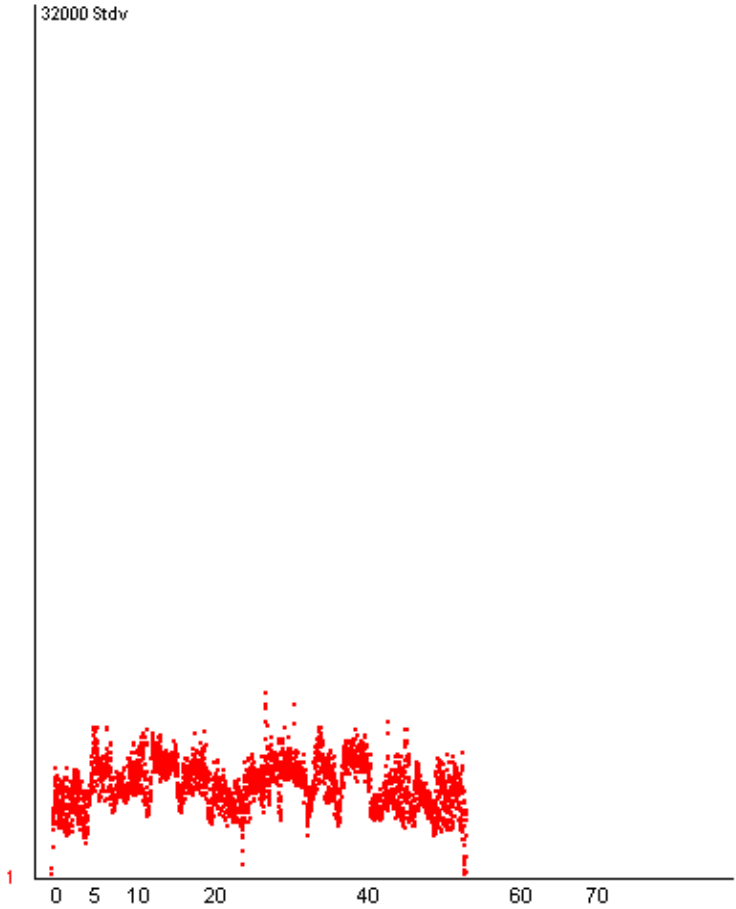
1 = \update\Spontaneous Music Ensemble - Boileau Road (1980).wav plot



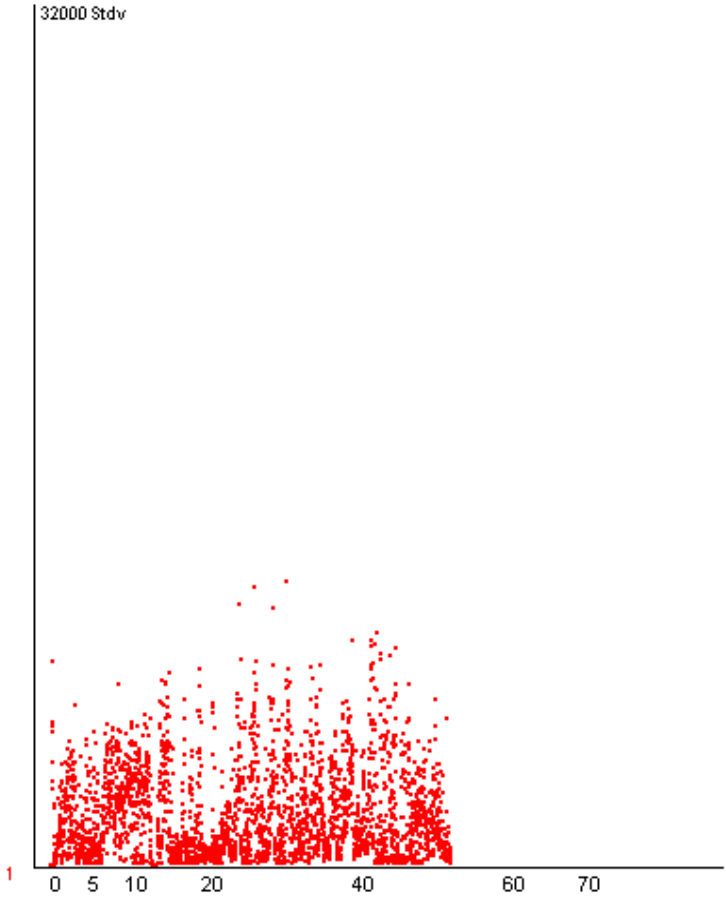
1 = ((std\update\Spontaneous Music Ensemble - Familie (1968).wav plot



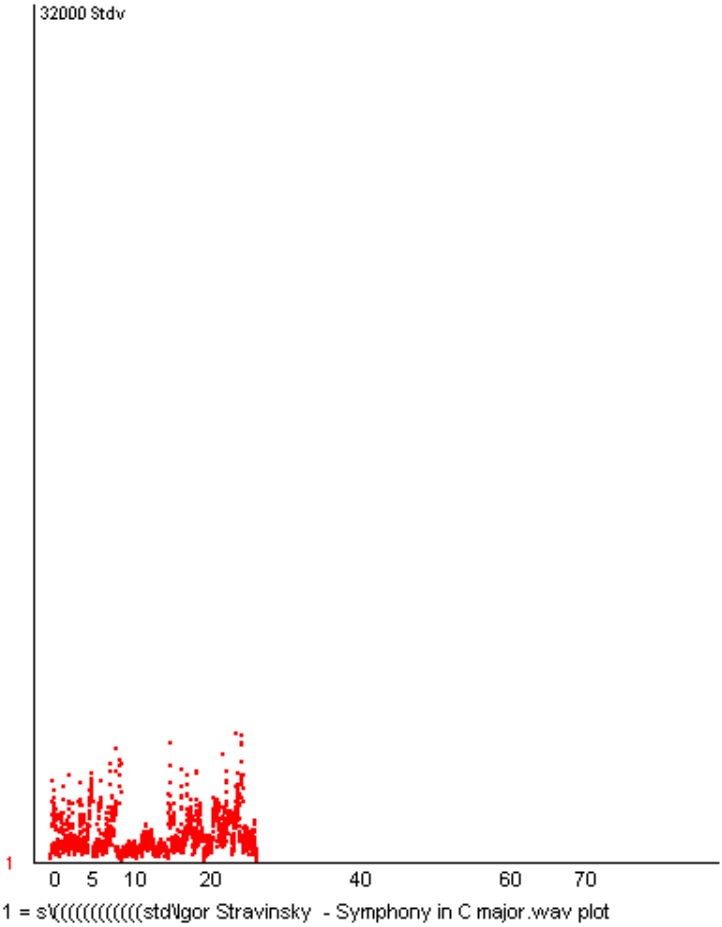
1 = y Documents\(((((((((((std\Steve Reich Different Trains.wav plot

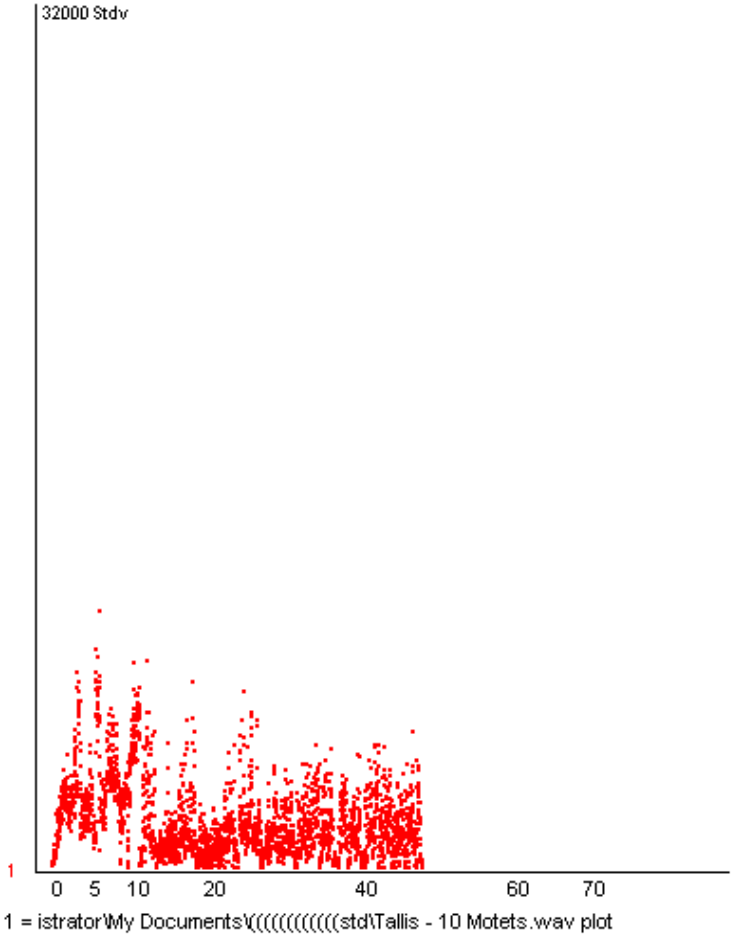


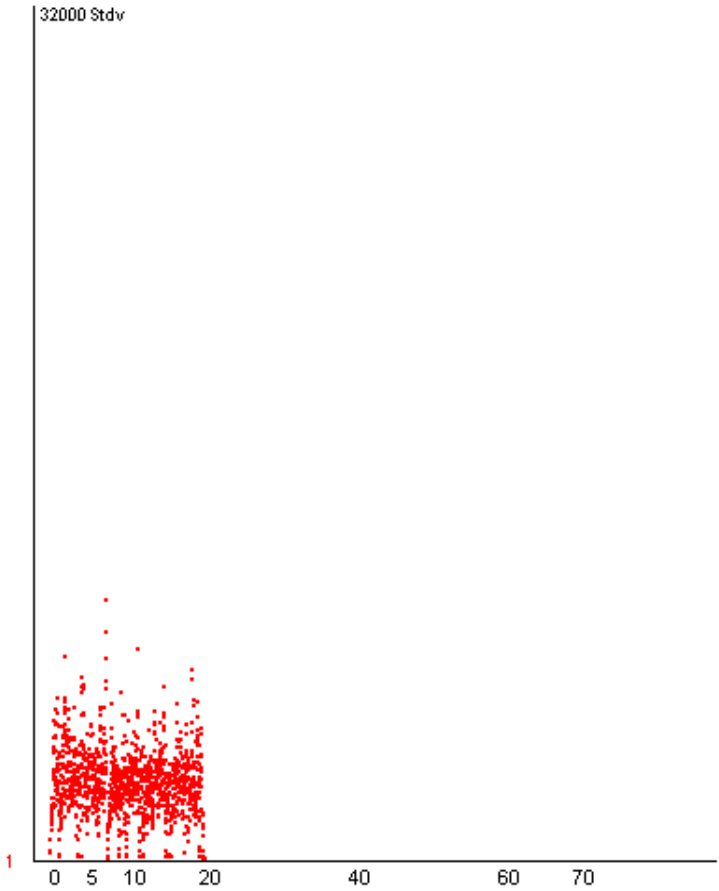
1 = (std\Steve Reich - Music for 18 Musicians [2008 JAPAN].wav plot



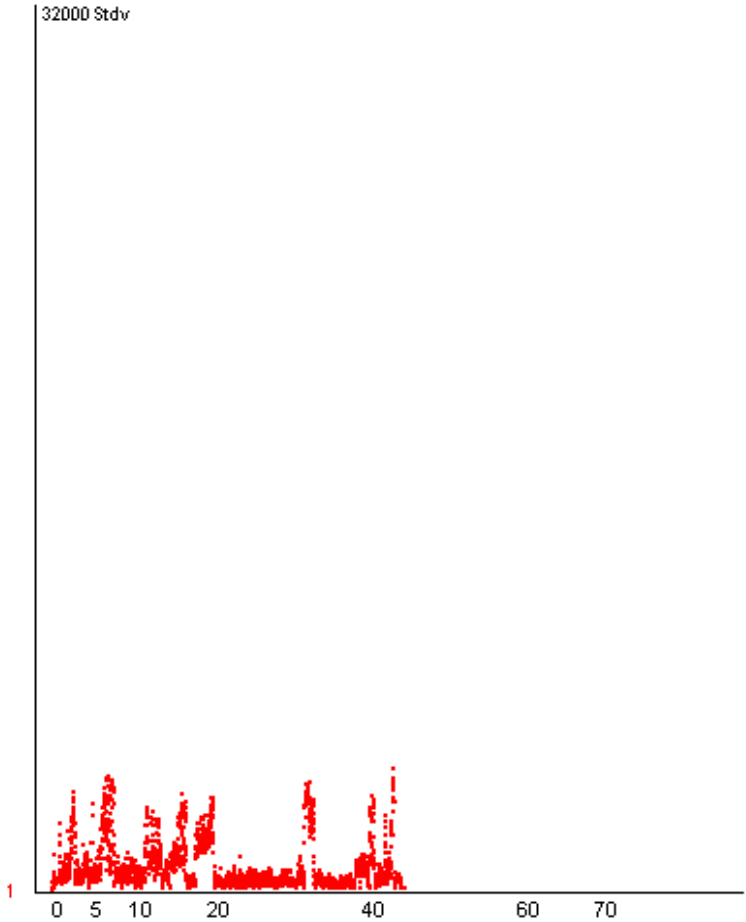
1 = (std\Karlheinz Stockhausen Gesang Der Jünglinge-Kontakte.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = s\(((((((((((((std\Igor Stravinsky - Symphony in C major.wav plot

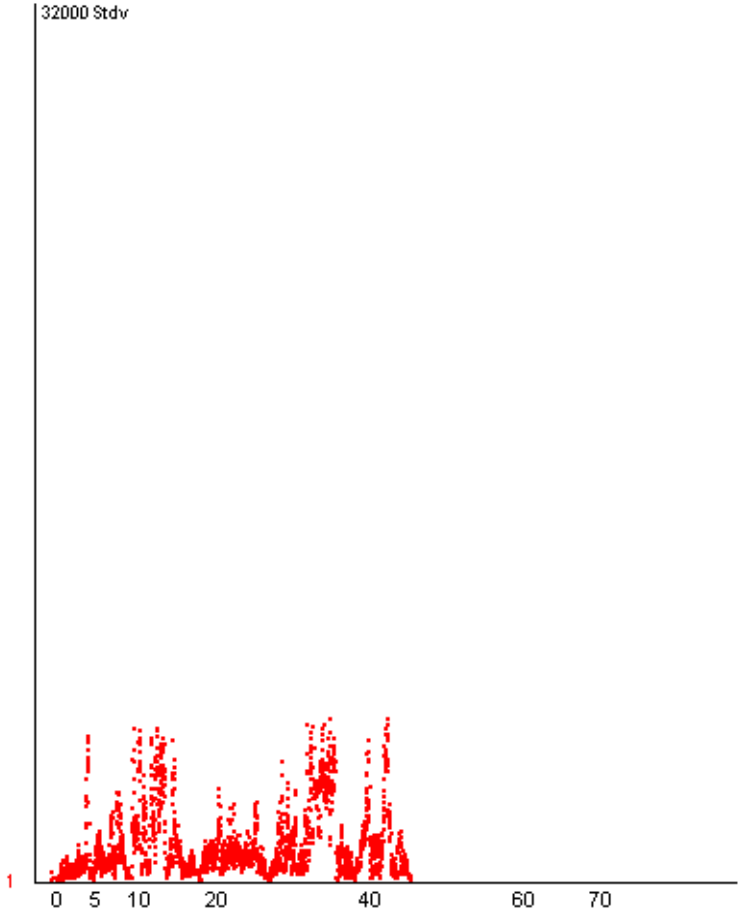

32000 Stdv
1
0
5
10
20
40
60
70
1 = istrator\My Documents\((((((((((((std\Tallis - 10 Motets.wav plot



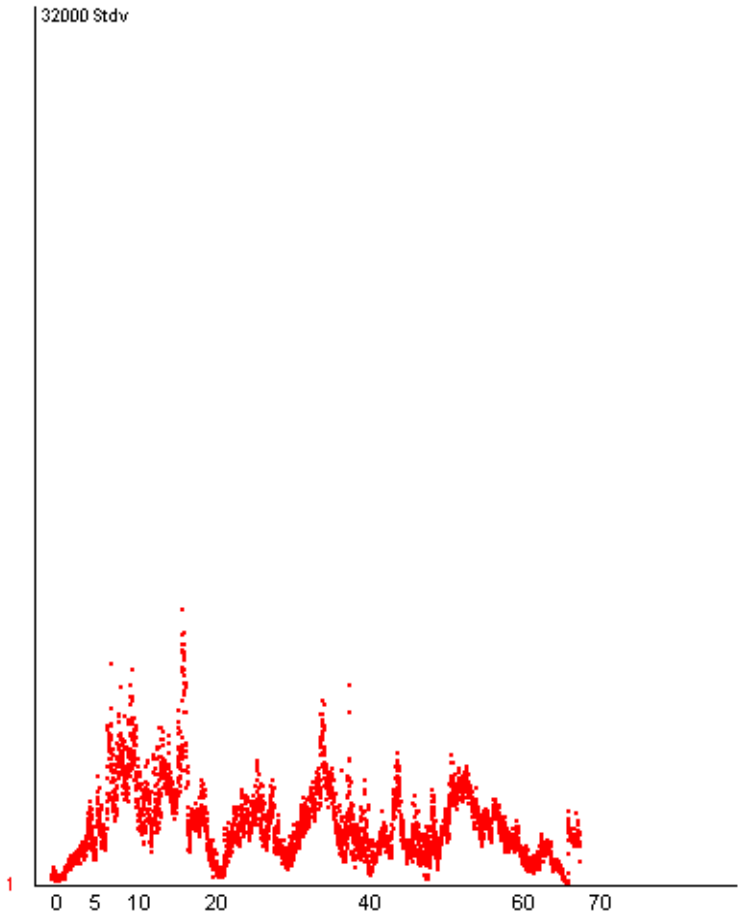
1 = My Documents\((((((((((((std\Thomas Tallis - Lamentation.wav plot



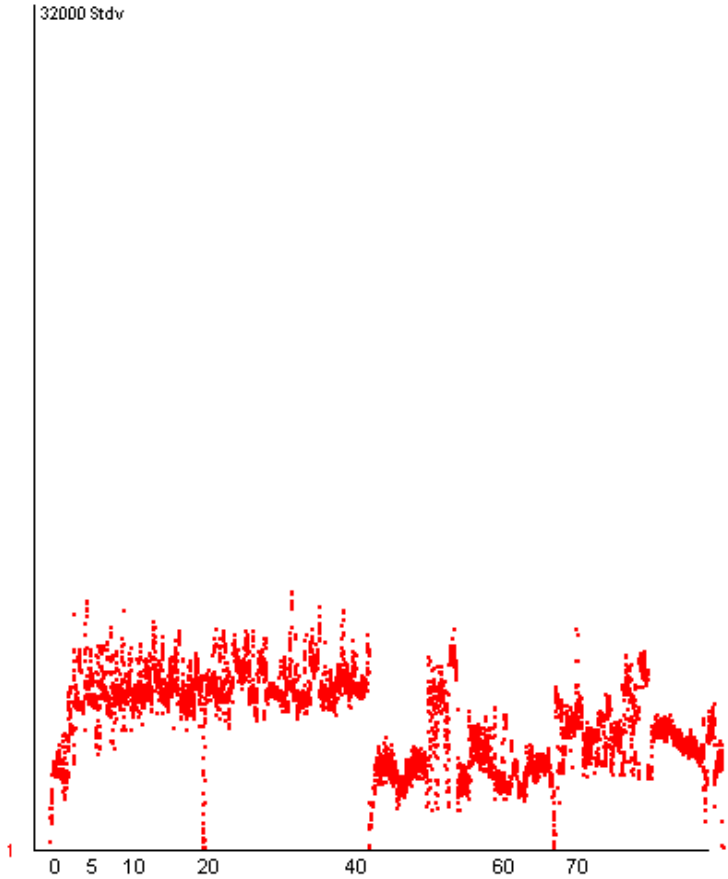
1 = My Documents\((((((((((((std\Tavener The Protecting Veil.wav plot



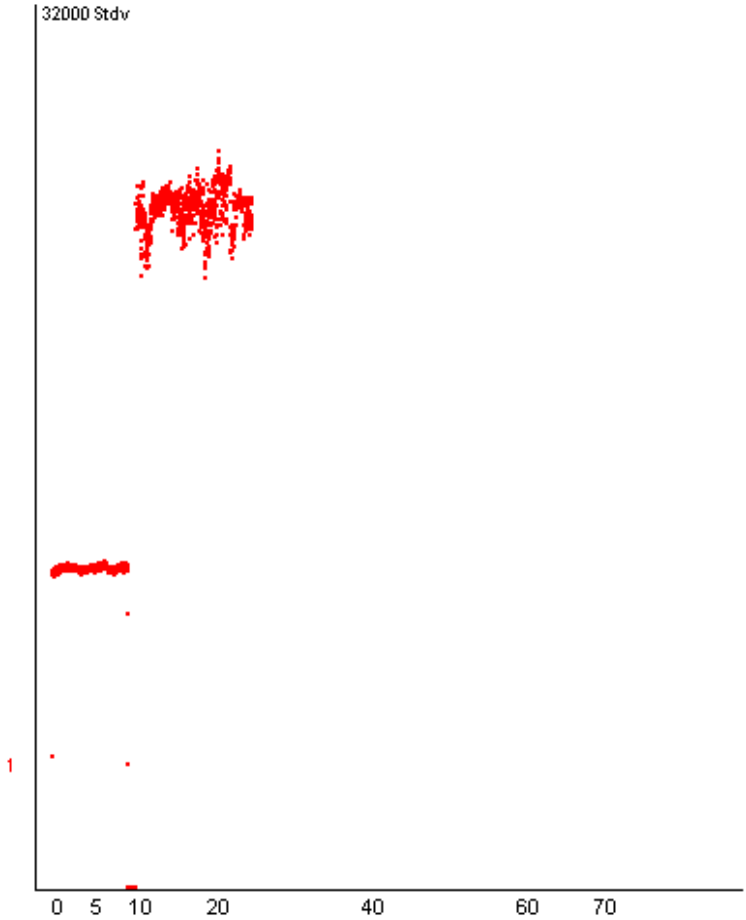
1 = \(((((((((((std\Tchaikovsky - Symphony no6 (Pathetique).wav plot



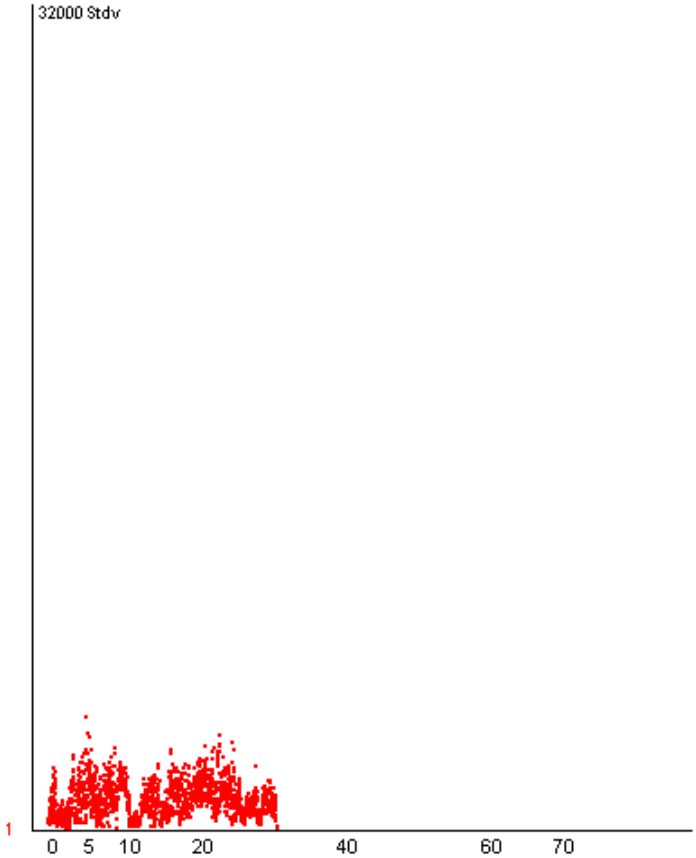
1 = riends – 'In C' – Boiler Room Amsterdam Live Performance.wav plot



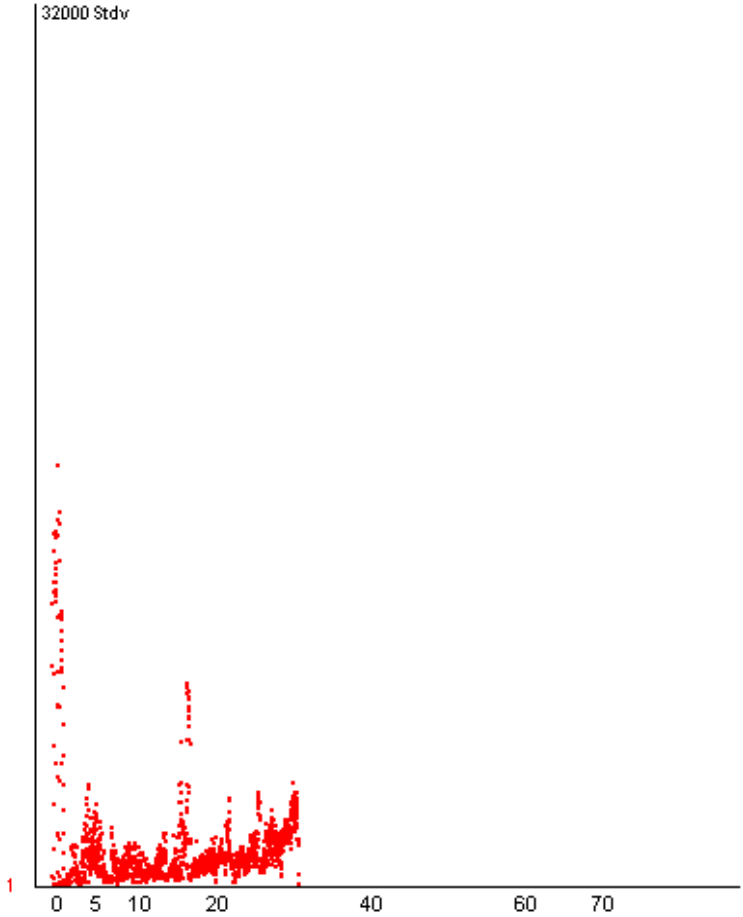
1 = \(((((((((((std\Terry Riley - Persian Surgery Dervishes.wav plot



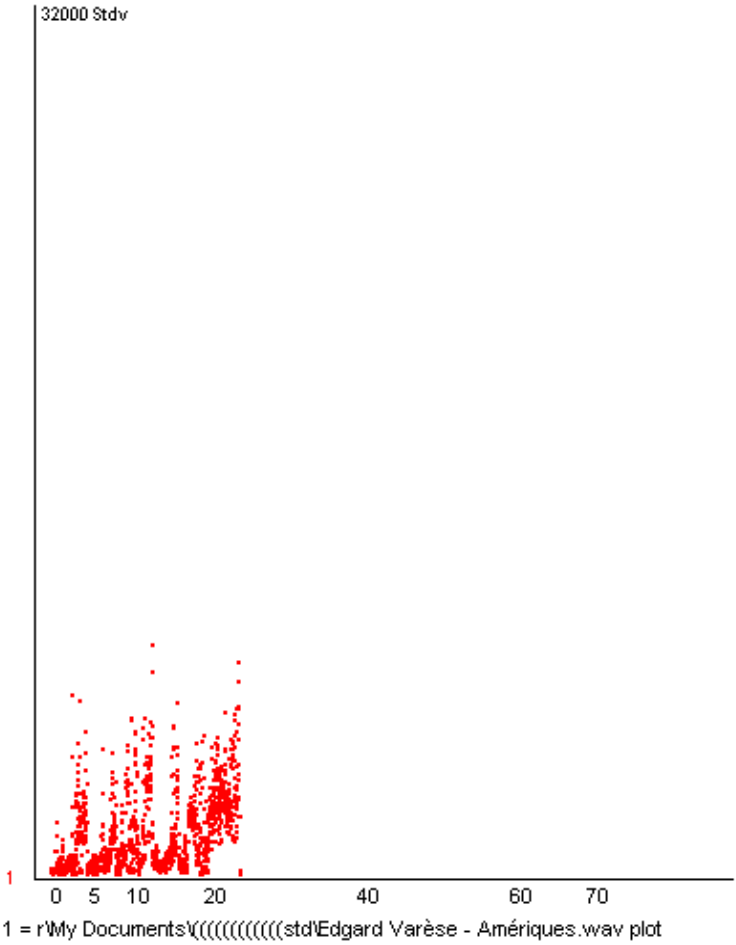
1 = e Rita - Rebianchilized (SIDE A Excerpt) _Anna Christie.wav plot

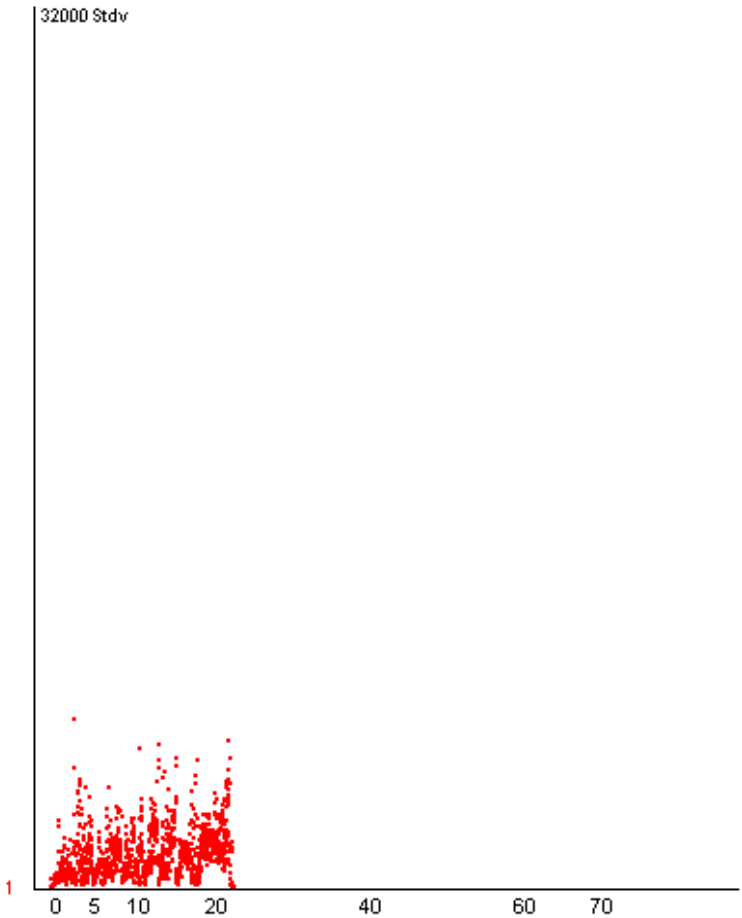


1 = y _Evan Parker - In Which the Listener Will Perceive....wav plot

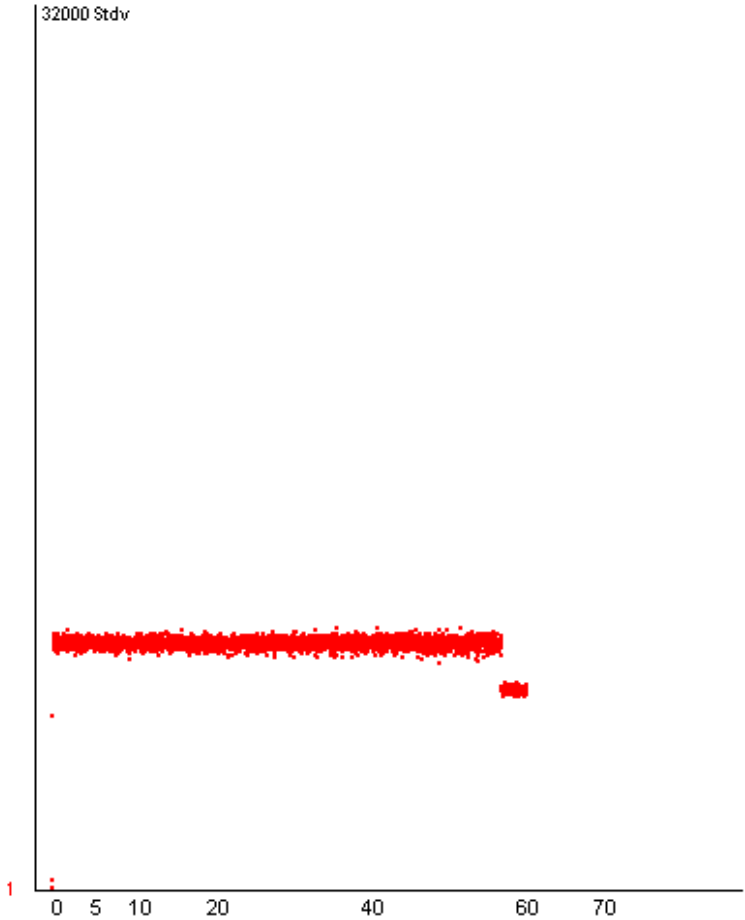


1 = nts\((((((((((((std\update\David Tudor - Phonemes (1981).wav plot

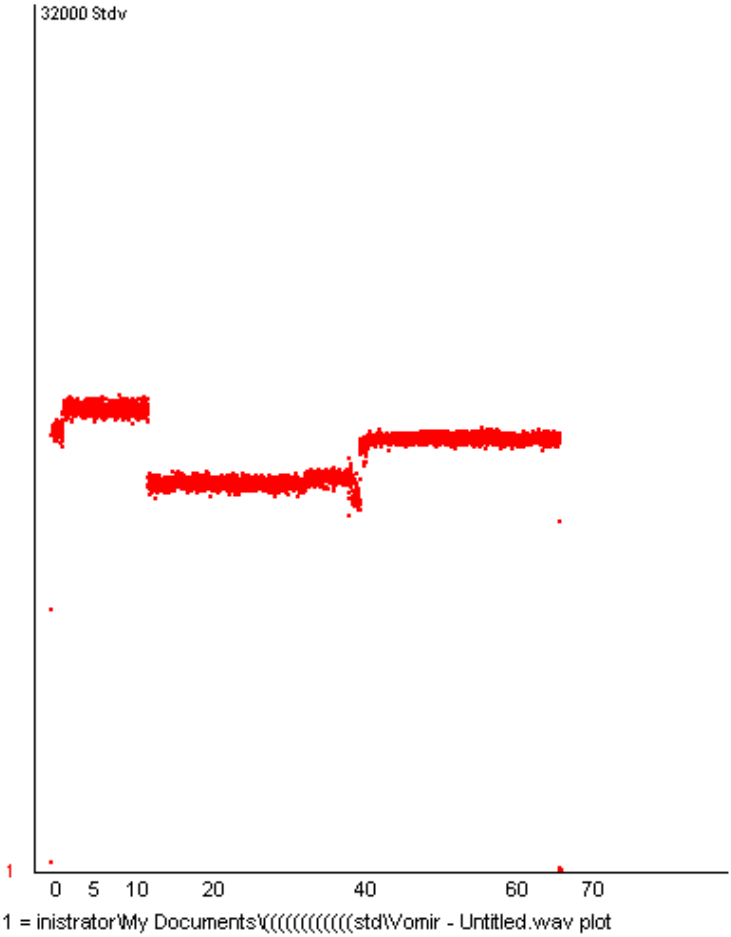

32000 Stdv
1
0
5
10
20
40
60
70
1 = r\My Documents\((((((((((((std\Edgard Varèse - Amériques.wav plot

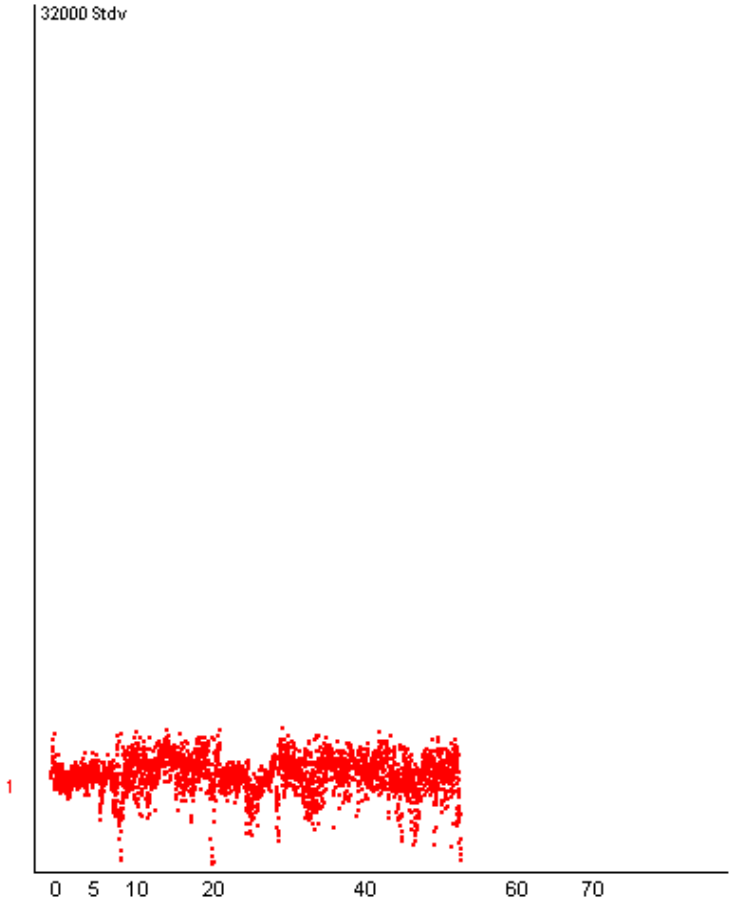


1 = riques - Ensemble intercontemporain - Matthias Pintscher.wav plot

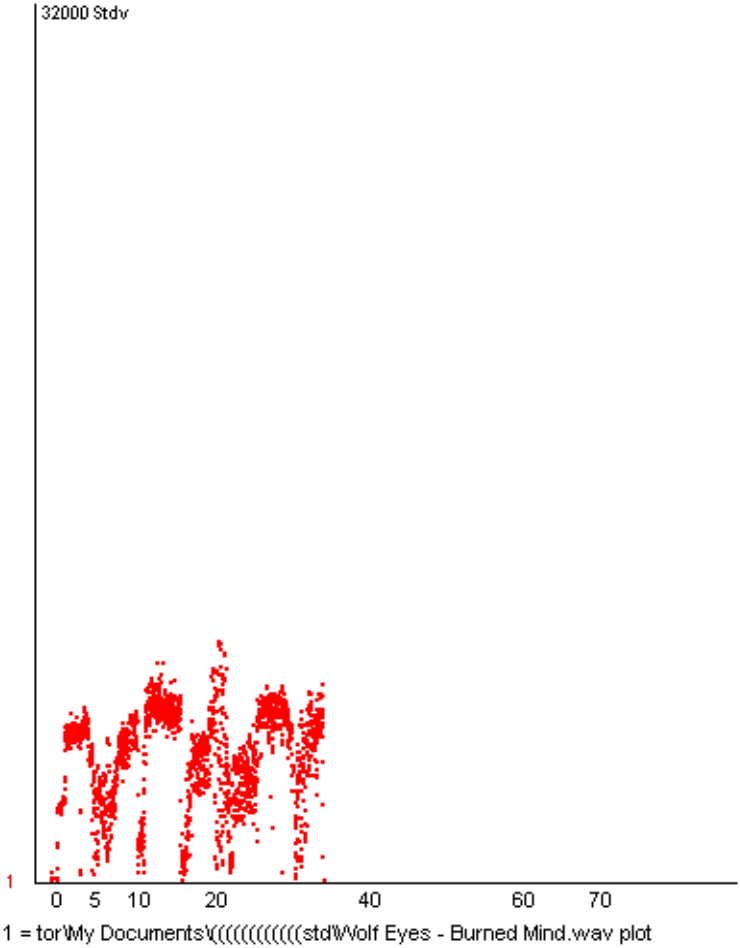


1 = ator\My Documents\((((((((((((std\Vomir - Claustration 1.wav plot

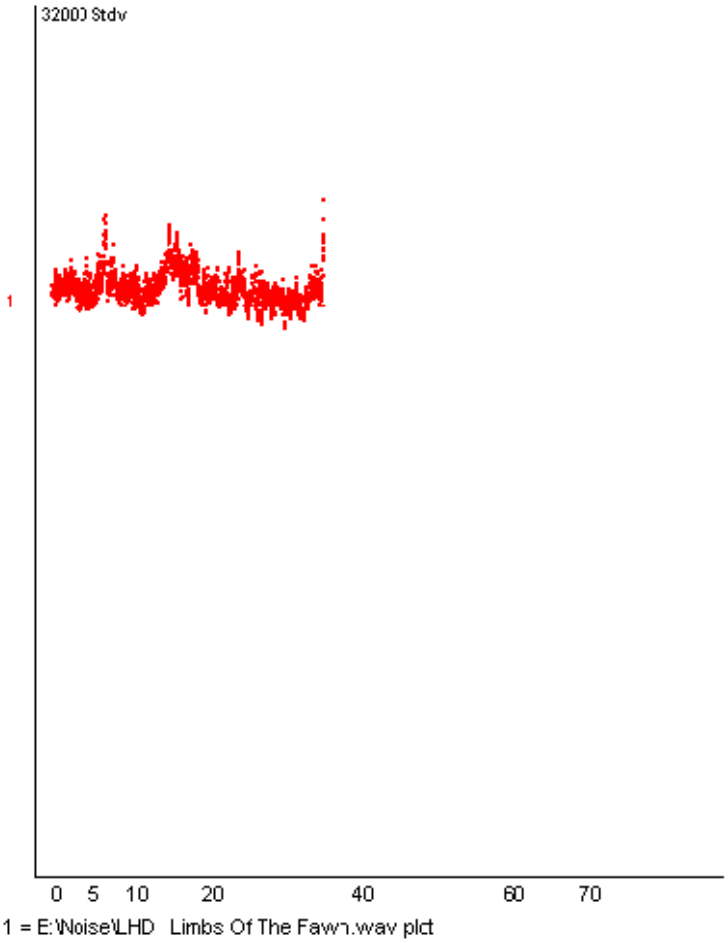

32000 Stdv
1
0
5
10
20
40
60
70
1 = inistrator\My Documents\((((((((((((std\Vomir - Untitled.wav plot

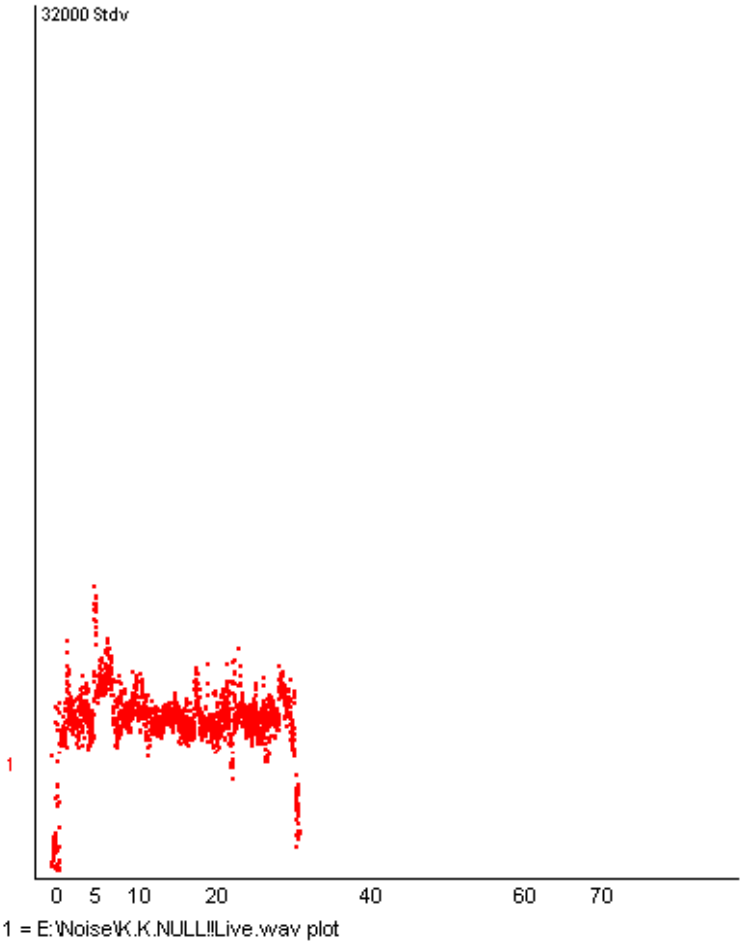


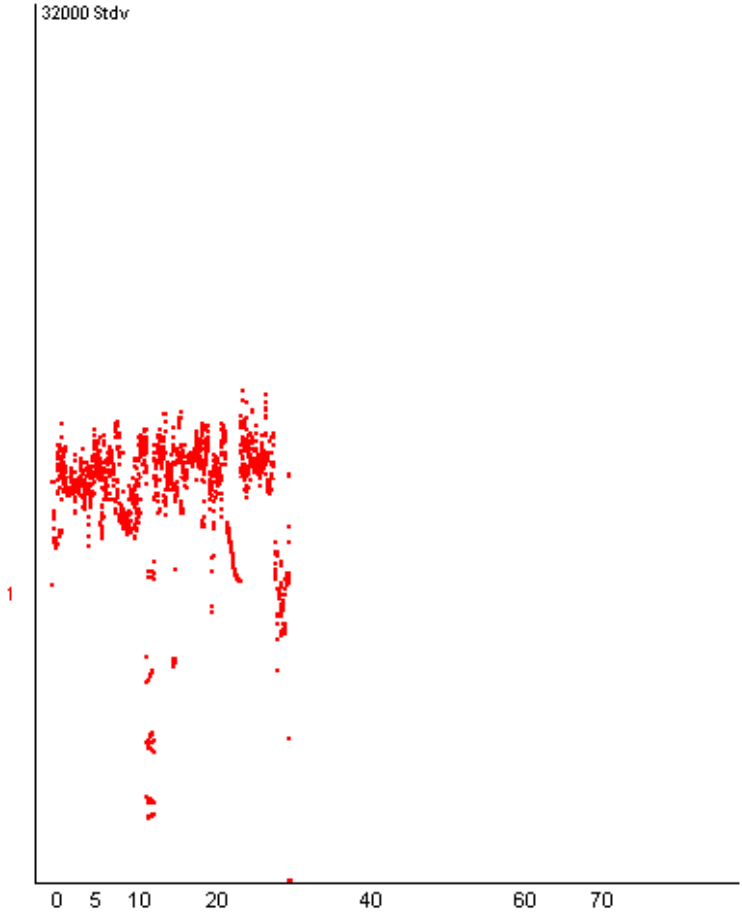
1 = \((((((((((((std\John Zorn - Jazz in Marciac - Live 2010.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = tor\My Documents\((((((((((((std\Wolf Eyes - Burned Mind.wav plot

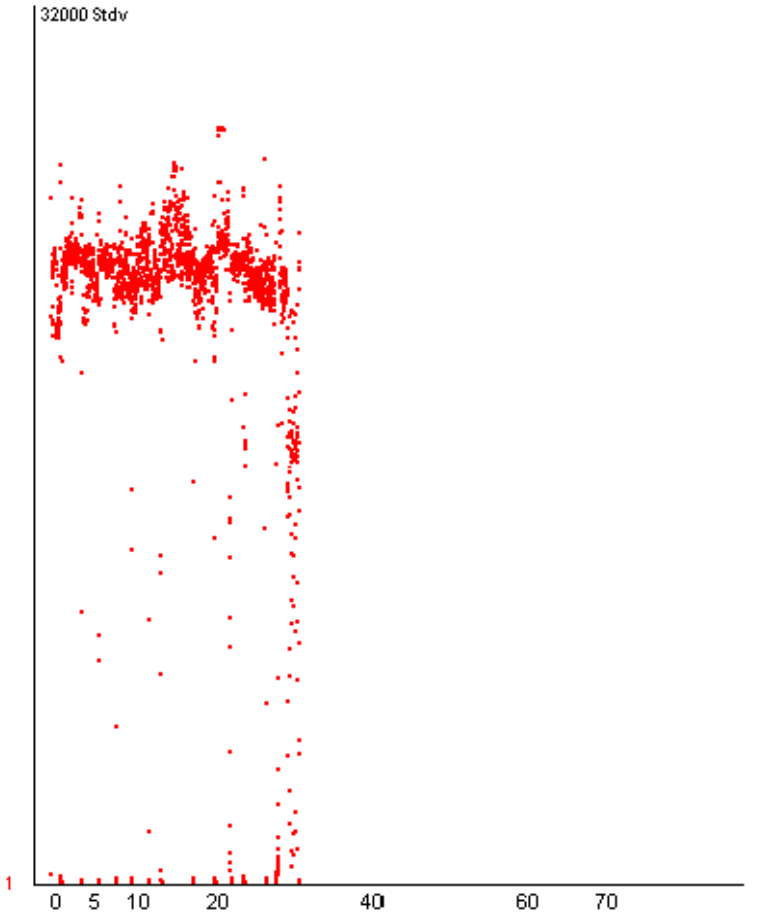
1 = ments\((((((((((((std\Yes - Awaken (Symphonic Live 2003).wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\LHD Limbs Of The Fawn.wav plot

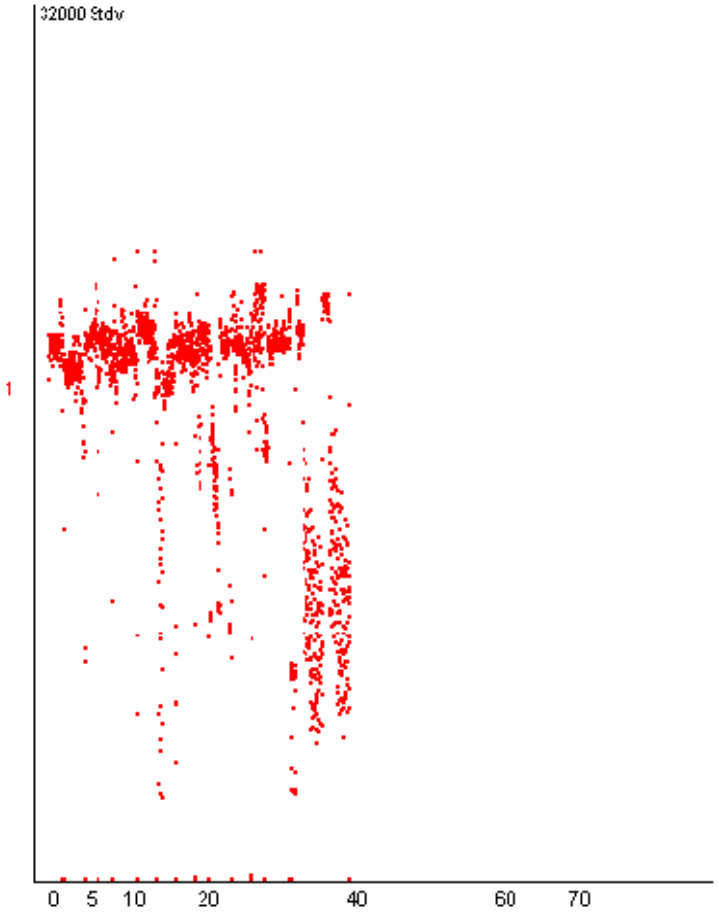

32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\K.K.NULL!!Live.wav plot



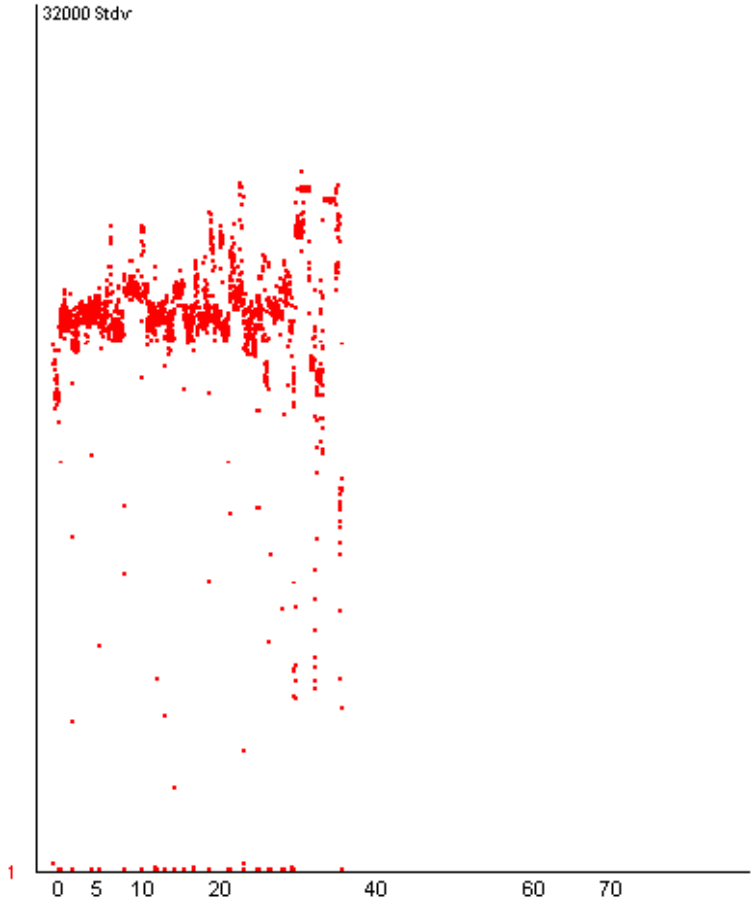
1 = E:\Noise\Masonna - Ejaculation Generater [full album].wav plot



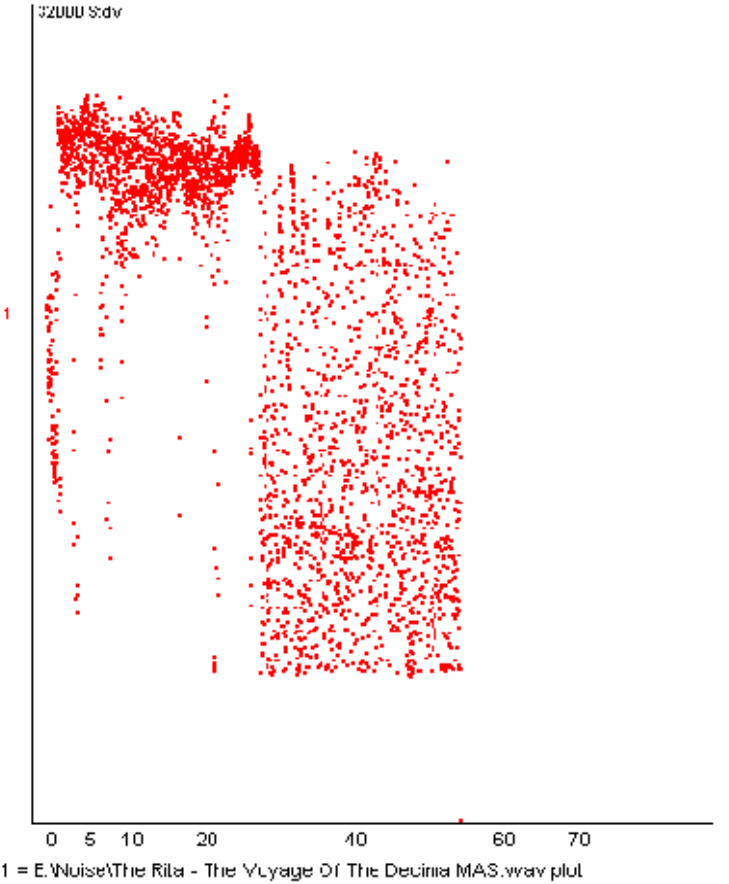
1 = E:\Noise\Masonna - Freak-out Electrolyze (Full Album).wav plot

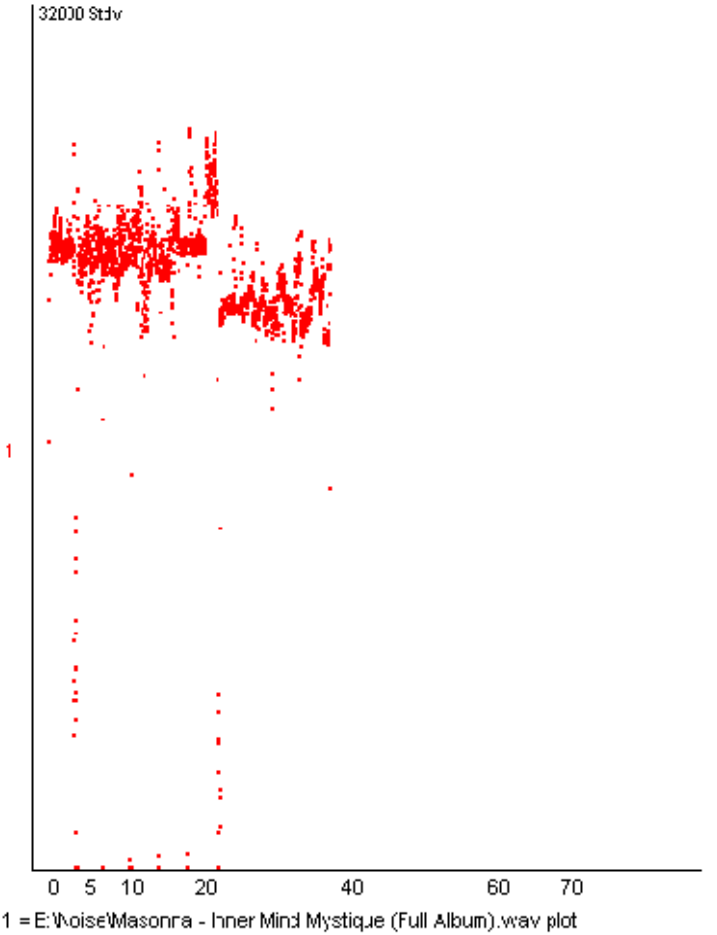


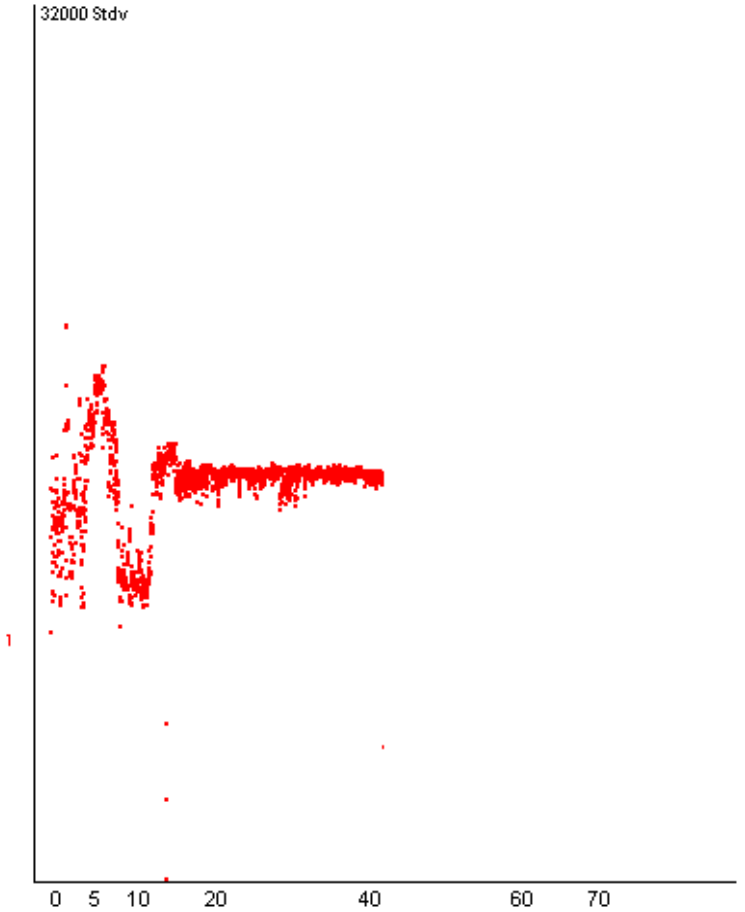
1 = E:\Noise\Masonna - Frecuency L.S.D. (Full Album).wav plot



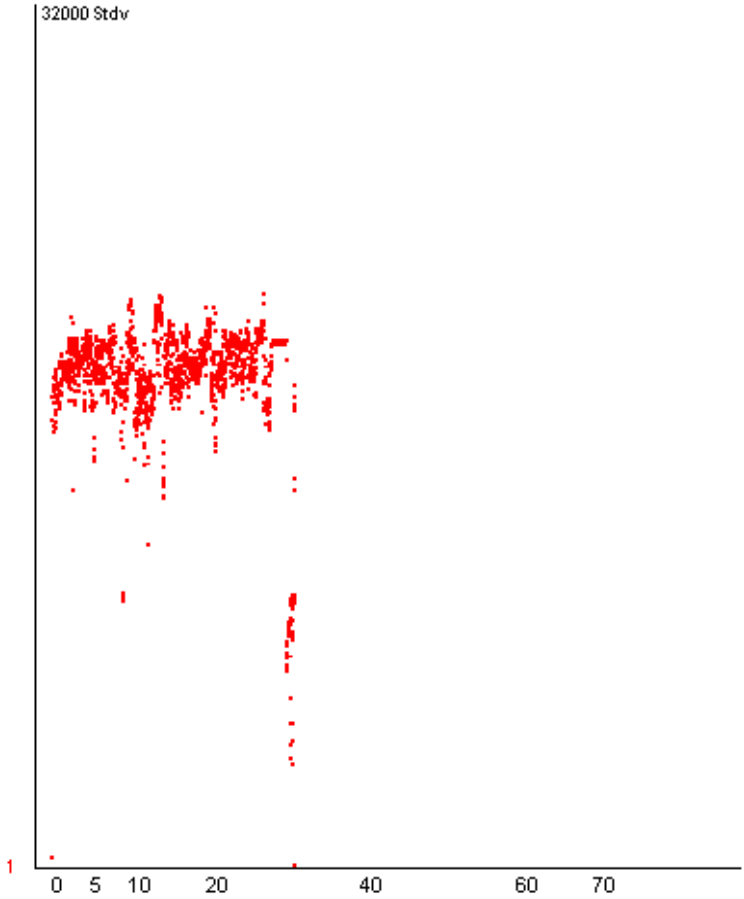
1 = E:\Noise\Masonna - Hyper Chaotic (Full Album).wav plot


32000 Sdv
1
0
5
10
20
40
60
70
1 = E:\Noise\The Rita - The Voyage Of The Decima MAS.wav plot

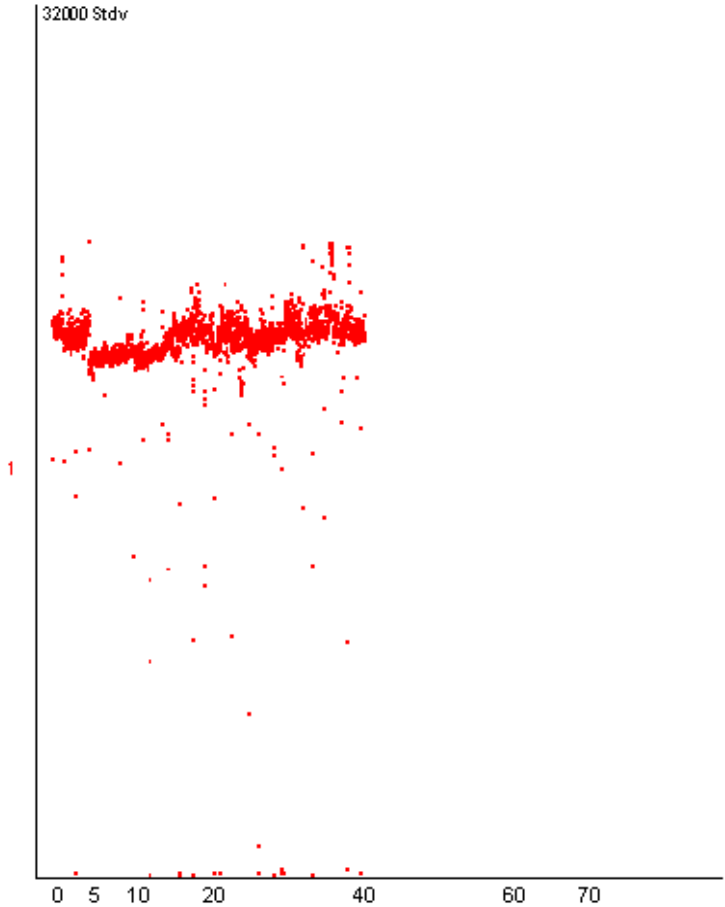

32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\Masonna - Inner Mind Mystique (Full Album).wav plot



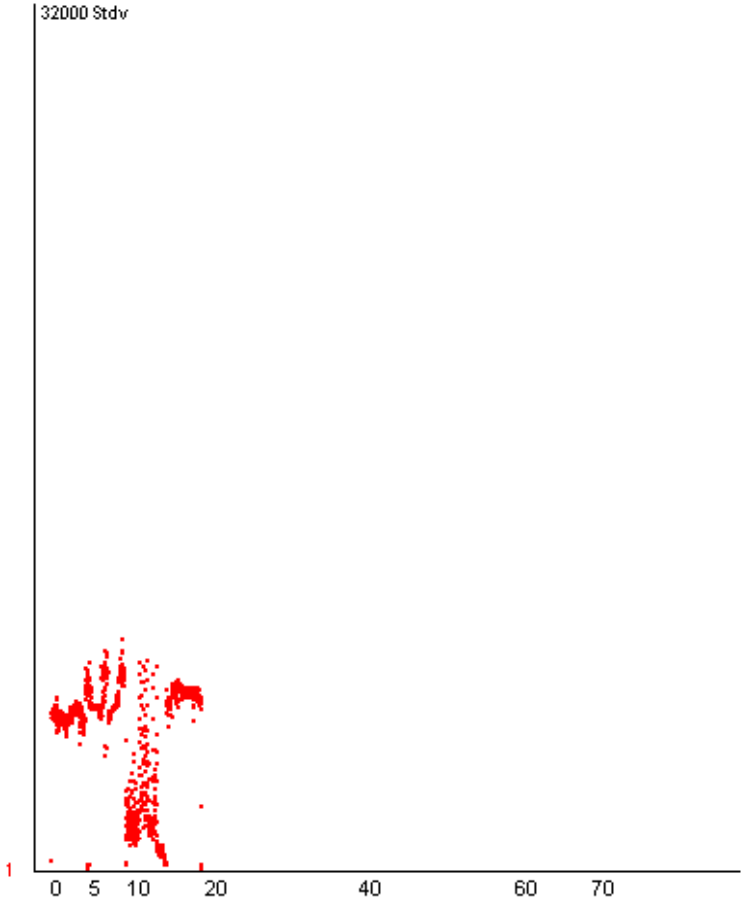
1 = E:\Noise\Masonna - Noisextra (Full Album).wav plot



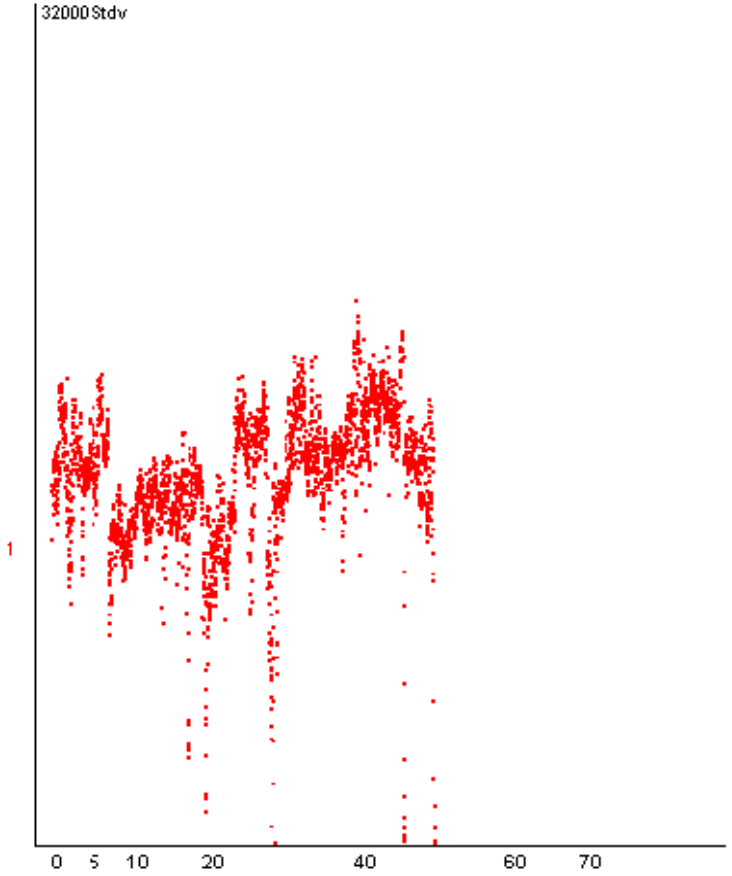
1 = E:\Noise\Masonna - Spectrum Ripper (Full Album).wav plot



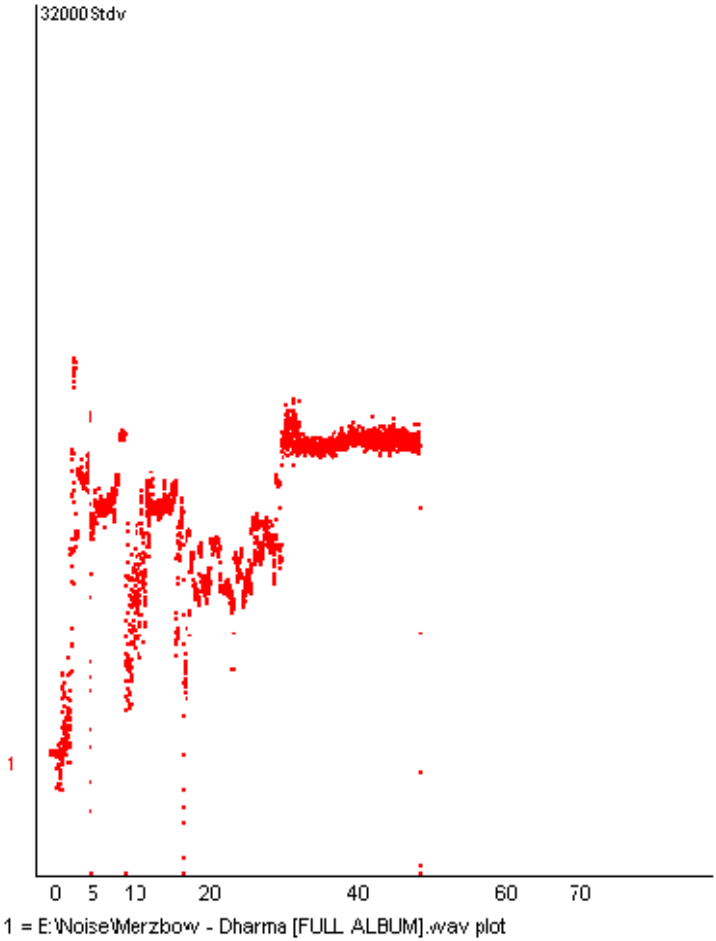
1 = E:\Noise\MASONNA - SHOCK ROCK.wav plot

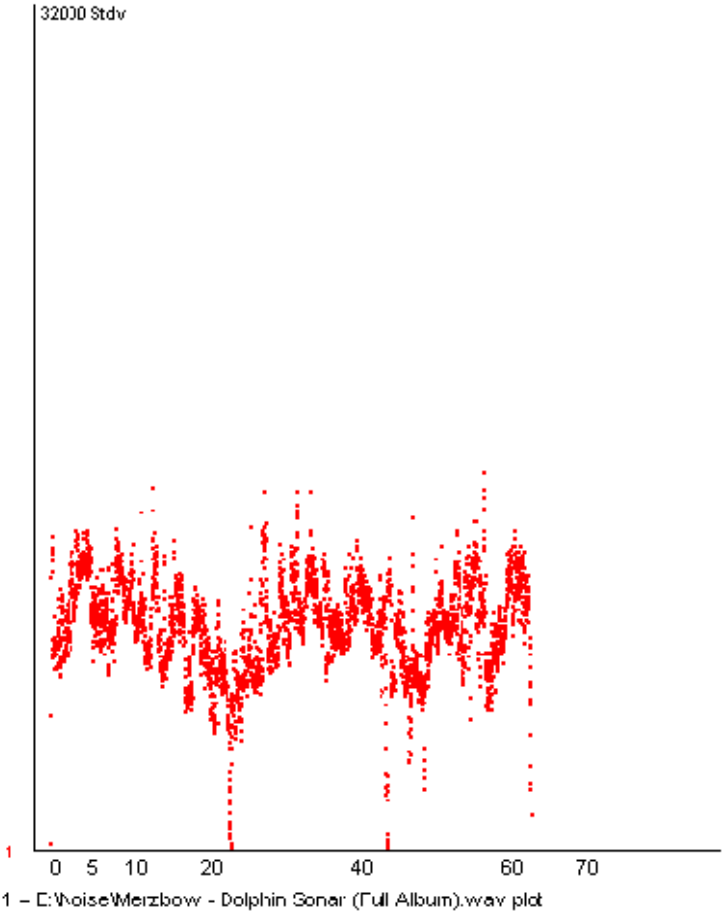


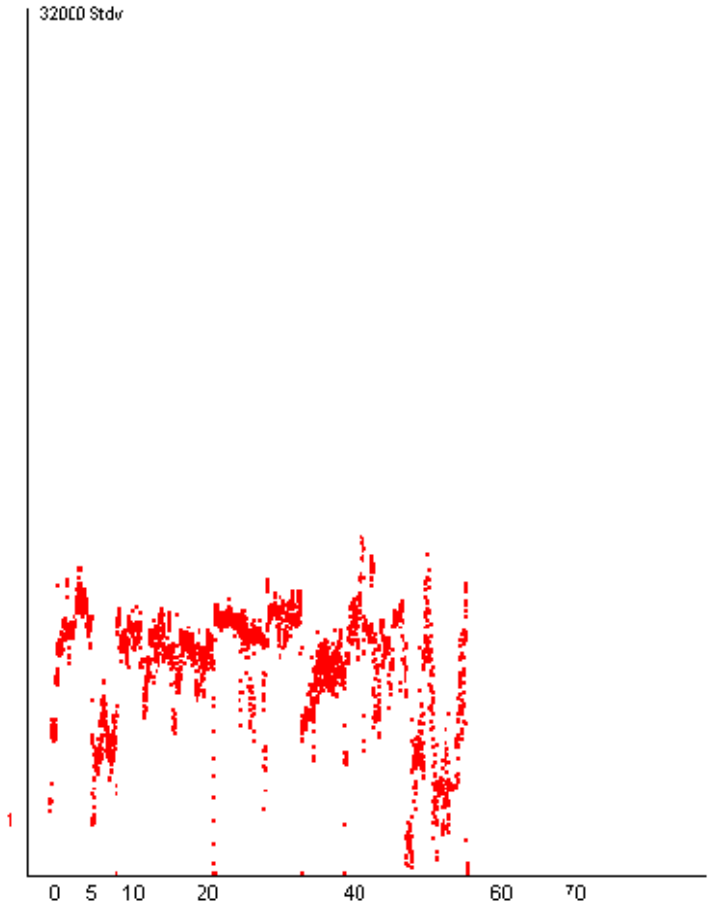
1 = E:\Noise\Masonna - The Passion Of Rubbers (FULL ALBUM).wav plot



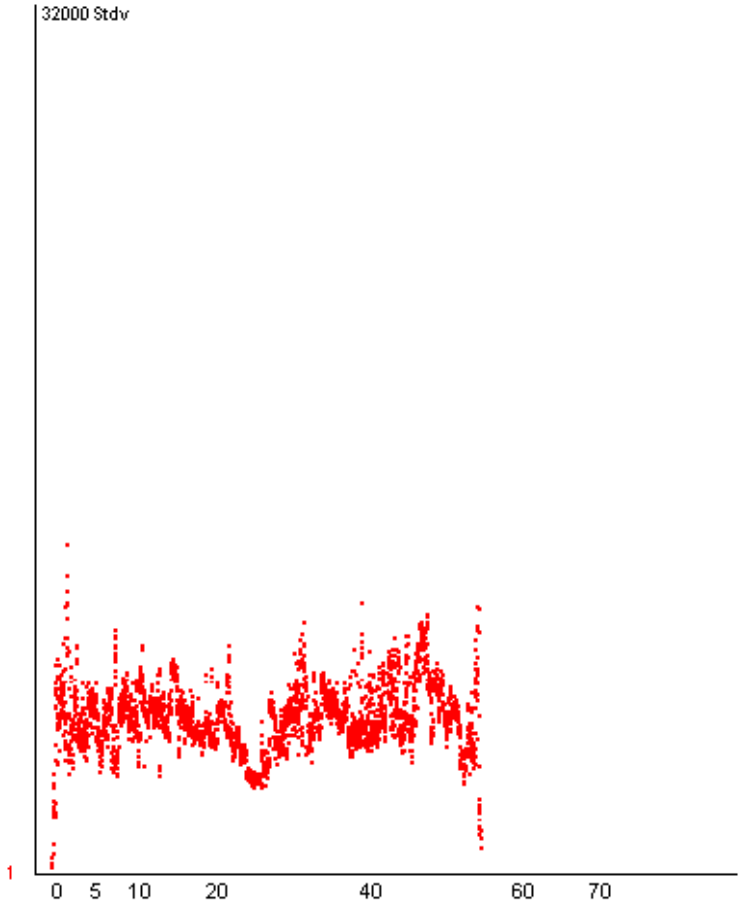
1 = E:\Noise\Merzbow - Aqua Necromancer (Full Album).wav pot


32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\Merzbow - Dharma [FULL ALBUM].wav plot

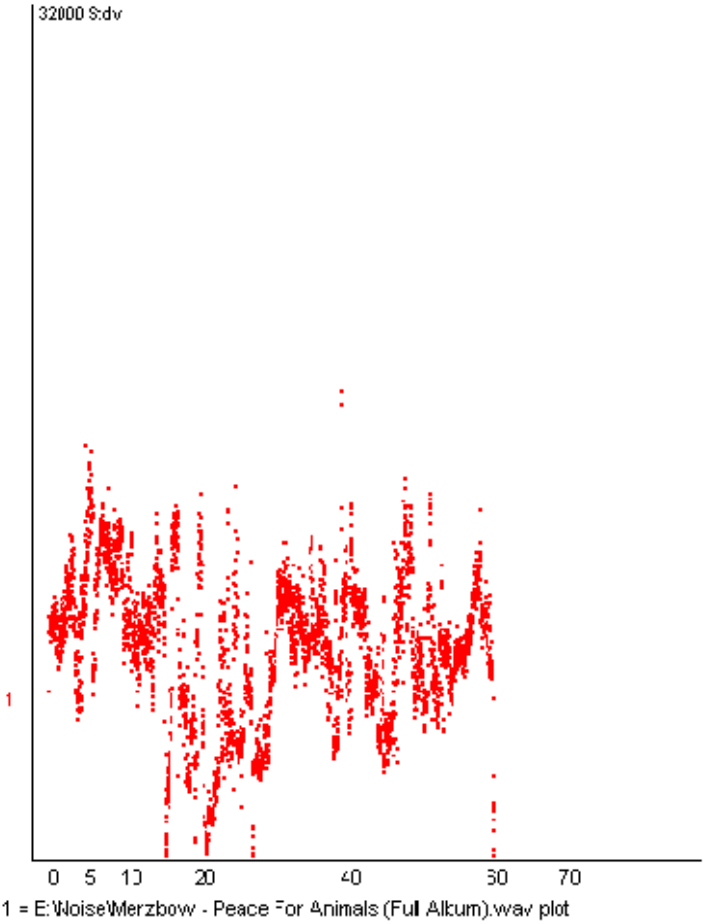

32000 Stdv
1
0
5
10
20
40
60
70
1 – E:\Noise\Merzbow - Dolphin Sonar (Full Album).wav plot

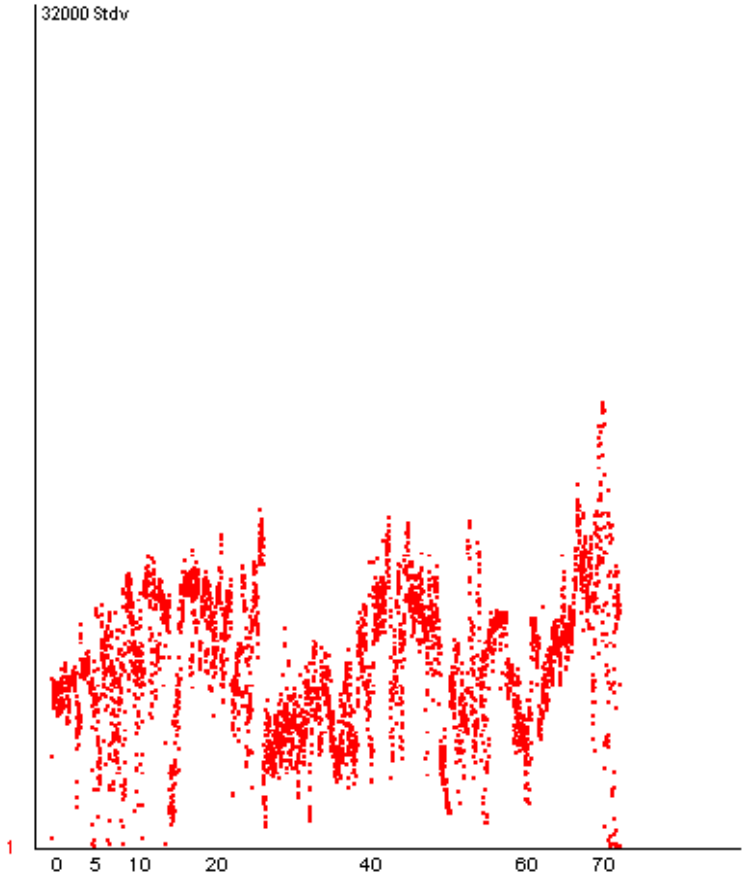


1 = E:\Noise\Merzbow - Merzbeat (Full Album).wav plot

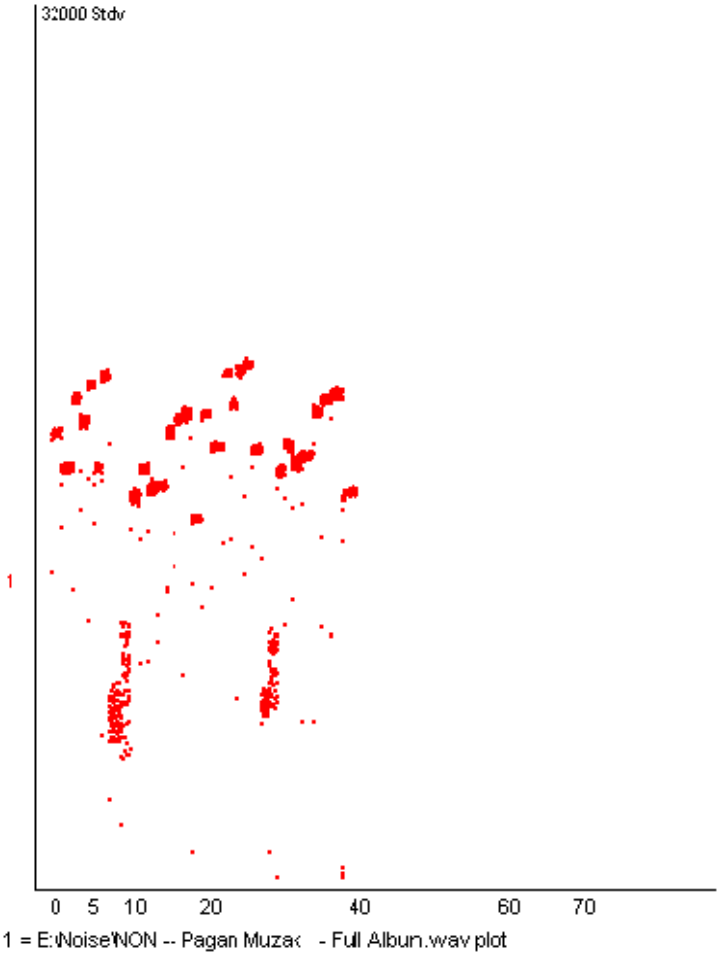


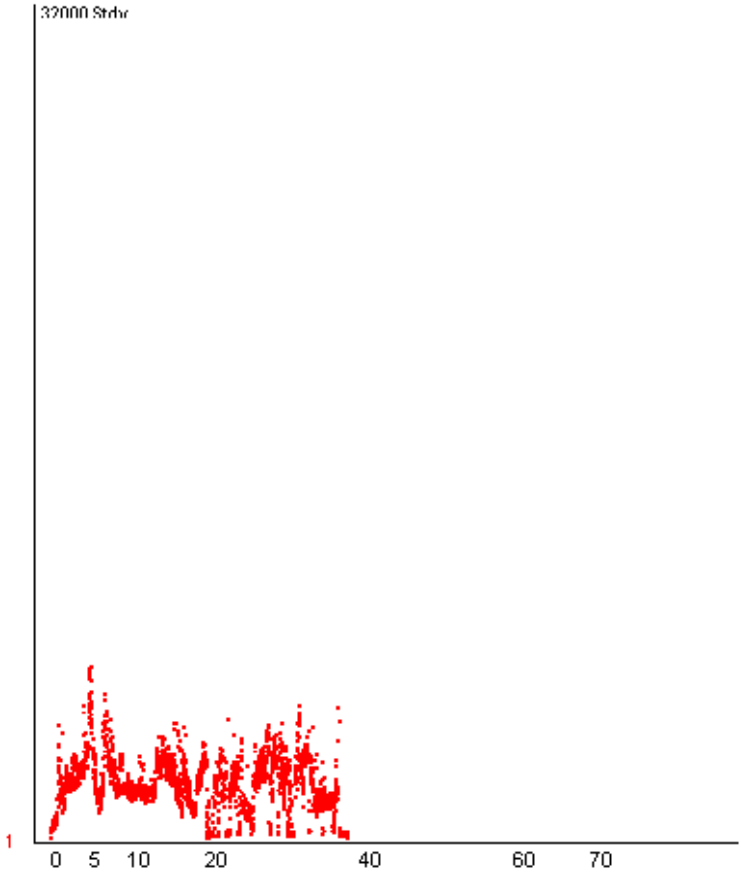
1 = E:\Noise\Merzbow - Live at Milan (10-14-12).wav plot


32000 Sdv
1
0
5
10
20
40
50
70
1 = E:\Noise\Merzbow - Peace For Animals (Full Album).wav plot

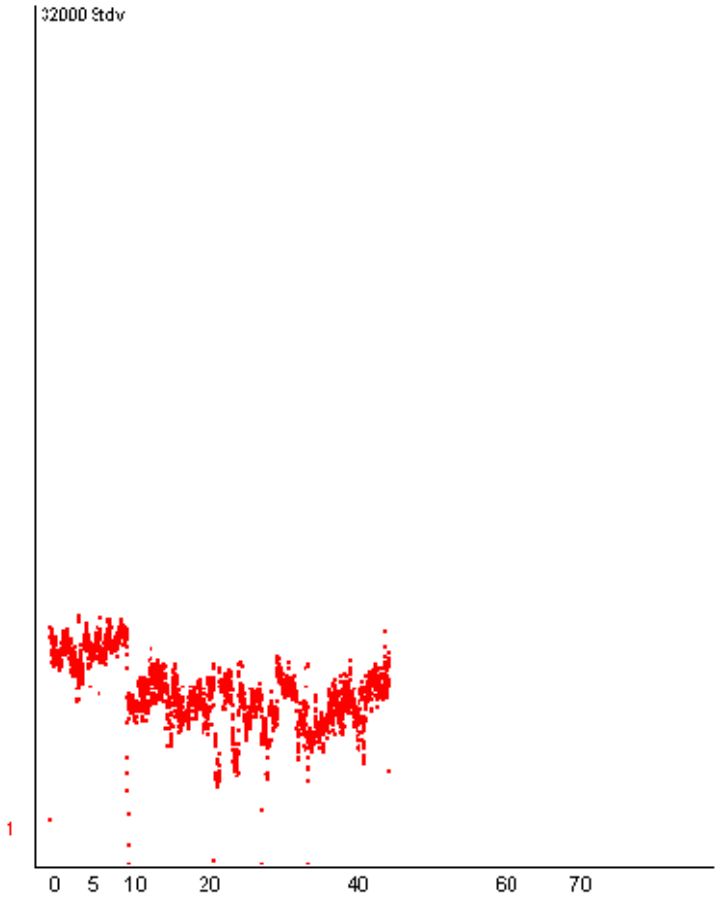


1 = E:\Noise\Merzbow - Rainbow Electronics (Full Album).wav plot

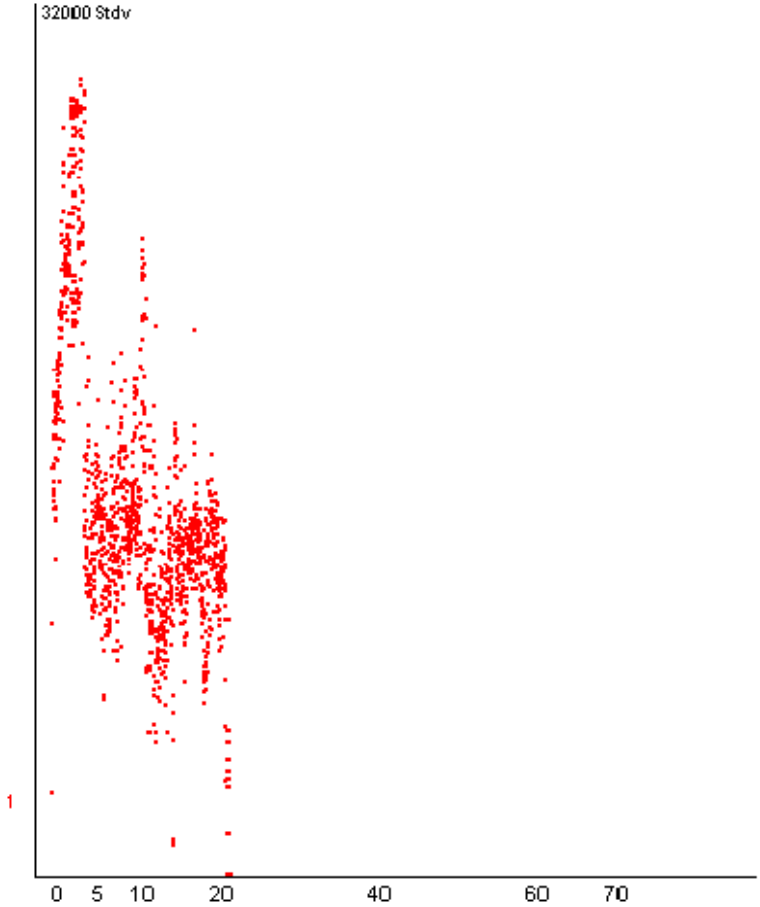

32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\NON -- Pagan Muzak - Full Album.wav plot



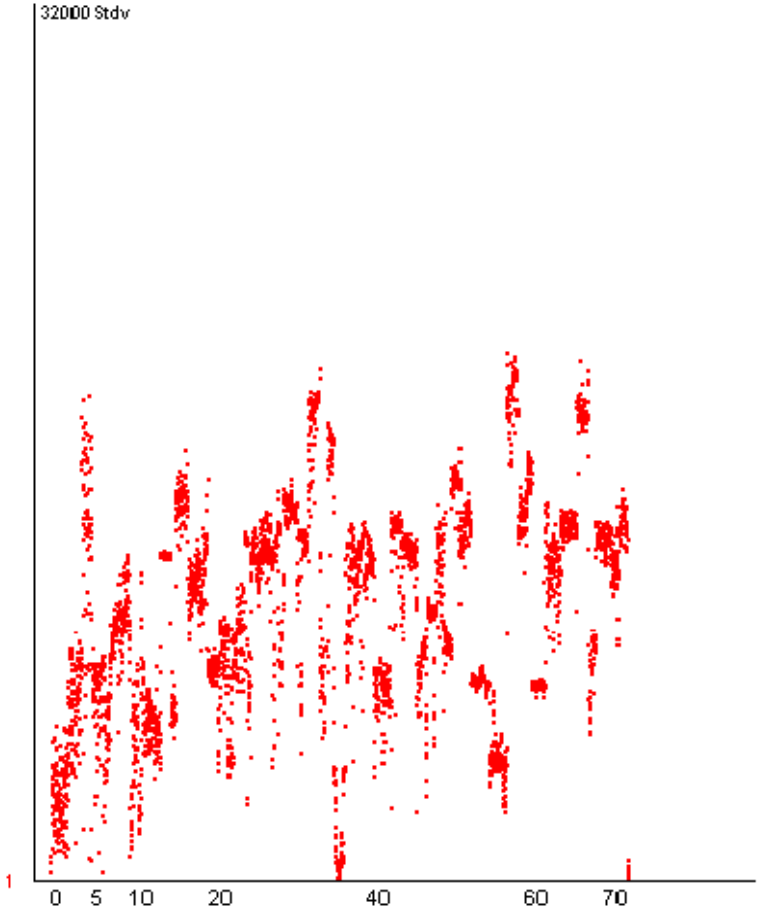
1 = se\Non - Boyd Rice - Düsseldorf 27.1.81 - Berlin 8.11.80.wav plot



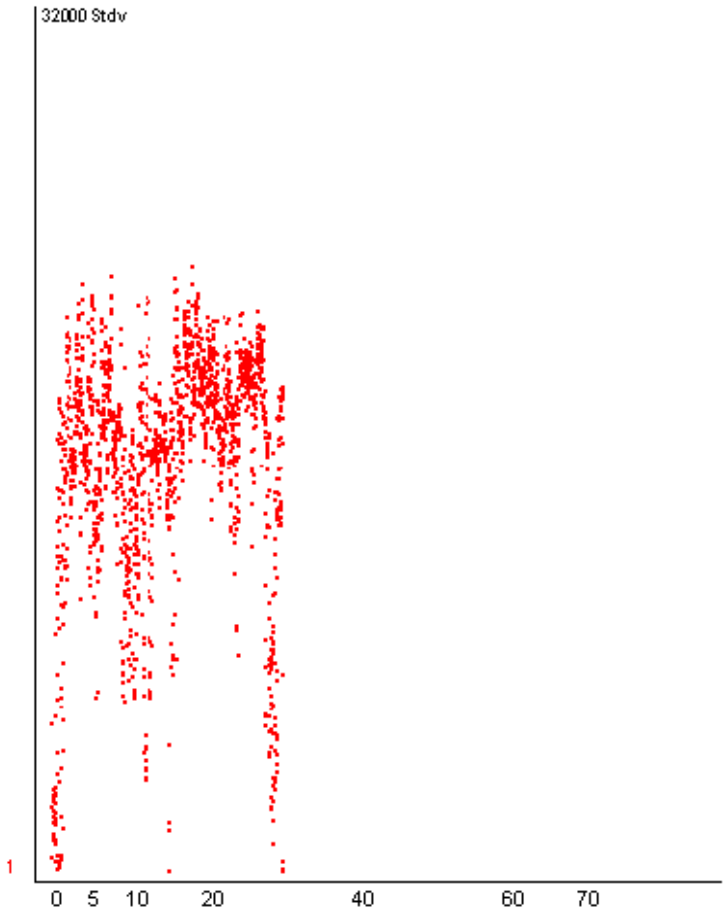
1 = E:\Noise\Pair Jerk - Retrogress .wav plot



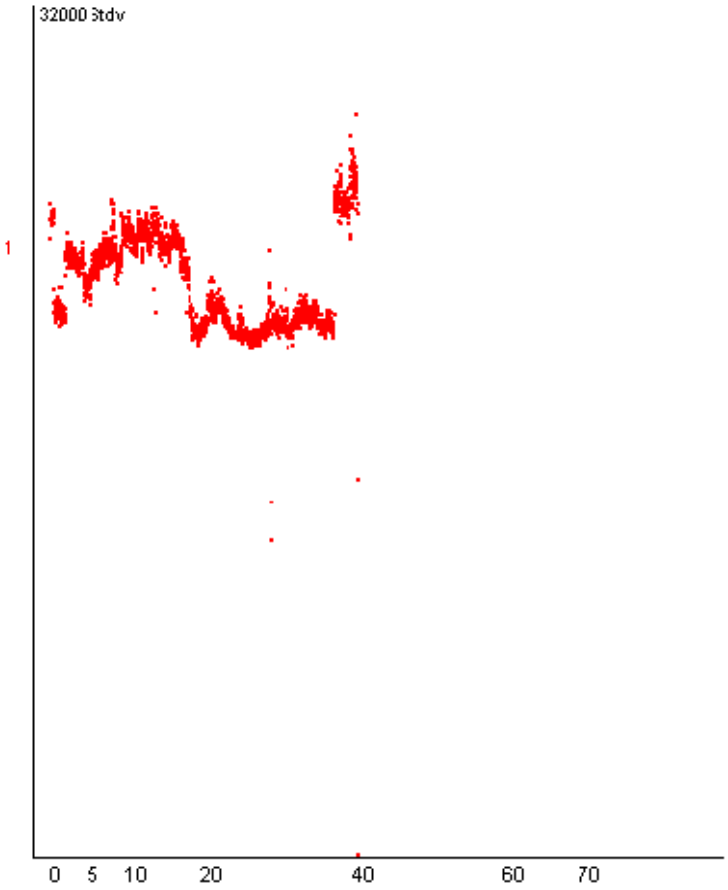
1 = E:\Noise\Prurient live at Screamscape.wav plot



1 = PAIN JERK - Russell Haswell's Mega Edit [Editions Mego].wav plot

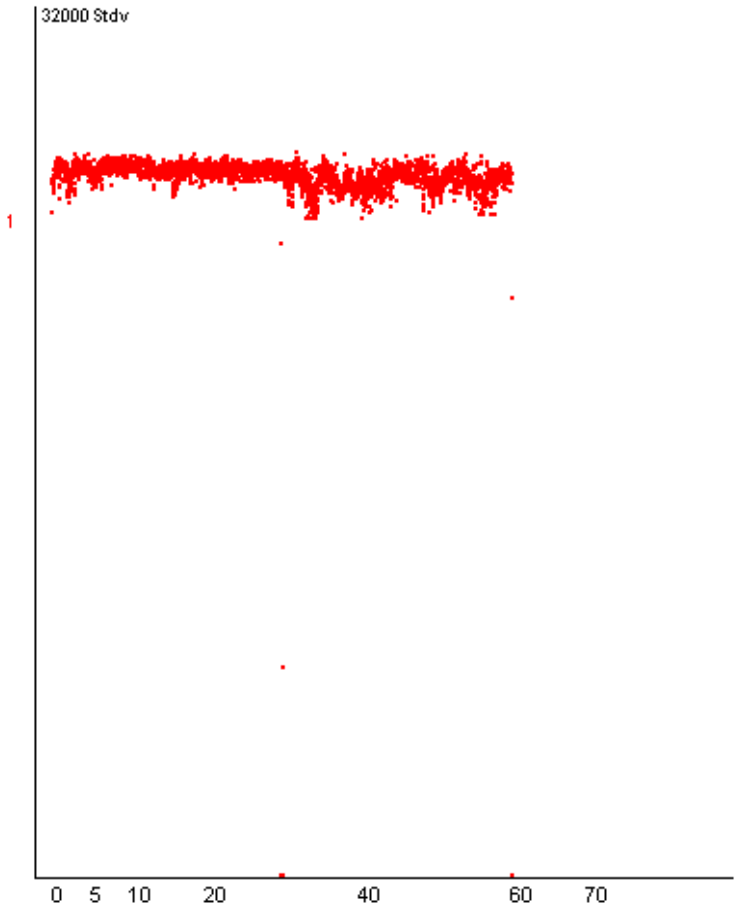


1 = E:\Noise\Richard Ramirez -Spinal Amputation.wav plot

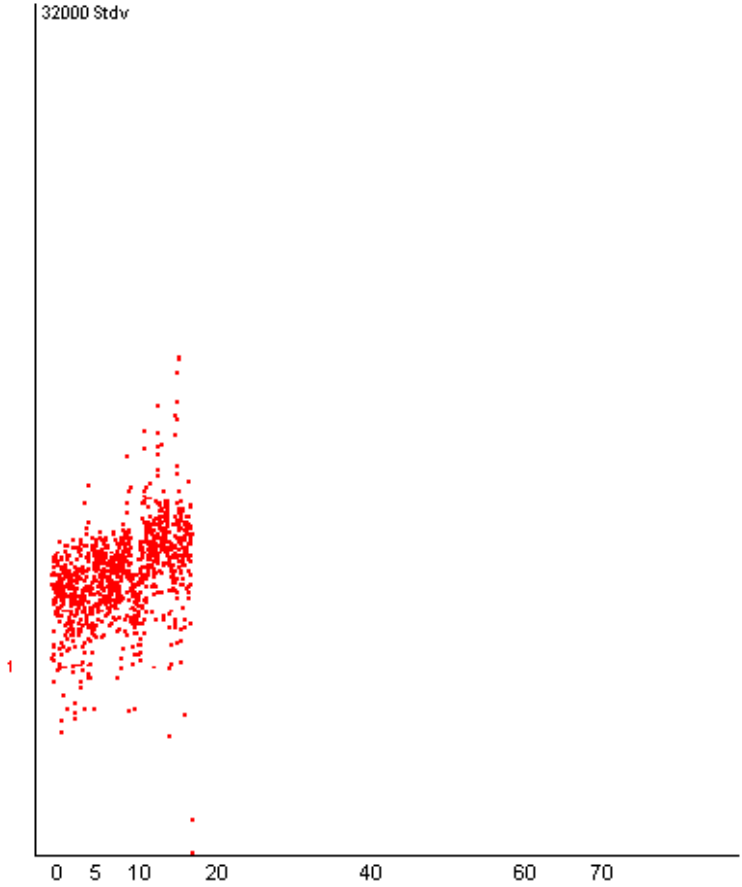


1 = he Cherry Point - Night of the Bloody Tapes (Full Album).wav plot

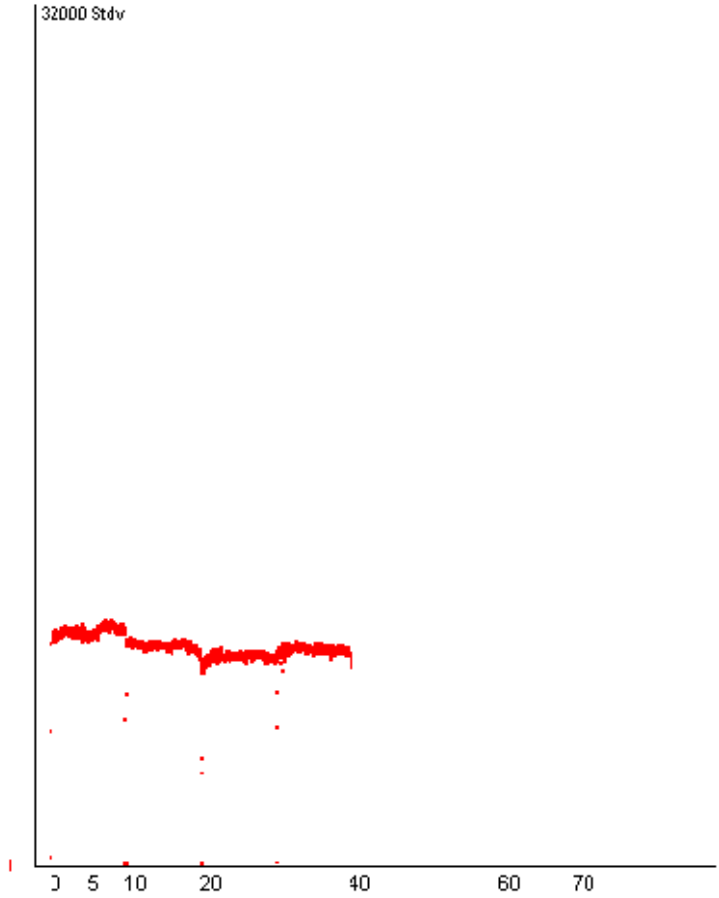
1 = E:\Noise\The Cherry Poirt _John Wiese White Gold.wav plot



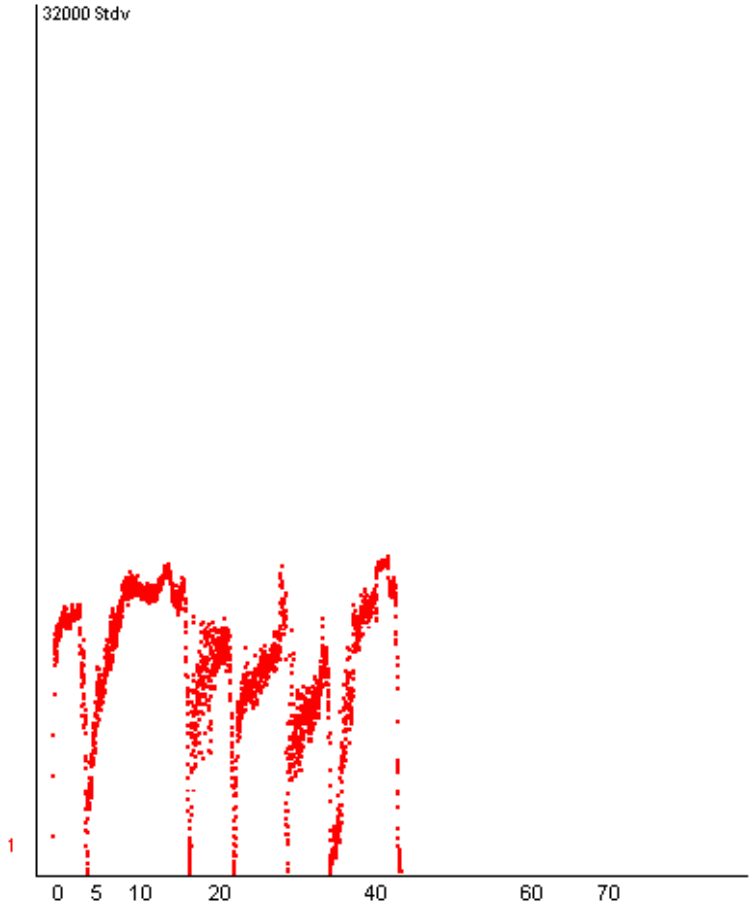
1 = E:\Noise\The Rita - Lake Depths Lurker.wav plot



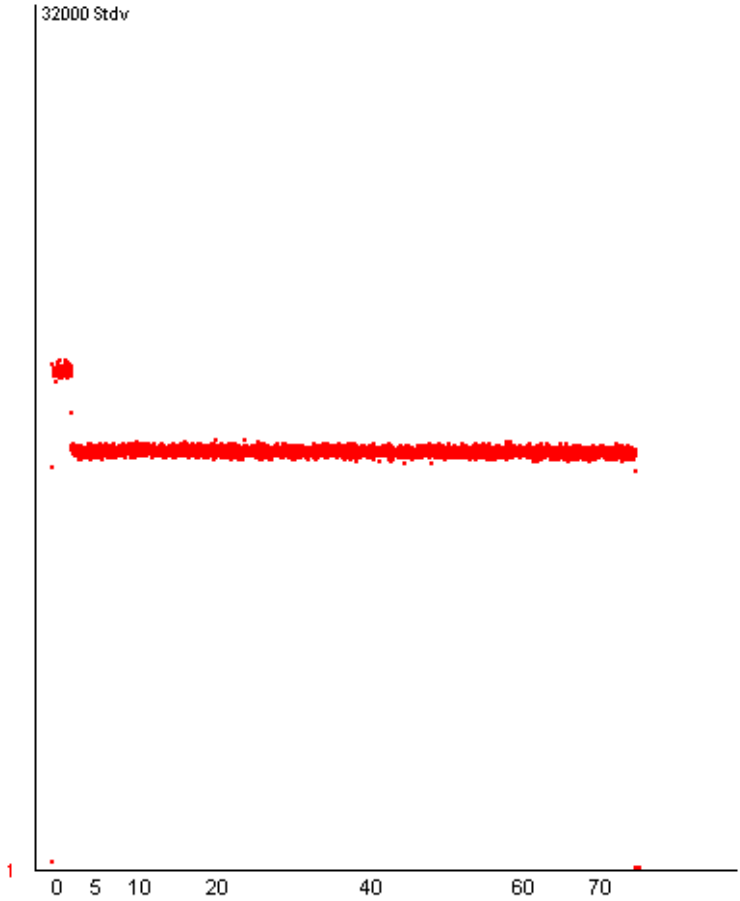
1 = E:\Noise\The Haters - Clici-Clic 1992.wav plot



1 = E:\Noise\The Rita - Revealing Leopard Skin.wav plot



1 = E:\Noise\Yellow Swans - Going Places [Full Album].wav plot

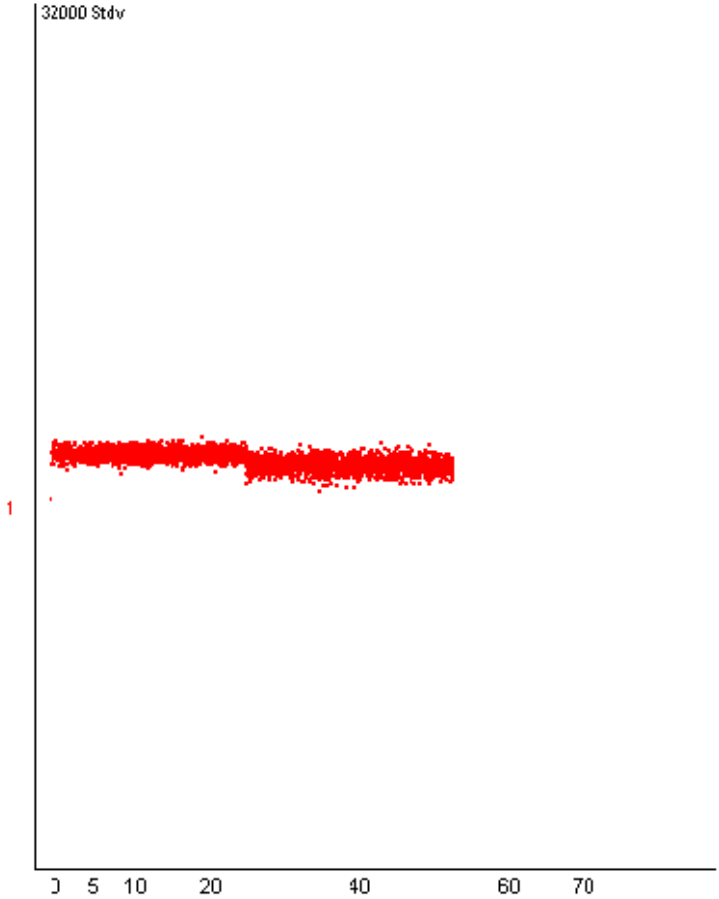


1 = E:\Noise\Vomir - Proanomie.wav plot

1 = E:\Noise\Vomir - No Entertainment.wav plot

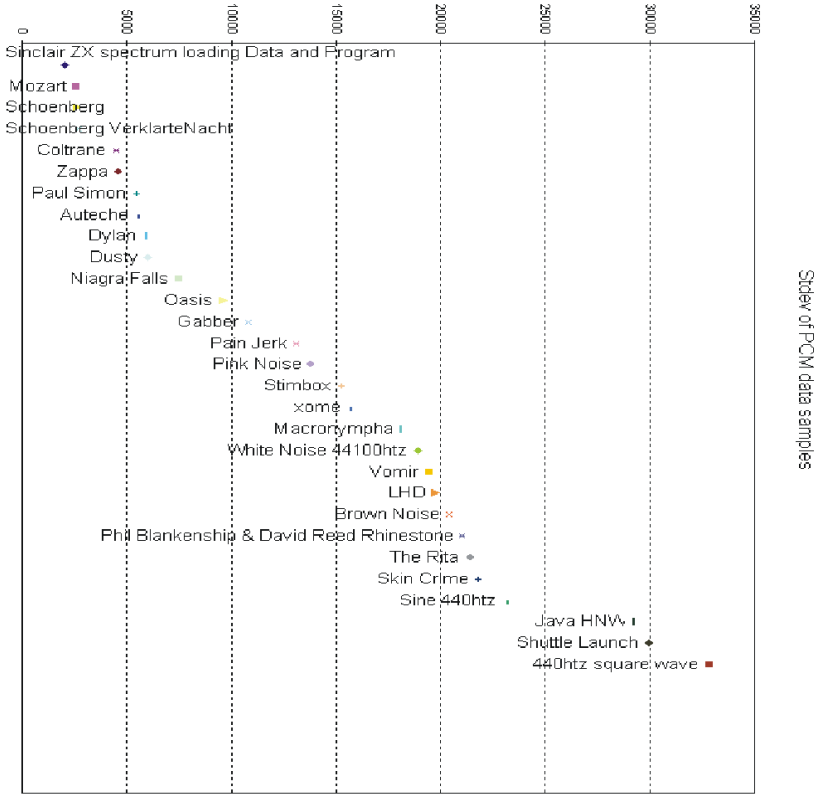
1 = E:\Noise\Vomir - Side B Untitled.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = E:\Noise\Vomir - Indecente.wav plot



1 = E:\Noise\Vomir - Black Bag.wav plot

The Original graph made by using Excel to calculate Stdevs of sound files. Single plots are made by taking the average of a number of random samples through each wave file. Note the errors in reference wavs of square and sine in this graph compared to those above. These are probably due to averaging of Stdevs in this case as opposed to plotting each second of the whole file.
Yet the trend of greater Stdevs for more noisy work is obvious even in this initial graph.



Names Index

Dunstable Works - 49
Electro House 2016 New EDM Dance Charts Songs - Club Music Remix - 47
Eno Brian - Textures - 22

Fripp and Eno – Swastika Girls - 52
Fushitsusha @ St John-at-Hackney 5-10-2012 - 53

Gamelan Music of Bali - 54
Gershwin - Concerto in F - 55
Gershwin - Rhapsody In Blue - 56
Glass Philip - Akhnaten (Complete) Act I (1-3) - 112
GX Jupitter Larsen - God Pussy - Untitled - 58

Hanatarash - Hanatarash 3 - 59
Hanatarash - Hanatarashi LP [1985] - 60
Hanatarasha - 61
Hendrix - Live Stockholm - 62
Hijokaidan- King of Noise - 63
Hijokaidan – Romance - 64
Hijokaidan - Viva Angel [Full album] - 65
Hyden – Symphony no. 6 - 70
Hyden- Symphony no. 60 - 71

Incapacitants - Default Standard - 72
Incapacitants - Eat! Meat!! Manifesto!!! - 73
Incapacitants - Fabrication - 74
Incapacitants - Ministry Of Foolishness - 75
Incapacitants - New Movements in CMPD - 76
Incapacitants - repo - 77
Incapacitants - Stone River - 78
Iron Maiden - Iron Maiden - 79
Ives Charles - Concord Sonata - 80
Ives Charles Symphony – No 1 James Sinclair - 66

Jarre - Equinoxe - 81
JLIAT- BAD Music 2 - 16
JLIAT- BAD Music 5 - 15
JLIAT – Drone 8-10-16 - 84
JLIAT – HNW 27-4-14 - 68
JLIAT – HNW 28 -5-13 - 67
JLIAT – HNW 29-10-16-a - 82
JLIAT – HNW 29-10-16-b - 83
JLIAT – HNW Java-BAD - 69
JLIAT- HN -7-12-14 - 1
JLIAT- HN-22-10-2016 - 85

K.K. Null & Fred Frith Guy Jim O'Rourke - Guitar Organism - 87
K.K. Null- Terminal Beach 1996 - 86
K.K.NULL!!Live - 143

Page Index

46 Cream - Live at The Albert Hall
47 Electro House 2016 New EDM Dance Charts Songs Remix
48 De Monte and the Habsburgs
49 Dunstable Works
50 Radigue Elaine - Islas
51 Beaulieau Emil America's Greatest Living Noise Artist
52 Fripp and Eno – Swastika Girls
53 Fushitsusha @ St John-at-Hackney 5-10-2012
54 Gamelan Music of Bali
55 Gershwin - Concerto in F
56 Gershwin - Rhapsody In Blue
57 Ribot M. Musical Improvisation in the Marlene Dumas Exhibition
58 GX Jupitter Larsen - God Pussy - Untitled
59 Hanatarash - Hanatarash 3
60 Hanatarash - Hanatarashi LP [1985]
61 Hanatarasha
62 Hendrix - Live Stockholm
63 Hijokaidan- King of Noise
64 Hijokaidan – Romance
65 Hijokaidan - Viva Angel [Full album]
66 Ives Charles Symphony – No 1 James Sinclair
67 JLIAT – HNW 28 -5-13
68 JLIAT – HNW 27-4-14
69 JLIAT – HNW Java-BAD
70 Hyden – Symphony no. 6
71 Hyden- Symphony no. 60
72 Incapacitants - Default Standard
73 Incapacitants - Eat! Meat!! Manifesto!!!
74 Incapacitants - Fabrication
75 Incapacitants - Ministry Of Foolishness
76 Incapacitants - New Movements in CMPD
77 Incapacitants - repo
78 Incapacitants - Stone River
79 Iron Maiden - Iron Maiden
80 Ives Charles - Concord Sonata
81 Jarre - Equinoxe
82 JLIAT – HNW 29-10-16-a
83 JLIAT – HNW 29-10-16-b
84 JLIAT – Drone 8-10-16
85 JLIAT- HN-22-10-2016
86 K.K. Null- Terminal Beach 1996
87 K.K. Null & Fred Frith Jim O'Rourke - Guitar Organism
87 Khusugtun Music
89 Klein – Sonatas
90 La Monte Young - Marian Zazeela - The Black Album -1969
91 La Monte Young - Rag Bhairava (Excerpt)
92 Lou Reed - Metal Machine
93 Mahler – Symphony no. 1
94 Mahler – Symphony no.2

95 Merzbow – Pulse Demon
96 Merzbow - 1930
97 Davis Miles - Bitches Brew (1970)
98 Monastic Chant 12th13th C
99 Morrocco Gnawa Music
100 Subotnick Morton - Silver Apples Of The Moon
101 Subotnick Morton - The Wild Beasts -1978
102 Mozart Symphony – no. 40 K550
103 New Blockaders – simp in Xmaj
104 Oasis – Whats the Story
105 Oldfield – Tubular bells II Live 1992
106 Reference Sources from nature (SETI PI PRIMES)
107 Pain Jerk – Neon Kid 1995
108 Palestrina – Motets
109 Part Arvo - Symphony No. 4 'Los Angeles' Proms 2010
110 Part Arvo - Tabula rasa
111 Partch Harry - And On The Seventh Day Petals Fell In Petaluma
112 Glass Philip - Akhnaten (Complete) Act I (1-3)
113 Reference Sources Waves (White noise WN filtered Sine Sq Saw)
114 Webern- op.28
115 Rolling Stones- Still life
116 Schaeffer and Pierre Henry - orp 53
117 Schoenberg – Op. 4
118 Schoenberg – Piano Concerto Op. 42
119 Sitar -Bose
120 Spontaneous Music Ensemble - Boileau Road (1980)
121 Spontaneous Music Ensemble - Familie (1968)
122 Reich Steve - Different Trains
123 Reich Steve - Music for 18 Musicians
124 Stockhaussen – Gesang Der
125 Stravinsky – Symphony in C maj
126 Tallis - 10 motets
127 Tallis – Lamentations
128 Tavener – The Protecting Veil
129 Tchaikovsky - Symphony no6 (Pathetique)
130 Riley Terry & Friends - 'In C' Boiler Room Amsterdam
131 Riley Terry - Persian Surgery Dervishes
132 The Rita - Rebianchilized and Anna Christie
133 Tilbury - Parker in which …
134 Tudor David - Phonemes (1981)
135 Varèse Edgard- Amériques
136 Varèse Edgard – intercontemporian
137 Vomir - Causation
138 Vomir – untitled
139 Zorn - Marcic
140 Wolf Eyes – Burned Mind
141 Yes -Awaken Symphonic Live 2003
142 LHD - Limbs Of The Fawn
143 K.K.NULL!!Live

144	Masonna - Ejaculation Generater
145	Masonna - Freak-out Electrolyze
146	Masonna - Frequency L.S.D.
147	Masonna - Hyper Chaotic
148	The Rita - The Voyage Of The Decima MAS
149	Masonna - Inner Mind Mystique
150	Masonna - Noisextra
151	Masonna - Spectrum Ripper
152	Masonna – Shock Rock
153	Masonna - The Passion Of Rubbers
154	Merzbow - Aqua Necromancer
155	Merzbow - Dharma
156	Merzbow - Dolphin Sonar
157	Merzbow - Merzbeat
158	Merzbow - Live at Milan (10-14-12)
159	Merzbow - Peace For Animals
160	Merzbow - Rainbow Electronics
161	NON -Pagan Muzak
162	Non - Boyd Rice - Desseldorf 27.1.81 - Berlin 8.11.80
163	Pain Jerk - Retrogress
164	Prurient- live at Screamscape
165	PAIN JERK - Russell Haswell's Mega Edit
166	Richard Ramirez - Amputation
167	The Cherry Point - Night of the Bloody Tapes
168	The Cherry Point & John Wiese - White Gold
169	The Rita - Lake Depths Lurker
170	The Haters - Clici-Clic 1992
171	The Rita - Revealing Leopard Skin
172	Yellow Swans - Going Places
173	Vomir - Proanomie
174	Vomir - No Entertainment
175	Vomir - Side B Untitled
176	Vomir - Indecente
177	Vomir - Black Bag

Appendices:

Brief explanation of PCM data.

Java source code for producing live and file HNW.

Plots of Java HNW .

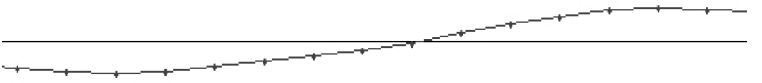
Brief Explanation of how sound is stored digitally as PCM data on Audio Cds.

A sound wave is converted into numeric data that 'models' the shape.

Here is a picture of a sound wave..

As we zoom in we can see how its made of discrete steps, each of which is represented on the CD as a number.

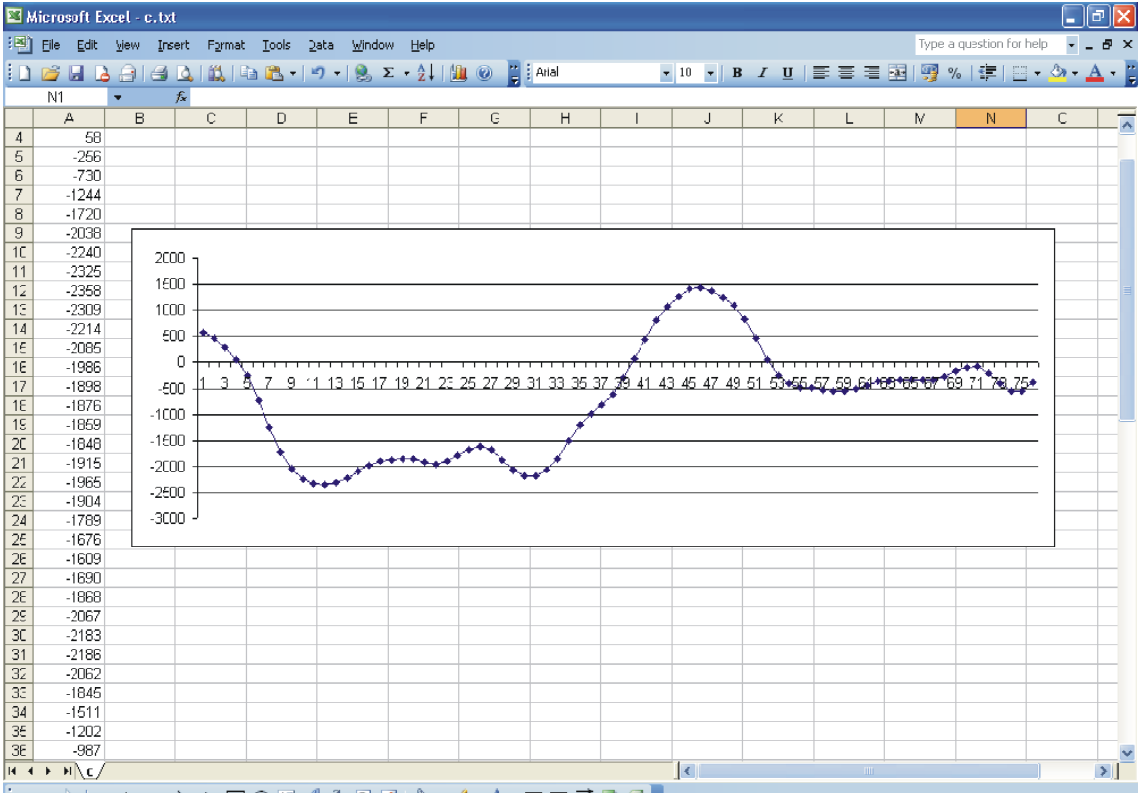
Each dot above is written to a file as a number. So the wave form is digitally recorded.



The CD Player reads these numbers and converts them back into an electrical signal which is then amplified and passed to the speakers.
The speaker cones convert the electrical energy into sound waves.
So the corresponding numbers relate to voltages which in turn relate to the sound we hear.

This means that its possible to 'make' sound files by just writing numbers to a file using a program. The following shows methods Java They write random numbers which produce random sound- i.e. Noise.

The rate at which these random numbers change makes the noise sound harsher if sufficient delay is made so that the human ear can detect these changes.
Each second of sound is made of 44100 samples and so altering each randomly makes for a smooth sounding white noise.
The more course the transition the harsher sounds the noise.


Microsoft Excel - c.txt
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
N1
A B C D E F G H I J K L M N C
4 58
5 -256
6 -730
7 -1244
8 -1720
9 -2038
10 -2240
11 -2325
12 -2358
13 -2309
14 -2214
15 -2085
16 -1986
17 -1898
18 -1876
19 -1859
20 -1848
21 -1915
22 -1965
23 -1904
24 -1789
25 -1676
26 -1609
27 -1690
28 -1868
29 -2067
30 -2183
31 -2186
32 -2062
33 -1845
34 -1511
35 -1202
36 -987
2000
1500
1000
500
0
-500
-1000
-1500
-2000
-2500
-3000

```
1 //LISTING OF JAVA PROGRAM TO PLAY HNW
2
3
4 import javax.sound.sampled.*;
5 import java.util.*;
6 public class HNW {
7  //The following are general instance variables
8  // used to create a SourceDataLine object.
9  AudioFormat audioFormat;
10  AudioInputStream audioInputStream;
11  SourceDataLine sourceDataLine;
12  //The following are audio format parameters.
13  float sampleRate = 16000.0F;
14  //Allowable 8000,11025,16000,22050,44100
15  int sampleSizeInBits = 16;
16  //Allowable 8,16
17  int channels = 1;
18  // MONO!
19  //Allowable 1,2
20  boolean signed = true;
21  //Allowable true,false
22  boolean bigEndian = true;
23  //Allowable true,false
24  //
25  // size of number of samples to smooth
26  // lager = more coarse sounds
27  static int harshness = 60;
28  static int noiselen = 0;  // length of piece
29  //-----------------------------------------//
30  public static void main(String args[]){
31   if (args.length > 0) {
32    harshness = Integer.parseInt(args[0]);
33   }// end of if   if (args.length > 1) {
34    noiselen = Integer.parseInt(args[1]);
35   }// end of if
36   new HNW();
37  }//end main
38  //-----------------------------------------//
39  public HNW(){//constructor
40   playData();
41  }//end constructor
42
43  private void playData() {
44   try{
45    //Get the required audio format
46    audioFormat = new AudioFormat(
47      sampleRate,
48      sampleSizeInBits,
49      channels,
50      signed,
51      bigEndian);
52    //Get info on the required data line
53    DataLine.Info dataLineInfo =
54     new DataLine.Info(
55       SourceDataLine.class,
56       audioFormat);
57    //Get a SourceDataLine object
58    sourceDataLine =
(SourceDataLine)AudioSystem.getLine(dataLineInfo);
```

```
59    new ListenThread().start();
60   }catch (Exception e) {
61    e.printStackTrace();
62    System.exit(0);
63   }//end catch
64  }//end playData
65  //=============================================//
66  class ListenThread extends Thread{
67   byte playBuffer[] = new byte[200];
68   // the size of this buffer = length of sound
69   public void run(){
70    // need to generate random numbers
71    Random generator = new Random();
72    // harshness the higher this value the more rough / harsh
73    double ss[] = new double[harshness];
74    double sa;
75    short s; // signed two byte integer  for PCM data
76    try{
77     sourceDataLine.open(audioFormat);
78     sourceDataLine.start();
79     for(int z = -1; z < noiselen; z++){ //make thisbig for long
pieces or
80      // if noiselen was 0 from commandline run forever!
81      // note only play half buffer to stop gaps???
82      if (noiselen < 1) z = -2;
83      //----------------------- loop ------------------------
84      for(int cnt = 0; cnt < 100; cnt = cnt + 2 ){
85       // for testing so i can see the data
86      //  for(int cnt = 0; cnt < 100; cnt = cnt + 2 ){
87       int r = generator.nextInt();
88       s = (short) r;
89       // store new sample - roll out others
90       For(int x = 0; x < harshness-1; x++){
91        ss[x] = ss[x+1];
92       } // end of x for
93       ss[harshness-1] = (short)s;
94       // average  the last harshness number of samples
95       // remove the higher pitches
96       sa = 0;
97
98       for(int x = 0; x < harshness; x++){
99        sa = sa + ss[x];
100       }
101       sa = sa / harshness;
102       s = (short) sa;
103       //============ Process  for new sounds ========
104       //    loudness====  !!!!
105       // basically no wimpy numbers - only fat ones
106       if (s > 0 & s < 30000) {s = (short) (s + 30000);}
107       if (s < 0 & s > -30000) {s = (short) (s - 30000);}
108
109       //   System.out.println(s);
110       //=============== End of Process ============
111      // as the output stream is a byte array load the low/high
integer
112       byte hexBase ; // A byte of all ones
113       hexBase = (byte) 255;
114       byte b1 = (byte) (hexBase & s);
115       byte b2 = (byte) ((hexBase <<(8)& s) >> 8);
116       //System.out.println(b1);
```

```
      playBuffer[cnt] = b2 ;
      playBuffer[cnt+1] = b1 ;
     }
     sourceDataLine.write(playBuffer, 0, 100);
    } // end of loop
    //Block and wait for internal buffer of the
    // SourceDataLine to become empty.
    sourceDataLine.drain();
    //Finish with the SourceDataLine
    sourceDataLine.stop();
    sourceDataLine.close();

   }catch (Exception e) {
    e.printStackTrace();
    System.exit(0);
   }//end catch
  }//end run
 }//end inner class ListenThread
}//end outer class
```

```
1  //  WRITES  TO  A  .SND  SOUND  FILE
2
3  import  java.util.*;
4  import  java.io.FileOutputStream;
5
6  public  class  hnwfile{
7   //  16000  16  bit  mono  signed  big  endian
8   //  size  of  number  of  samples  to  smooth
9   //  lager  =  more  coarse  sounds
10   static  int  harshness  =  60;
11   //  200  *  100  =  10  minutes
12   static  int  noiselen  =  200  *  100; //  length of piece
13  //-----------------------------------------//
14   public  static  void  main(String  args[]){
15    new  hnwfile();
16   }//end  main
17  //-----------------------------------------//
18   public  hnwfile(){//constructor
19   playData();
20   }//end  constructor
21
22   private  void  playData()  {
23   try{
24     FileOutputStream  out  =  null;
25     out  =  new  FileOutputStream("HNW.AU");
26     out.write(46);  //  .
27     out.write(115);  //  S
28     out.write(110);  //  N
29     out.write(100);  //  D
30     //  Header  size
31     out.write(0);  //  0
32     out.write(0);  //  0
33     out.write(0);  //  0
34     out.write(24);  //  24
35     //  Data  size  -  default  ffff
36     out.write(255);  //  ff
37     out.write(255);  //  ff
38     out.write(255);  //  ff
39     out.write(255);  //  ff
40     //  sample  type  3  =  pcm
41     out.write(0);  //  0
42     out.write(0);  //  0
43     out.write(0);  //
44     out.write(3);  //       //  sample  rate  16000
45     out.write(0);  //  0
46     out.write(0);  //  0
47     out.write(62);  //  16000
48     out.write(128);  //       //  2  channels
49     out.write(0);  //  0
50     out.write(0);  //  0
51     out.write(0);  //
52     out.write(2);  //
53     //
54     //  need  to  generate  random  numbers
55     Random  generator  =  new  Random();
56     //  harshness the higher  this value the more rough/harsh
57     double  ss[]  =  new  double[harshness];
58     double  sa;
59     short  s;  //  signed  two  byte  integer    for  PCM  data
```

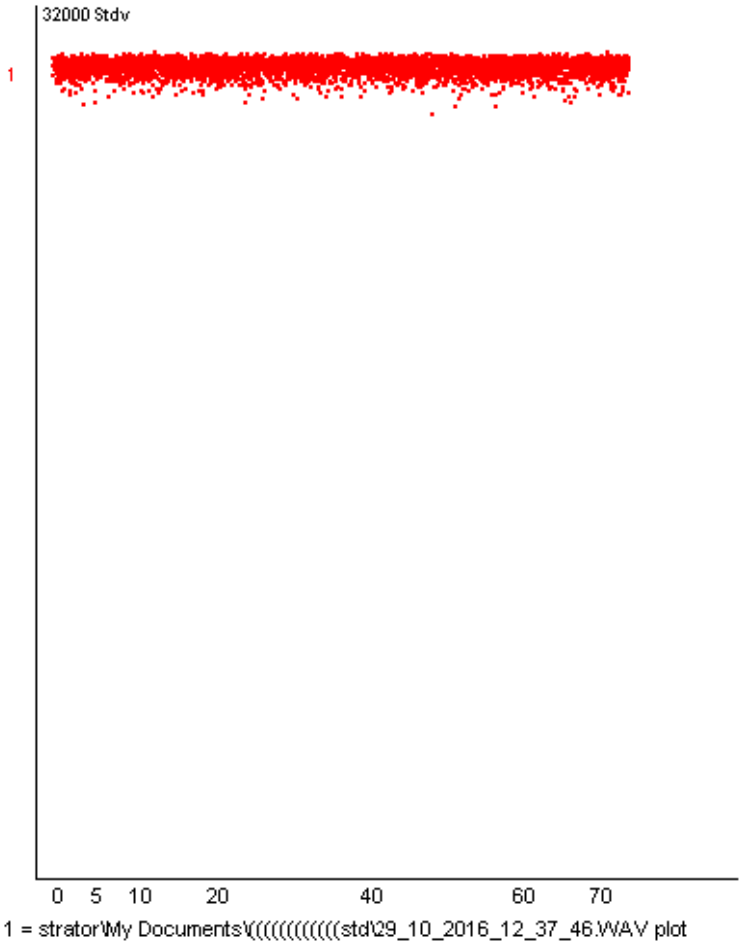
```
60     for(int  z  =  -1;  z  <  noiselen;  z++){
       //  make  this  big  for  long  pieces  or     61      //  if
noiselen  was  0  from  commandline  run  forever!
62      //  note  only  play  half  buffer  to  stop  gaps???
63      if  (noiselen  <  1)  z  =  -2;
64      //----------------------  loop  ------------------------
65      for(int  cnt  =  0;  cnt  <  1000;  cnt  =  cnt  +  2  ){
66       //  for  testing  so  i  can  see  the  data
67       //    for(int  cnt  =  0;  cnt  <  100;  cnt  =  cnt  +  2
){
68       int  r  =  generator.nextInt();
69       s  =  (short)  r;
70       //  store  new  sample  -  roll  out  others
71       for(int  x  =  0;  x  <  harshness-1;  x++){
72        ss[x]  =  ss[x+1];
73       }  //  end  of  x  for
74       ss[harshness-1]  =  (short)s;
75       //  average    the  last  harshness  number  of  samples
76       //  remove  the  higher  pitches
77       sa  =  0;
78       for(int  x  =  0;  x  <  harshness;  x++){
79        sa  =  sa  +  ss[x];
80      }
81       sa  =  sa  /  harshness;
82       s  =  (short)  sa;
83       //============  Process    for  new  sounds  ========
84      //        loudness====    !!!!
85       //  basically  no  wimpy  numbers  -  only  fat  ones
86       if  (s  >  0  &  s  <  30000)  {s  =  (short)  (s  +
30000);}
87       if  (s  <  0  &  s  >  -30000)  {s  =  (short)  (s  -
30000);}
88
89       //      System.out.println(s);
90       //===============  End  of  Process  ============
91       //  as  the  output  stream  is  a  byte  array  load
         // the  low/high  integer
92             byte  hexBase  ;  //  A  byte  of  all  ones
93       hexBase  =  (byte)  255;
94       byte  b1  =  (byte)  (hexBase  &  s);
95       byte  b2  =  (byte)  ((hexBase  <<(8)&  s)  >>  8);
96      out.write(b2);
97      out.write(b1);
98      }  //  end  of  process  this  200  bytes
99     }  //  end  of  for  loop  =  length  of  piece
100    out.close();
101    }catch  (Exception  e)  {
102    e.printStackTrace();
103    System.exit(0);
104    }//end  catch
105
106   }//end  playData
107
108  }//end  outer  class
```

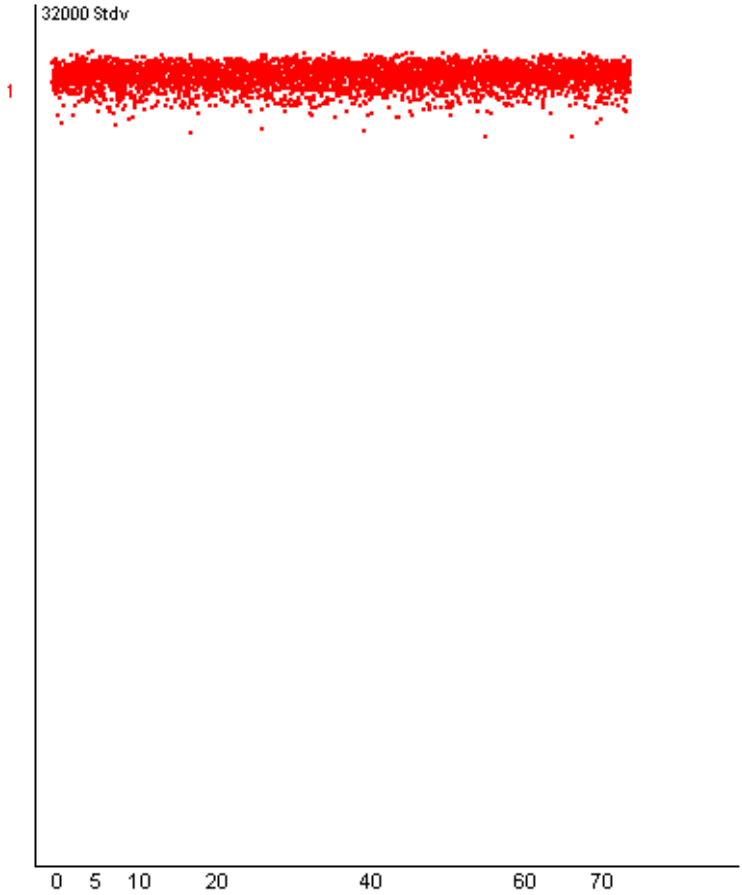
```
1  //MAKES  HNW  EACH  RUN  USES  RANDOM  KEY  -
   //THIS  WAS  USED  IN  THE  FOLLOWING 3 EXAMPLES
2  import  javax.sound.sampled.*;
3  import  java.util.*;
4  public  class  HNW90{
5   //The  following  are  general  instance  variables
6   //  used  to  create  a  SourceDataLine  object.
7   AudioFormat  audioFormat;
8   AudioInputStream  audioInputStream;
9   SourceDataLine  sourceDataLine;
10   //The  following  are  audio  format  parameters.
11   float  sampleRate  =  16000.0F;
12   //Allowable  8000,11025,16000,22050,44100
13   int  sampleSizeInBits  =  16;
14   //Allowable  8,16
15   int  channels  =  1;
16   //  MONO!
17   //Allowable  1,2
18   boolean  signed  =  true;
19   //Allowable  true,false
20   boolean  bigEndian  =  true;
21   //Allowable  true,false
22  //
23   //  size  of  number  of  samples  to  smooth
24   //  lager  =  more  coarse  sounds
25   static  int  harshness  =  200;
26   static  int  noiselen  =  0;    //  length  of  piece
27  //-----------------------------------------//
28   public  static  void  main(String  args[]){
29    Random  generator  =  new  Random();
30    harshness    =  generator.nextInt(200);
31    new  HNW90();
32   }//end  main
33  //-----------------------------------------//
34   public  HNW90(){//constructor
35   playData();
36   }//end  constructor
37
38   private  void  playData()  {
39   try{
40     //Get  the  required  audio  format
41     audioFormat  =  new  AudioFormat(
42      sampleRate,
43      sampleSizeInBits,
44      channels,
45      signed,
46      bigEndian);
47     //Get  info  on  the  required  data  line
48     DataLine.Info  dataLineInfo  =
49      new  DataLine.Info(
50       SourceDataLine.class,
51       audioFormat);
52     //Get  a  SourceDataLine  object
53     sourceDataLine =
(SourceDataLine)AudioSystem.getLine(dataLineInfo);
54     new  ListenThread().start();
55    }catch  (Exception  e)  {
56    e.printStackTrace();
57    System.exit(0);
```

```
58    }//end  catch
59   }//end  playData
60  //==============================================//       61
class  ListenThread  extends  Thread{
62    byte  playBuffer[]  =  new  byte[200];
63    //  the  size  of  this  buffer  =  length  of  sound
64    public  void  run(){
65     //  need  to  generate  random  numbers
66     Random  generator  =  new  Random();
67     //  harshness  the  higher  this  value the more rough/harsh
68     double  ss[]  =  new  double[harshness];
69     double  sa;
70     short  s;  //  signed  two  byte  integer    for  PCM  data
71    try{
72     sourceDataLine.open(audioFormat);
73     sourceDataLine.start();
74      for(int  z  =  -1;  z  <  noiselen;  z++){
        //  make  this  big  for  long  pieces  or
75       //  if  noiselen  was  0  from  commandline  run  forever!
76       //  note  only  play  half  buffer  to  stop  gaps???
77       if  (noiselen  <  1)  z  =  -2;
78       //----------------------  loop  ------------------------
79       for(int  cnt  =  0;  cnt  <  100;  cnt  =  cnt  +  2  ){
80        //  for  testing  so  i  can  see  the  data
81        //    for(int  cnt  =  0;  cnt  <  100;  cnt  =  cnt  +  2
){
82        int  r  =  generator.nextInt();
83        s  =  (short)  r;
84        //  store  new  sample  -  roll  out  others
85        For(int  x  =  0;  x  <  harshness-1;  x++){
86         ss[x]  =  ss[x+1];
87        }  //  end  of  x  for
88        ss[harshness-1]  =  (short)s;
89        //  average    the  last  harshness  number  of  samples
90        //  remove  the  higher  pitches
91        sa  =  0;
92        for(int  x  =  0;  x  <  harshness;  x++){
93         sa  =  sa  +  ss[x];
94       }
95        sa  =  sa  /  harshness;
96        s  =  (short)  sa;
97        //============  Process    for  new  sounds  ========
98       //        loudness====    !!!!
99        //  basically  no  wimpy  numbers  -  only  fat  ones
100        if  (s  >  0  &  s  <  30000)  {s  =  (short)  (s  +
30000);}
101        if  (s  <  0  &  s  >  -30000)  {s  =  (short)  (s  -
30000);}
102
103        //      System.out.println(s);
104        //===============  End  of  Process  ============
105       //asheutput stream is a  byte array load the low/high
integer
106        byte  hexBase  ;
107        //  A  byte  of  all  ones
108        hexBase  =  (byte)  255;
109        byte  b1  =  (byte)  (hexBase  &  s);
110        byte  b2  =  (byte)  ((hexBase  <<(8)&  s)  >>  8);
111       //System.out.println(b1);
112        playBuffer[cnt]  =  b2  ;
```

```
113        playBuffer[cnt+1]  =  b1  ;
114      }
115       sourceDataLine.write(playBuffer,  0,  100);
116      }  //  end  of  loop
117      //  -------------------------  loop  --------------------
118      //Block  and  wait  for  internal  buffer  of  the
119      //  SourceDataLine  to  become  empty.
120     sourceDataLine.drain();  121      //Finish  with  the
SourceDataLine
122     sourceDataLine.stop();
123     sourceDataLine.close();
124
125     }catch  (Exception  e)  {
126     e.printStackTrace();
127     System.exit(0);
128     }//end  catch
129    }//end  run
130   }//end  inner  class  ListenThread
131  }//end  outer  class
```


32000 Stdv
1
0
5
10
20
40
60
70
1 – tor\My Documents\((((((((((((stc\ InW_29_10_2010_17_44_07.wav plot


32000 Stdv
1
0
5
10
20
40
60
70
1 = strator\My Documents\((((((((((((std\29_10_2016_12_37_46.WAV plot



1 = strator\My Documents\((((((((((((std\29_10_2016_14_39_01.WAV plot