



دانشگاه صنعتی شریف

دانشکده مهندسی برق

مسعود ناطقی ۹۶۱۰۲۵۶۷

تمرین درس تجزیه‌های تانسوری

دکتر حاجی‌پور

```

function [B, C] = mynnmf(A, j, algorithm, B0, C0, MaxItter, Tol)
n = size(A, 1);
m = size(A, 2);
if nargin == 2
    algorithm = "als";
    B0 = rand(n, j);
    C0 = rand(j, m);
elseif nargin == 5
    MaxItter = 100;
    Tol = 1e-4;
elseif nargin == 6
    Tol = 1e-4;
end
epsilon = 1e-16;
Itter = 1;
E = ones(size(A));
B = B0;
C = C0;
while Itter <= MaxItter && norm(E, "fro") >= Tol
    if algorithm == "als"
        C = max(epsilon, pinv(B' * B) * B' * A);
        B = max(epsilon, A * C' * pinv(C * C'));
    elseif algorithm == "mult"
        C = C .* (B' * A) ./ (B' * B * C + epsilon);
        B = B .* (A * C') ./ (B * C * C' + epsilon); %#ok<MHERM>
    end
    E = A - B * C;
    Itter = Itter + 1;
end

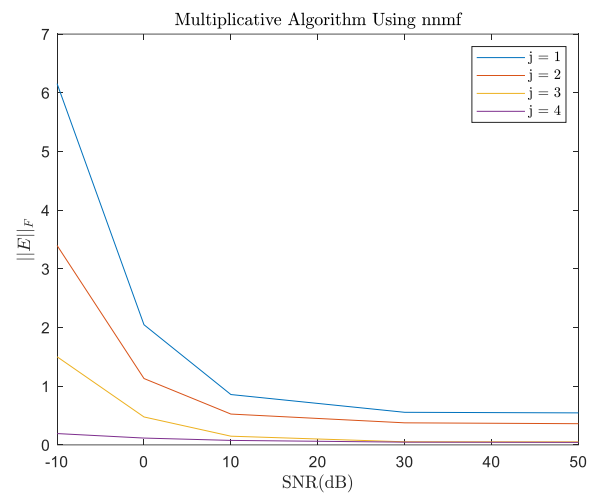
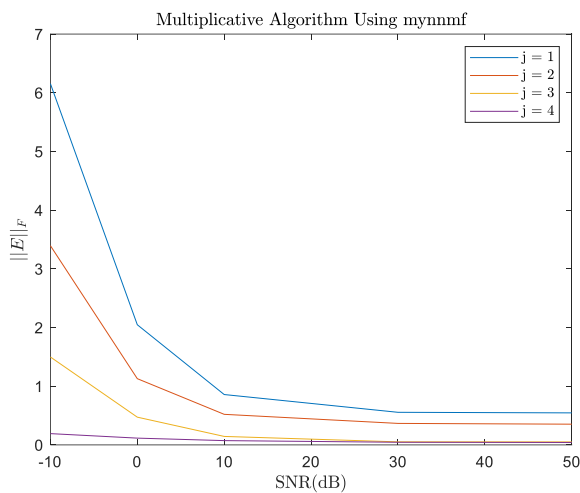
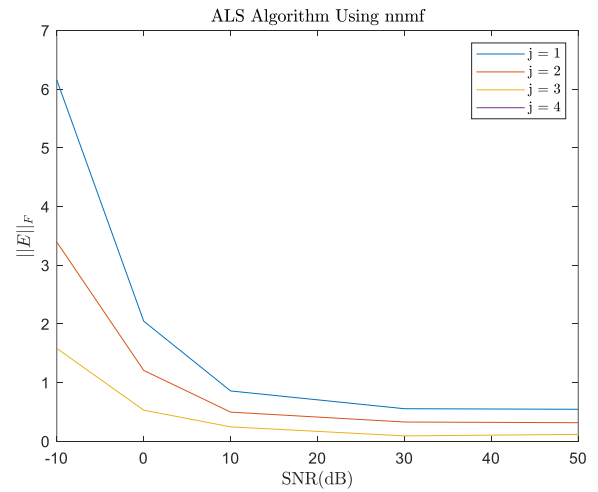
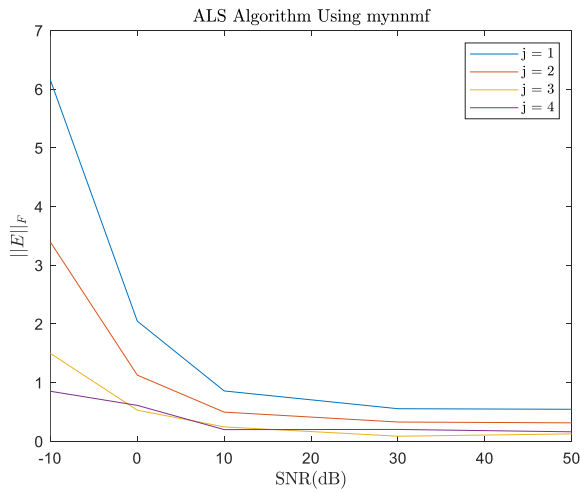
```

در شبیه‌سازی الگوریتم‌ها از شرط همگرایی ماکزیمم تکرار الگوریتم و همچنین یک حد پایین برای خطا استفاده کرده‌ایم. در ادامه برای بررسی کارکرد درست الگوریتم از یک مثال استفاده می‌کنیم و همزمان الگوریتم‌های ALS و Multiplicative را که توسط ما و MATLAB پیاده‌سازی شده‌اند، مقایسه می‌کنیم.

در قسمت اول کد main.m، ماتریس $A \in R_+^{6 \times 4}$ با استفاده از ۳ منبع ساخته شده‌است. در ادامه به ازای تعداد منابع مختلف ماتریس‌های B و C را به گونه‌ای می‌یابیم که $A = BC + E$ که در این رابطه سعی می‌کنیم تا $\|E\|_F$ را مینیمم کنیم. با توجه به حجم زیاد محاسبات، نتایج را برای $j = 3$ نشان می‌دهیم. مابقی نتایج در کد قابل مشاهده است.

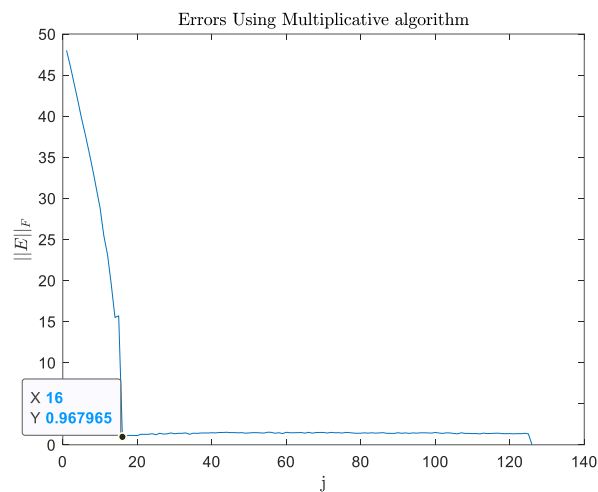
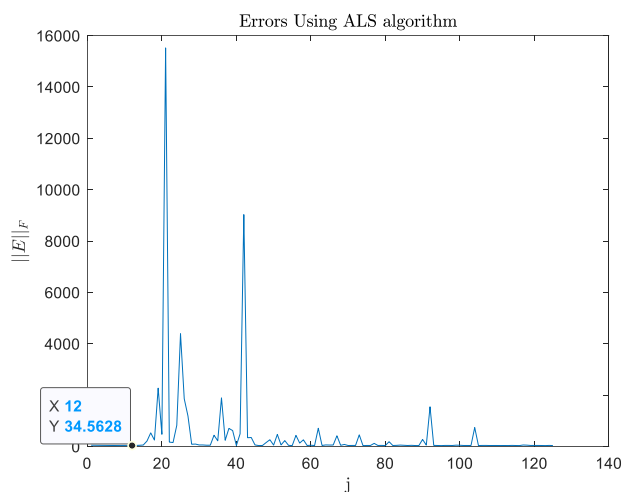
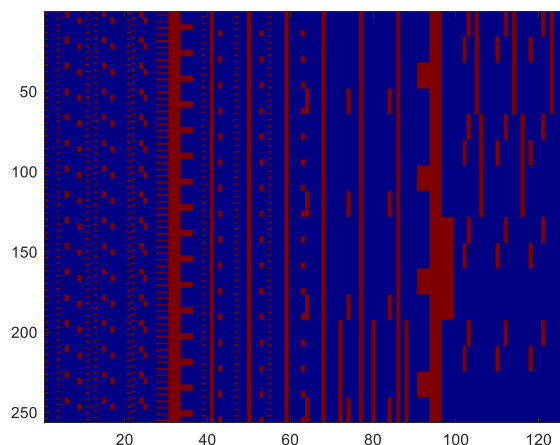
j = 3				B (Multiplicative mynnmf):			
B (ALS mynnmf):				0.4635	0.2584	1.4487	
0.0000	0.0000	1.1560		0.1075	0.0991	0.3664	
0.0000	0.0569	0.2762		0.1293	1.2632	0.7124	
0.0000	1.2183	0.7311		0.7164	0.0609	1.3310	
0.0000	0.4033	0.7989		0.4092	0.3119	0.7877	
0.0000	0.5820	0.4832		0.7818	0.1058	0.4751	
0.0000	1.1668	0.0000					
C (ALS mynnmf):				C (Multiplicative mynnmf):			
0.0000	0.0000	0.0000	0.0000	0.9864	0.7969	0.9547	0.4349
0.8272	0.7493	0.9363	0.6031	1.0606	0.9621	1.3362	0.8185
0.8203	0.8901	1.2614	1.1039	0.1705	0.3032	0.4348	0.5935
A_hat (ALS mynnmf):				A_hat (Multiplicative mynnmf):			
0.9483	1.0289	1.4582	1.2761	0.9782	1.0572	1.4177	1.2729
0.2736	0.2885	0.4017	0.3392	0.2735	0.2921	0.3943	0.3453
1.6075	1.5636	2.0629	1.5419	1.5887	1.5344	2.1211	1.5131
0.9889	1.0132	1.3853	1.1251	0.9981	1.0330	1.3441	1.1514
0.8778	0.8662	1.1545	0.8845	0.8687	0.8650	1.1499	0.9008
0.9651	0.8742	1.0924	0.7037	0.9643	0.8688	1.0943	0.7085
Error (ALS mynnmf):				Error (Multiplicative mynnmf):			
0.1370				0.1226			
B (ALS nnmf):				B (Multiplicative nnmf):			
2.3926	0	0		0.5485	0.7629	1.1790	
0.5716	0.0892	0		0.2103	0.1769	0.2982	
1.5086	1.9239	0		2.6814	0.2129	0.5798	
1.6523	0.6352	0		0.1293	1.1790	1.0832	
0.9983	0.9184	0		0.6620	0.6735	0.6411	
0	1.8438	0		0.2246	1.2866	0.3866	
C (ALS nnmf):				C (Multiplicative nnmf):			
0.3969	0.4306	0.6100	0.5338	0.4996	0.4532	0.6295	0.3856
0.5243	0.4751	0.5940	0.3828	0.5993	0.4842	0.5801	0.2642
0	0	0	0	0.2094	0.3725	0.5343	0.7293
A_hat (ALS nnmf):				A_hat (Multiplicative nnmf):			
0.9497	1.0301	1.4595	1.2771	0.9782	1.0572	1.4177	1.2729
0.2736	0.2885	0.4017	0.3392	0.2735	0.2921	0.3943	0.3453
1.6075	1.5636	2.0630	1.5418	1.5887	1.5344	2.1211	1.5131
0.9889	1.0132	1.3853	1.1252	0.9981	1.0330	1.3441	1.1514
0.8778	0.8662	1.1545	0.8845	0.8687	0.8650	1.1499	0.9008
0.9667	0.8760	1.0951	0.7059	0.9643	0.8688	1.0943	0.7085
Error (ALS nnmf):				Error (Multiplicative nnmf):			
0.1371				0.1226			

ملاحظه می‌شود که \hat{A} های به دست آمده از mynnmf و nnmf بسیار شبیه به یکدیگر هستند. علت تفاوت در ماتریس‌های B و C از روش‌های مختلف در این است که در روش‌های مختلف کماکان ابهام در دامنه و ترتیب منابع داریم.



ملاحظه می‌شود با اضافه کردن تعداد منابع (رنگ‌های مختلف نمودار)، خطا بصورت کلی کاهش می‌یابد که مطابق انتظار است. تعداد دقیق منابع ۳ عدد است که البته با ۴ عدد هم می‌توان به خطای دقیقاً برابر با صفر رسید. کافی است الگوریتم را برای تعداد منابع ۳ و ۴ با تکرار بیشتری اجرا کنیم تا به خطای دقیقاً صفر برسیم. هم‌چنین در وضعیت یکسان مشاهده می‌شود خطای حاصل از الگوریتم Multiplicative، کمتر از الگوریتم ALS است و در مواردی الگوریتم ALS حتی ناپایدار می‌شود (نمودار بالا سمت راست به ازای $j = 4$ الگوریتم ALS ناپایدار شده است). و در آخر هرچه نسبت سیگنال به نویز (SNR) بیشتر باشد، خطای الگوریتم نیز کمتر است که این مورد هم مطابق انتظار است.

۲. ابتدا به ازای همه تعداد منابع ممکن ($1 \leq j \leq 126$) الگوریتم ALS و Multiplicative را روی ماتریس مشاهدات اعمال می‌کنیم و نرم خطا را بر حسب تعداد منابع رسم می‌کنیم.



با توجه به نمودارها مشاهده می‌کنیم که الگوریتم Multiplicative بصورت کلی خطای بسیار کمتری از الگوریتم ALS دارد. بنابراین در این مساله از الگوریتم Multiplicative استفاده می‌کنیم. هم‌چنین در نمودار خطای الگوریتم Multiplicative کمترین خطا به ازای $j = 16$ به دست آمده است و لذا تعداد منابع را ۱۶ عدد در نظر می‌گیریم.

