Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

Ψηφιακή Επεξεργασία Σήματος

Εξάμηνο 6ο (Εαρινό Εξάμηνο 2020-2021)

3ο Εργαστηριακό Project Σπανός Νικόλαος – el18822

Η εργασία θα εξηγηθεί και θα παρουσιαστεί για το πρώτο πλαίσιο του σήματος. Στο τέλος θα παρουσιαστούν αποτελέσματα από όλη την διαδικασία , όπως τα plots των σημάτων και τα MSE μεταξύ του αρχικού σήματος.

Μέρος 1ο : Ψυχοακουστικό Μοντέλο

1.0 Αρχικά ,κανονικοποιούμε το σήμα με χρήση της np.max() στο απόλυτο του ολικού σήματος μουσικής και παίρνουμε τα πρώτα 512 δείγματα.

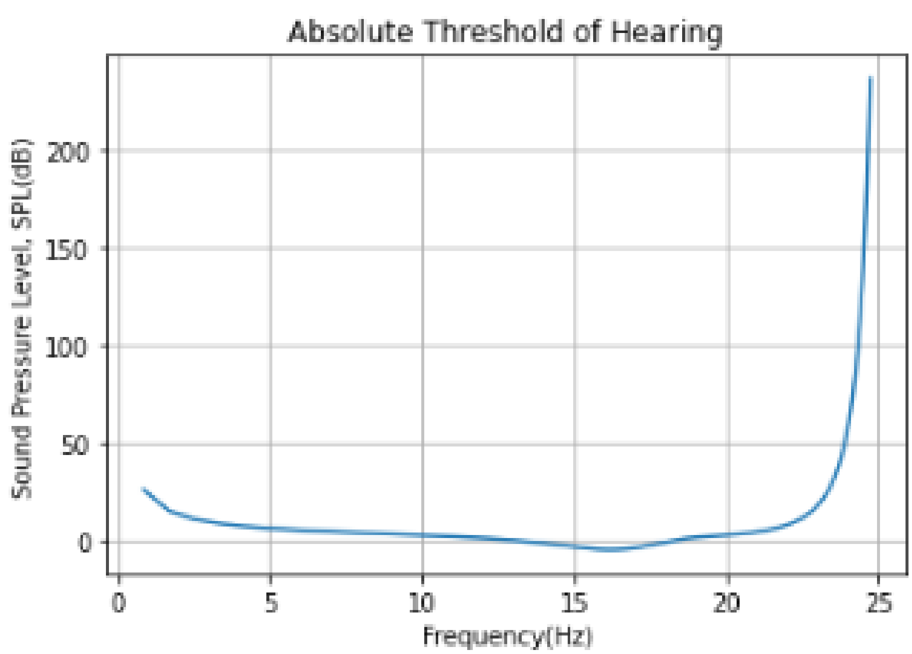
1.1 Για ορισμό του παραθύρου Hanning , χρησιμοποιούμε την np.hanning(N) της numpy.

Ορίζουμε τις συχνότητες Bark και το κατώφλι ακοής σύμφωνα με τις δοσμένες συναρτήσεις.

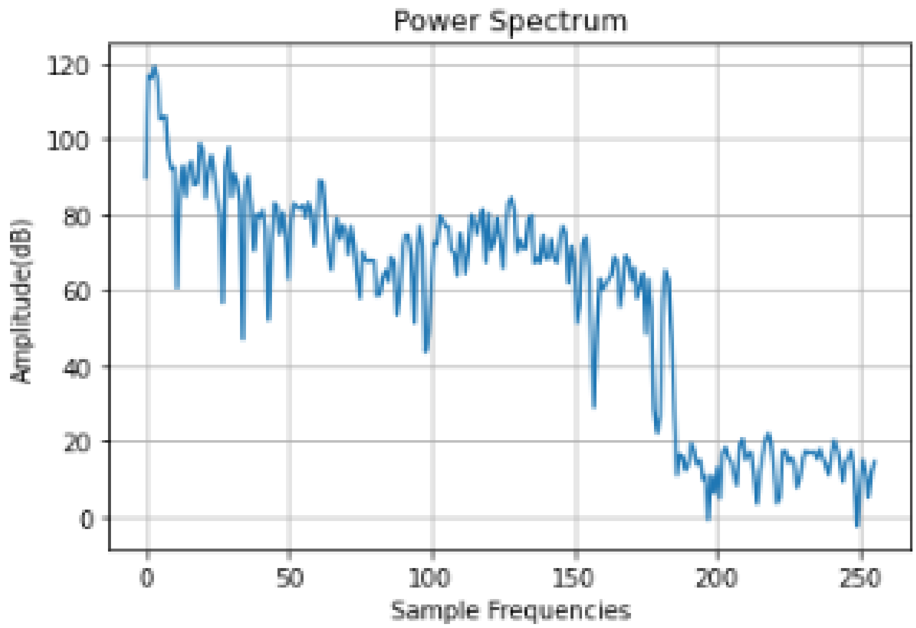
$$T_q(f) = 3.64(f/1000)^{-0.8} - 6.5e^{-0.6(f/1000-3.3)^2} + 10^{-3}(f/1000)^4 \text{ (dB SPL)}.$$

$$b(f) = 13\arctan(.00076f) + 3.5\arctan[(f/7500)^2] \text{ (Bark)},$$

Παρατίθεται η γραφική παράσταση του κατωφλίου προς τις συναρτήσεις Bark:



Ορίζουμε το φάσμα ισχύος σύμφωνα με τον τύπο ,χρησιμοποιώντας το παράθυρο Hanning που δημιουργήσαμε προηγουμένως και το πρώτο πλαίσιο ανάλυσης.Παρατίθεται η γραφική παράσταση προς τα δείγματα:



1.2 Για να εντοπίσουμε τις μάσκες , ελέγχουμε στα κατάλληλα διαστήματα ανάλογα με το δείγμα. Για κάθε φορά, ελέγχουμε αν η συνθήκη ισχύει για τα Δκ που ορίζονται.

$$P(k) = PN + 10\log_{10}\left|\sum_{n=0}^{N-1} w(n)x(n)e^{-j\frac{2\pi kn}{N}}\right|^2, 0 \le k \le \frac{N}{2}.$$

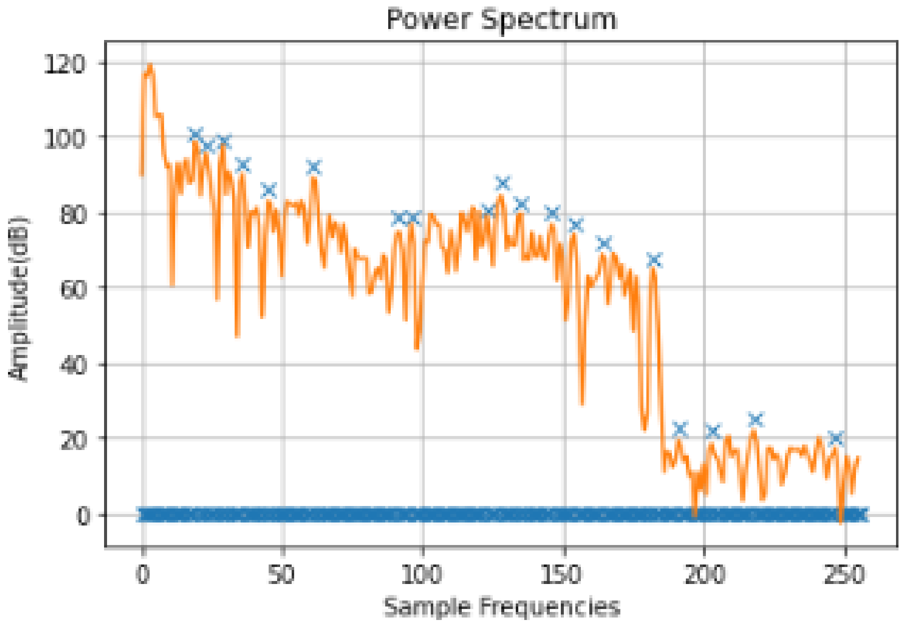
$$S_T(k) = \begin{cases} 0, & \text{αν } k \notin [3, 250] \\ P(k) > P(k \pm 1) \bigwedge P(k) > P(k \pm \Delta_k) + 7\text{dB}, & \text{αν } k \in [3, 250] \end{cases}$$

όπου το $\bigwedge$ ισοδυναμεί με boolean and και το Δ_k:

$$\Delta_k \in \begin{cases} 2, & 2 < k < 63 & (0.17 - 5.5\text{kHz}) \\ [2, 3] & 63 \le k < 127 & (5.5 - 11\text{kHz}) \\ [2, 6] & 127 \le k \le 250 & (11 - 20\text{kHz}) \end{cases}$$

$$P_{TM}(k) = \begin{cases} 10\log_{10}(10^{0.1(P(k-1))} + 10^{0.1(P(k))} + 10^{0.1(P(k+1))})(\text{dB}), & \text{αν } S_T(k) = 1 \\ 0, & \text{αν } S_T(k) = 0 \end{cases}$$

Αν ισχύει η σχέση για κάποιο Δκ, θεωρούμε πως βρίσκουμε μάσκα.Ύστερα, όπου η St(k) είναι διάφορη του μηδενός, υπολογίζουμε την ισχύ της μάσκας:



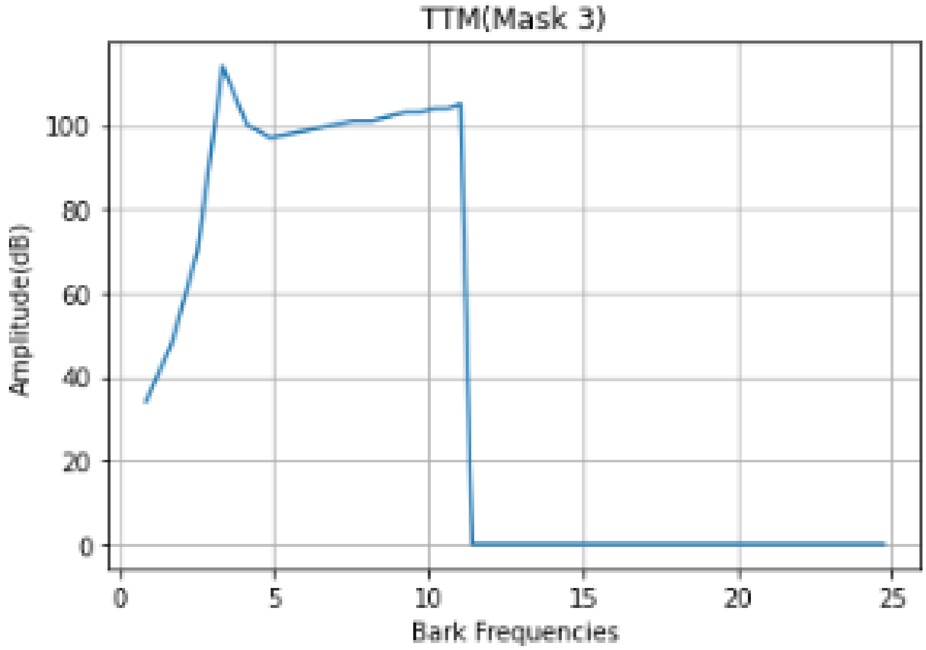
Τα ‘Χ’ που είναι διάφορα του μηδενός αντιπροσωπεύουν τις μάσκες που έχουν βρεθεί στο φάσμα.

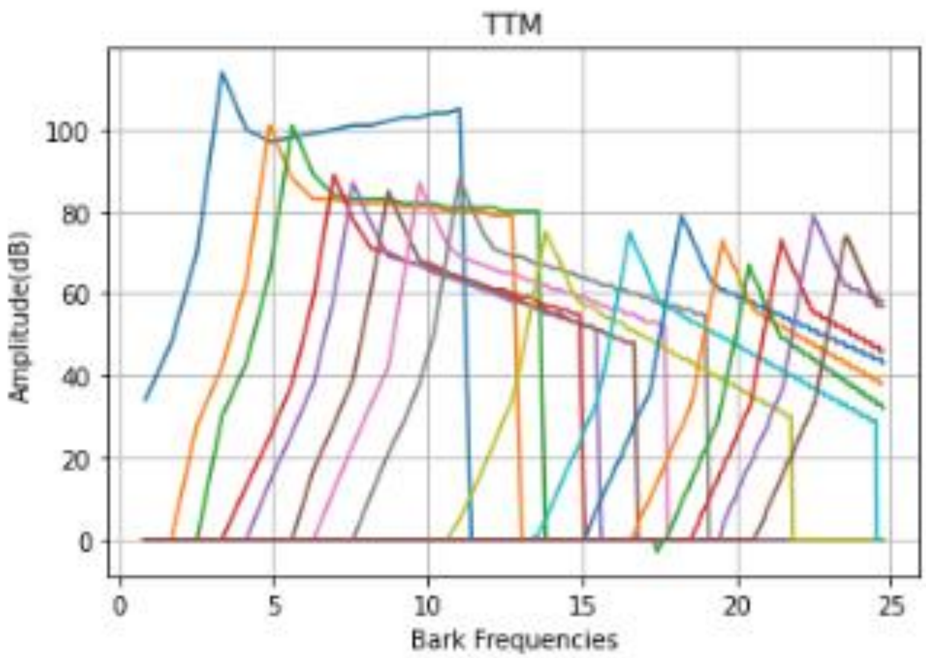
1.3 Σε αυτό το βήμα φορτώνουμε στον κώδικα μας τους πίνακες που μας δώθηκαν στο υλικό της άσκησης. Αφού τους φορτώσουμε, μηδενίζουμε όλες τις τιμές για κ<3 και κ>250 και προχωρούμε στο επόμενο βήμα

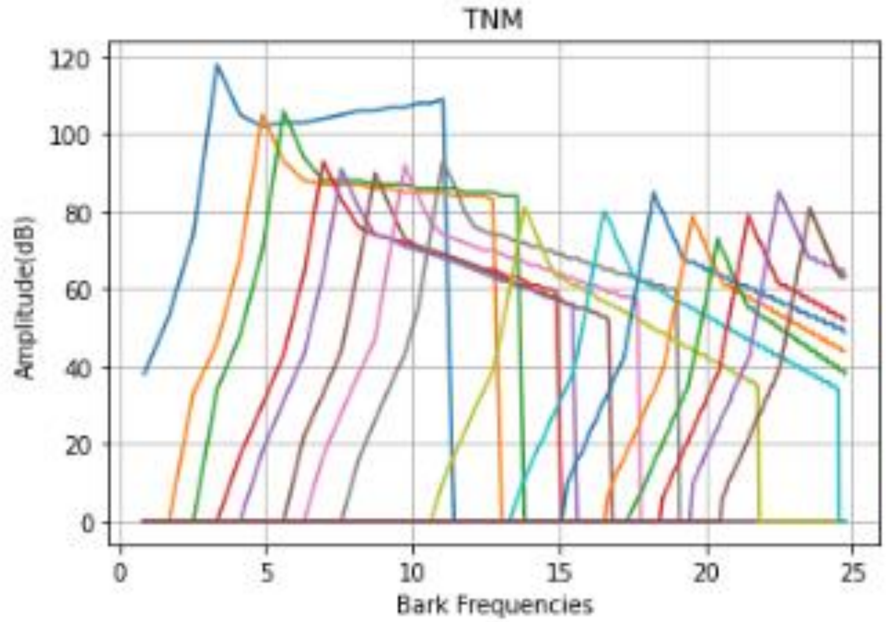
1.4 Ορίζουμε μια συνάρτηση SF που παίρνει όρισμα το Δb , το σημείο j που βρίσκεται η μάσκα και τον πίνακα PTM ή PNM, για να το χρησιμοποιήσουμε ανάλογα στον προσδιορισμό των δύο κατωφλίων κάλυψης.

$$SF(i,j) = \begin{cases} 17\Delta_b - 0.4P_{TM}(j) + 11, & -3 \le \Delta_b < -1 \\ (0.4P_{TM}(j) + 6)\Delta_b, & -1 \le \Delta_b < 0 \\ -17\Delta_b, & 0 \le \Delta_b < 1 \\ (0.15P_{TM}(j) - 17)\Delta_b - 0.15P_{TM}(j), & 1 \le \Delta_b < 8 \end{cases}$$

Για κάθε σημείο που βρίσκουμε μάσκα ορίζουμε έναν πίνακα 256 μηδενικών. Στο διάστημα 12-Bark τοποθετούμε τις ανάλογες τιμές στα ανάλογα i. Οι υπόλοιπες τιμές μένουν μηδενικές. Ο πίνακας τοποθετείται σε ένα λεξικό με κατάλληλο αναγνωριστικό. Για τα σημεία που δεν έχουν μάσκες, δεν ορίζουμε πίνακες , καθώς δεν τα χρησιμοποιούμε. Για παράδειγμα, παίρνουμε αυτό το αποτέλεσμα για την μάσκα στην θέση 3 και για ολές τις μάσκες στο φάσμα:

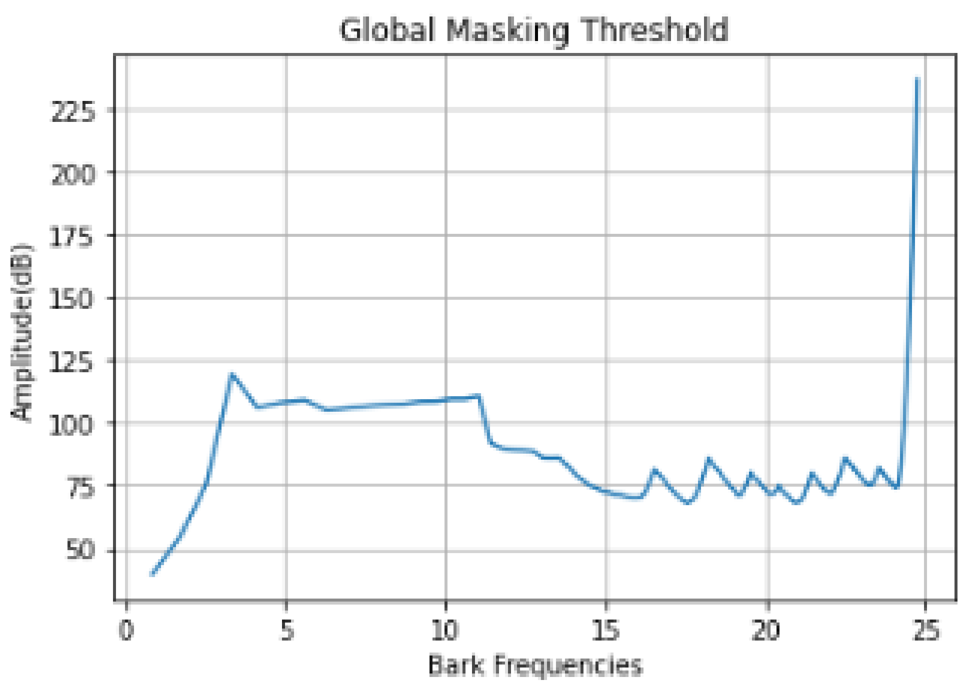

TTM(Mask 3)
Amplitude(dB)
Bark Frequencies
0
20
40
60
80
100
0
5
10
15
20
25


TTM
Amplitude(dB)
Bark Frequencies
0
20
40
60
80
100
0
5
10
15
20
25



1.5 Στην συνέχεια, αθροίζουμε βάσει του τύπου και υπολογίζουμε το συνολικό κατώφλι:

$$T_g(i) = 10\log_{10}\left(10^{0.1T_q(i)} + \sum_{l=1}^{L} 10^{0.1T_{TM}(i,\ell)} + \sum_{m=1}^{M} 10^{0.1T_{NM}(i,m)}\right)$$



Μέρος 2ο: Χρονο-Συχνοτική Ανάλυση με Συστοιχία Ζωνοπερατών Φίλτρων

2.0 Ορίζουμε τα φίλτρα ανάλυσης και σύνθεσης σύμφωνα με τους τύπους:

$$h_k(n) = \sin\left[\left(n+\frac{1}{2}\right)\frac{\pi}{2M}\right]\sqrt{\frac{2}{M}}\cos\left[\frac{(2n+M+1)(2k+1)\pi}{4M}\right]$$

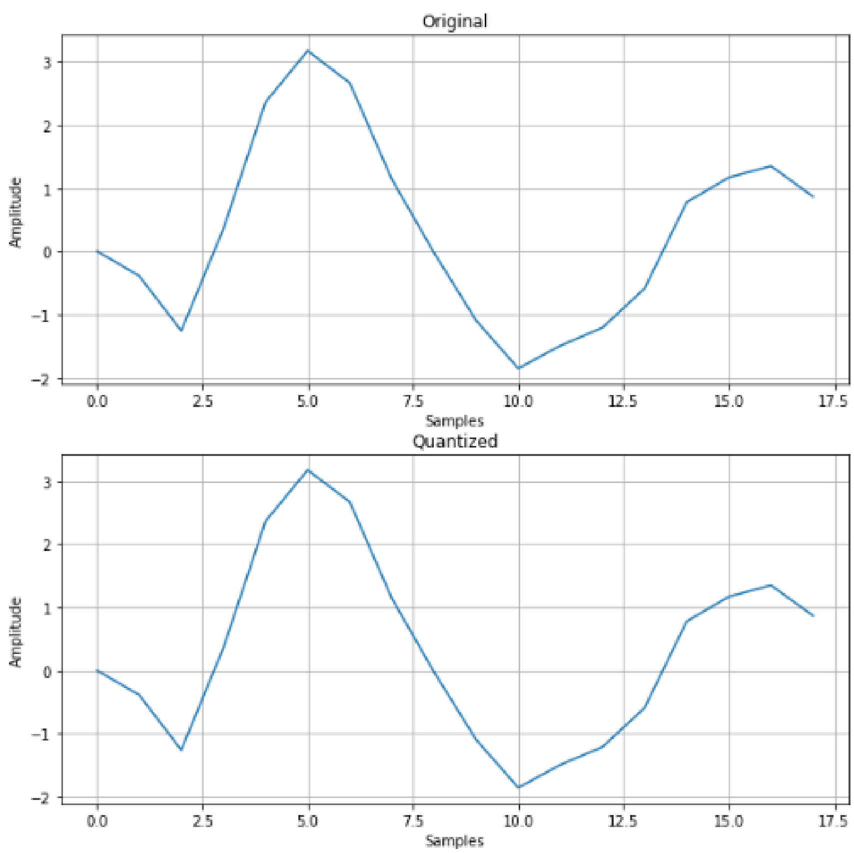
$$g_k(n) = h_k(2M-1-n)$$

και τα διατηρούμε σε δύο λεξικά , μέχρι που θα τα χρησιμοποιήσουμε στα επόμενα βήματα.

2.1 Εκτελούμε συνέλιξη του πλαισίου με τα φίλτρα ανάλυσης και κάνουμε αποδεκατισμό με παράγοντα 32, χρησιμοποιώντας μια αυτοσχέδια συνάρτηση.

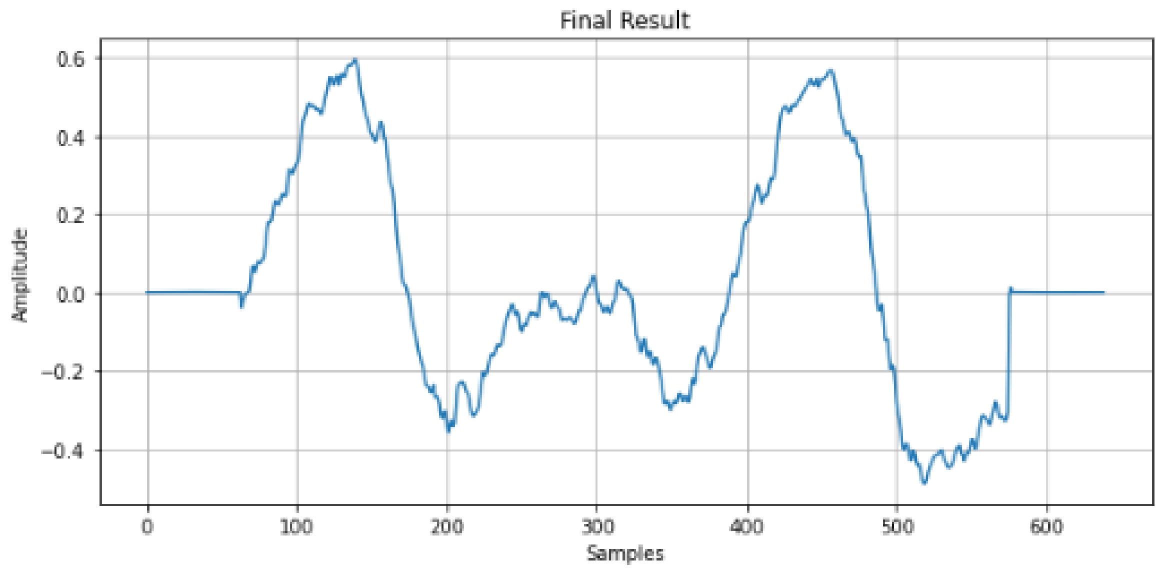
2.2 Για να βρούμε το κατάλληλο διάστημα, ορίζουμε μια for loop για να βρούμε τα i που ορίζουν το διάστημα. Την πρώτη φορά που θα ικανοποιηθεί η συνθήκη, διατηρούμε τον δείκτη σαν αρχή και ανα κάθε loop διατηρούμε ως τέλος τον τρέχον δείκτη. Έτσι , όταν δεν ισχύει η συνθήκη, θα έχουμε και το τέλος του διαστήματος. Προσδιορίζουμε τα bits κωδικοποίησης για κάθε φίλτρο , από τα οποία πάιρνουμε το άνω όριο, καθώς αυτό είναι απαραίτητο για σωστή κωδικοποίηση.

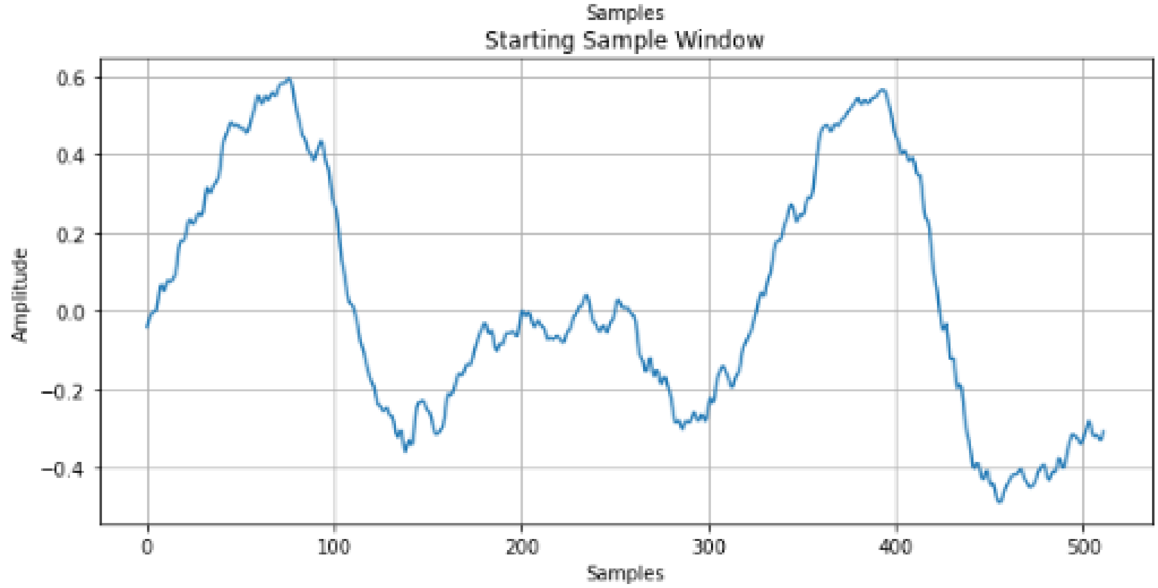
Χρησιμοποιούμε τον κβαντιστή μας για να κβαντίσουμε το σήμα:



Η διαφορά δεν είναι πού αισθητή , καθώς έχουμε μόνο 18 τιμές , όμως το σήμα έχει κβαντιστεί κατά τα ανάλογα bits. Αυτό αποτελεί το plot του πρώτου φίλτρου.

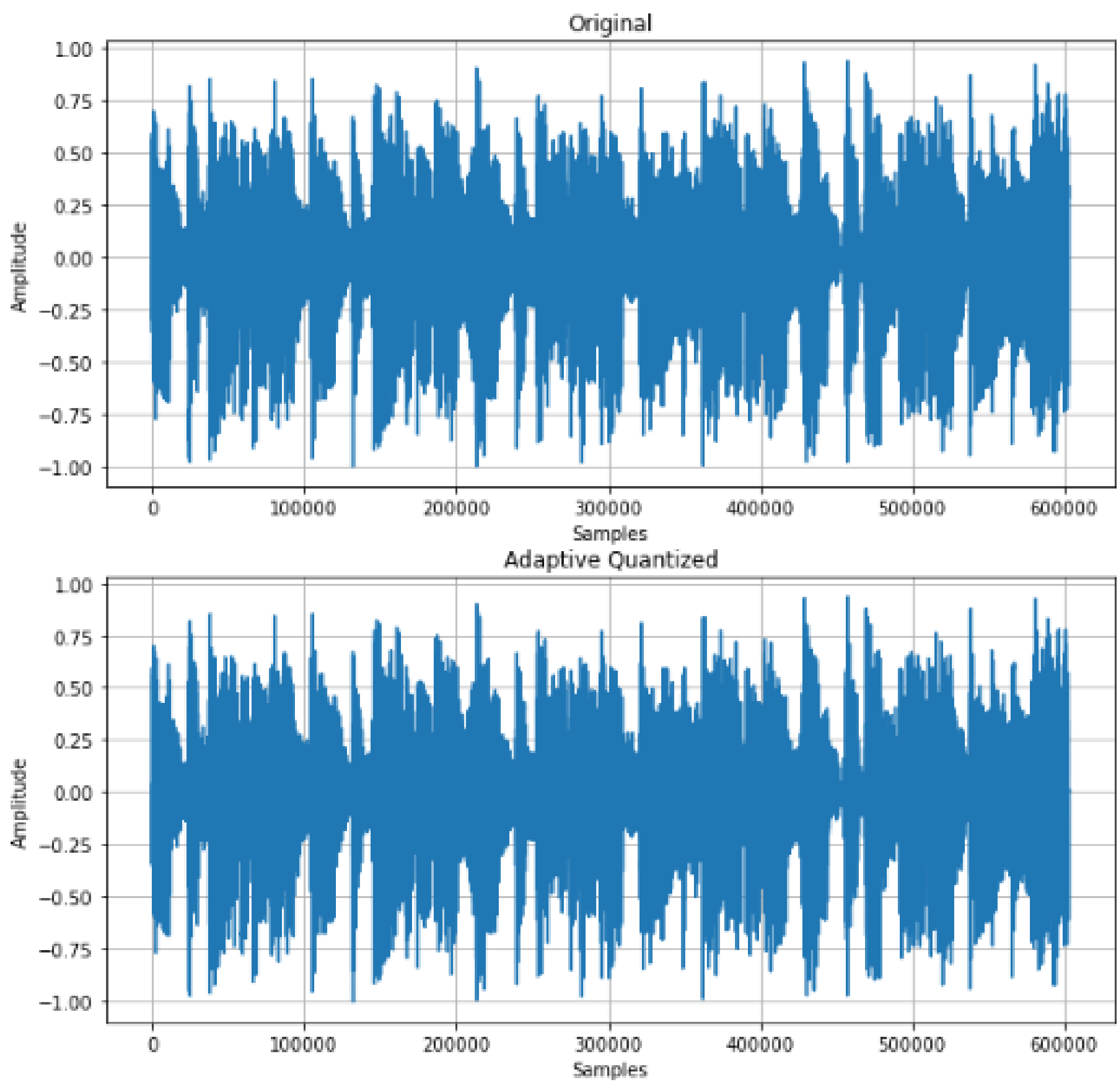
2.3 Ύστερα, εκτελούμε υπερδειγματοληψία με παράγοντα 32 , πηγαίνοντας συνολικά στα 575 σημεία από τα 512. Εκτελούμε συνέλιξη με τα φίλτρα σύνθεσης που υπολογίσαμε στο Βήμα **2.0** και φτάνουμε συνολικά στα 639 σημεία. Το τελικό αποτέλεσμα σε σύγκριση με το αρχικό είναι το εξής:

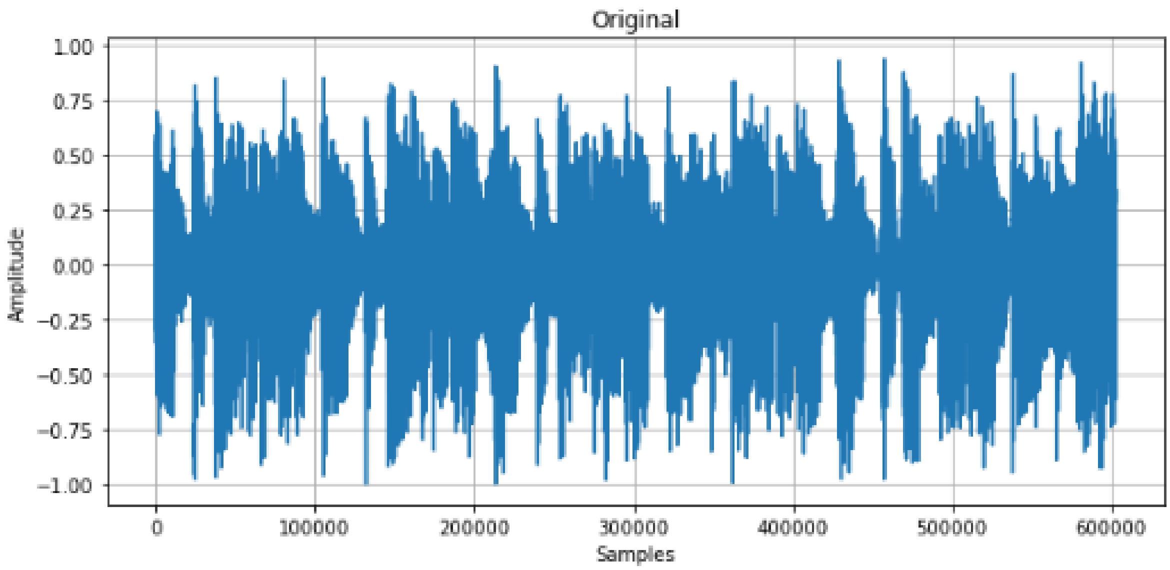


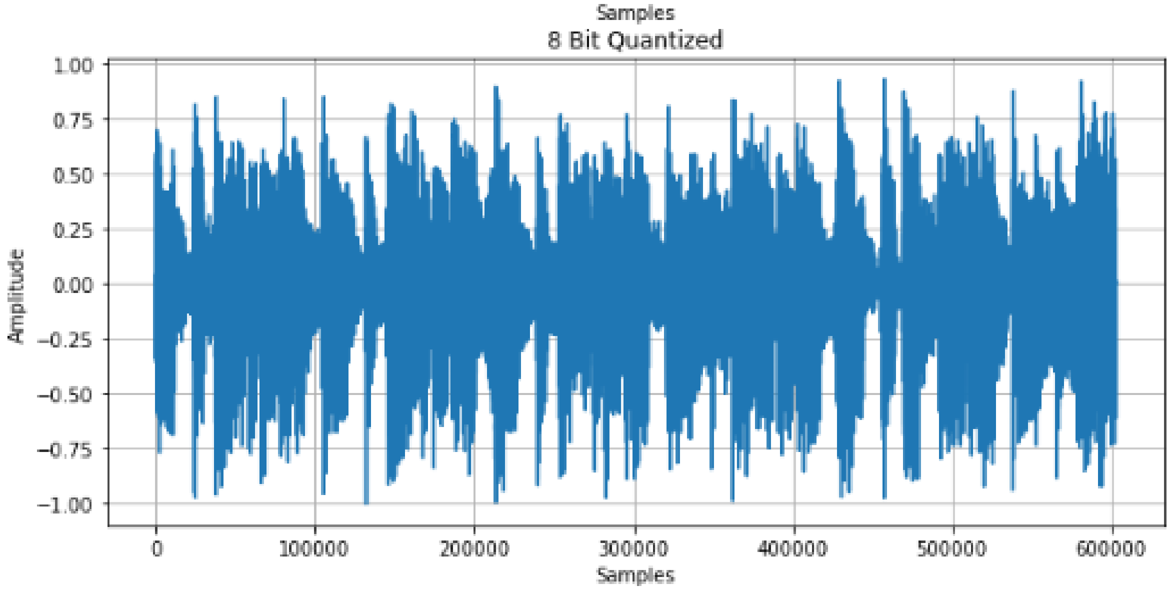


Το συνθετικό σήμα είναι αρκετά όμοιο με το αρχικό. Το συνθετικό , διαισθητικά και με δοκιμές , βρέθηκε ότι είναι μετατοπισμένο κατά 64 δείγματα σε σχέση με το αρχικό.Χρησιμοποιούμε μία συνάρτηση για να δούμε πόσες ξεχωριστές τιμές βρίσκονται στο κάθε σήμα. Το σήμα μουσικής έχει 610 μοναδικές τιμές και το συνθετικό 511. Άρα, όπως φαίνεται , το σήμα είναι συμπιεσμένο.

Η διαδικασία αυτή εκτελείται για κάθε πλαίσιο. Το τελευταίο πλαίσιο αποφάσισε να αγνοηθέι για ευκολία στην υλοποίηση, καθώς δεν διαθέτει 512 σημεία. Τα συνολικά αποτελέσματα αθροίστηκαν σε με την μέθοδο OverlapAdd όπου τα τελευταία 127 του πλαισίου κ-1 αθροίζονται στα πρώτα 127 σημεία στο πλαίσιο κ. Παρακάτων παρατίθενται οι γραφικές παραστάσεις των δύο τελικών αποτελεσμάτων για το adaptive και για την κωδικοποίηση με 8 bit:



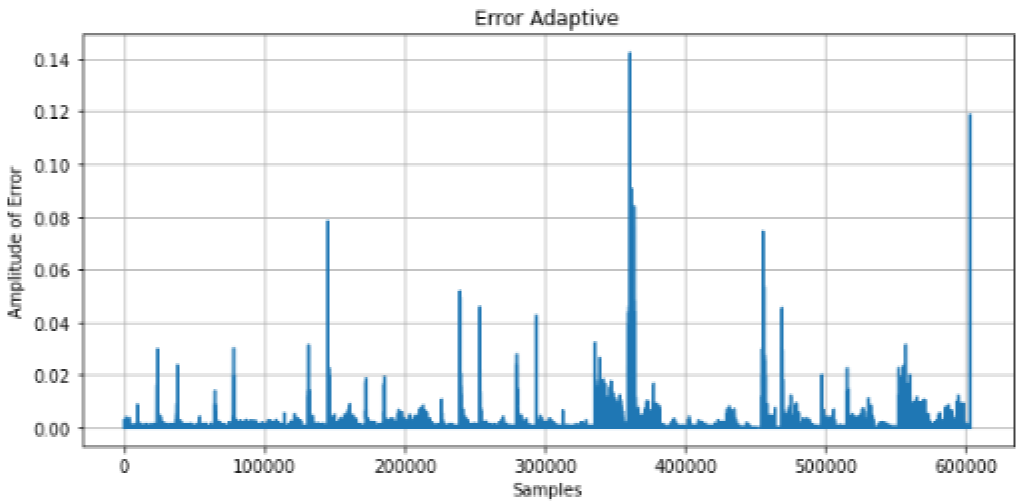


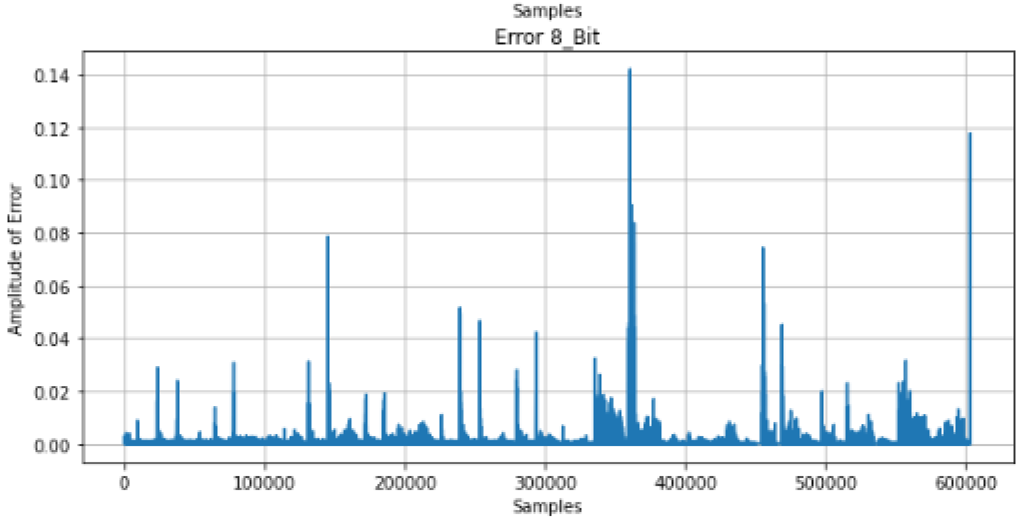


Για να υπολογίσουμε το MSE μετατοπίζουμε τα σήματα κατά 64 δείγματα αριστερά. Παίρνουμε τα εξής αποτελέσματα:

- MSE_adaptive_error= 0.00035487550250893284

- MSE_8_bit_error= 0.0003557365425929375





Παρατηρούμε ότι ο adaptive κβαντιστής έχει ελάχιστα λιγότερο error, όμως η διαφορά στα σήματα είναι αισθητή. Ο adaptive παράγει ένα σήμα που είναι σχεδόν ολόιδιο με το αρχικό σήμα, ενώ η κβάντιστη με 8 bit παράγει έναν θόρυβο πίσω από το σήμα. Είναι σαφές πως η μέθοδος με τον adaptive κβαντιστή είναι λίγο πιο περίπλοκη , αλλά παράγει καλύτερα αποτελέσματα ,όσον αφορά την ποιότητα του σήματος. Και στα δύο τελικά σήματα τα αρχεία είναι 1,179 MB που μας οδηγεί τελικά σε συμπίεση περίπου της τάξης του 50%, μία σημαντική διαφορά για αρχεία μεγάλου μεγέθους.