

Problem 4 : Signal Separation using Non Negative Matrix Factorization

i. Learning Bases

NMF code

```
function [B, W, obj, k] = nmf(M,B_init,W_init,n_iter)
[D,N] = size(M);
B = B_init;
W = W_init;
k = 0;
obj= compute_objective(M, B, W);
%iter
for k = 1:n_iter
    B = B.*((M./(B*W))*W')./(ones(1,N)*W');
    W = W.*((B'*(M./(B*W)))./(B'*ones(D,1)));
    obj= compute_objective(M, B, W);
end
end
```

Plot:

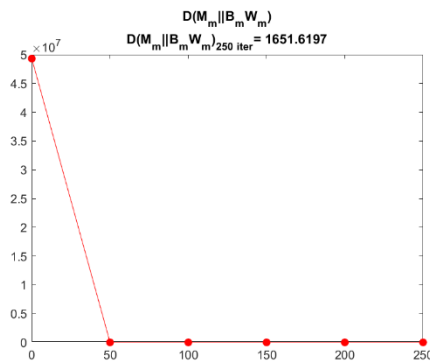


Figure 1 KL divergence of music

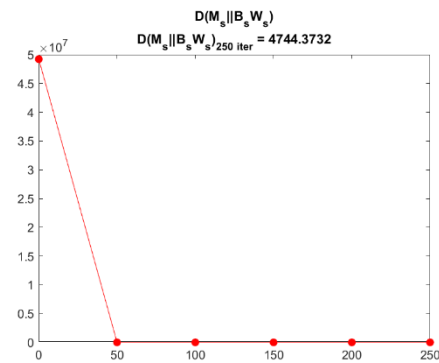


Figure 2 KL divergence of speech

Note: Since the value at zero is so big that the graph automatically change its scale to fit, the decreasing trend of last five value cannot be directly seen, and the actual value shown in Table 1.

The value of $D(M_m || B_m W_m)$ for $n_iter = 250$ is 1651.6197

The value of $D(M_s || B_s W_s)$ for $n_iter = 250$ is 4744.3732

Iteration	0	50	100	150	200	250
$D(M_m B_m W_m)$	5.018E+07	3.215E+03	2.225E+03	1.906E+03	1.749E+03	1.652E+03
$D(M_s B_s W_s)$	5.064E+07	7.831E+03	5.816E+03	5.231E+03	4.929E+03	4.744E+03

Table 1 KL divergence for music and speech

ii. Signal Separation**Code**

```
function [ M_speech_rec, M_music_rec ] = B_nmf(M,Bs,Bm,n_iter)

load('data\Wm_init.csv');
load('data\Ws_init.csv');

[D,N] = size(M);
B = [Bs,Bm];
W = [Ws_init;Wm_init];
%iter
for k = 1:n_iter
    W = W.*((B'*(M./(B*W)))./(B'*ones(D,1)));
end
M_speech_rec = Bs*W(1:200,:);
M_music_rec = Bm*W(201:400,:);
end

%% signal seperation
[mixed, fs] = audioread('data\mixed.wav');
spectrogram = stft(mixed',2048,256,0,hann(2048));
Mmixed = abs(spectrogram);
iter = 250;
[ M_speech_rec, M_music_rec ] = B_nmf(Mmixed,Bs,Bm,iter);
csvwrite(['results\M speech
rec_',num2str(iter),'.csv'],M_speech_rec);
csvwrite(['results\M music
rec_',num2str(iter),'.csv'],M_music_rec);
iter = 500;
[ M_speech_rec, M_music_rec ] = B_nmf(Mmixed,Bs,Bm,n_iter);
csvwrite(['results\M speech
rec_',num2str(iter),'.csv'],M_speech_rec);
csvwrite(['results\M music
rec_',num2str(iter),'.csv'],M_music_rec);
```