به نام خدا

Blind Source Separation (BSS)

تکلیف شماره

8

محمدرضا آرانی

810100511

دانشگاه تهران

1402/03/08

جدول محتویات

بخش اول:

در این تمرین می خواهیم جداسازی کور منابع را با فرض استقلال منابع حل کنیم.

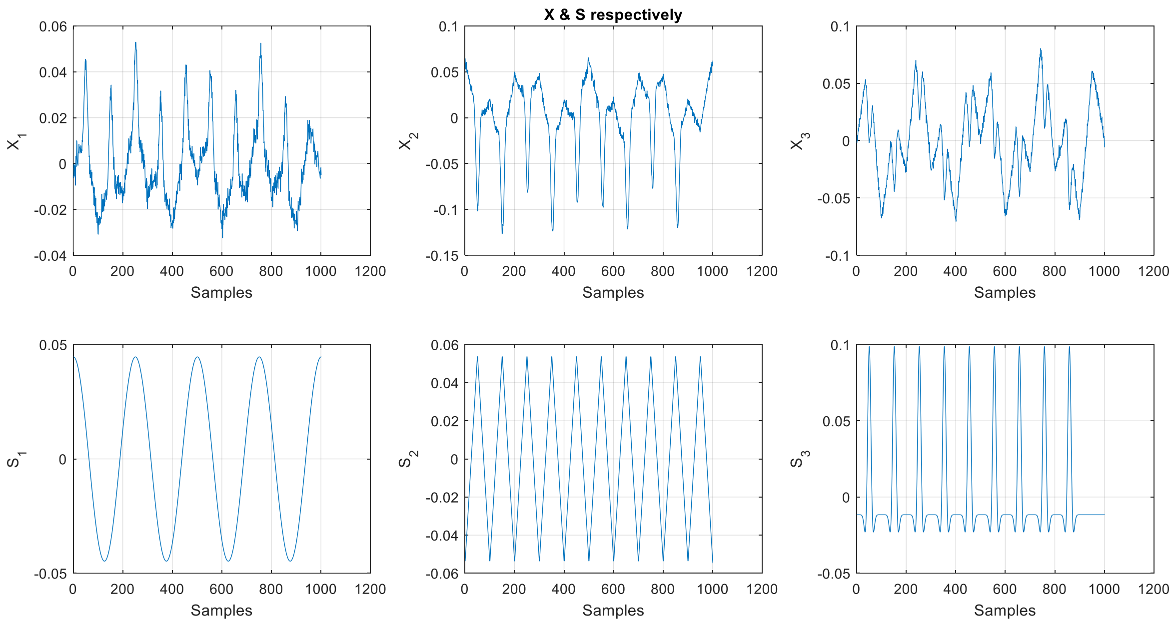
ماتریس مخلوط کننده A ، ماتریس منابع S و ماتریس Noise در فایل hw8.mat در اختیار شما قرار داده شده است. ابتدا ماتریس مشاهدات X را با رابطه ی X = A S + Noise به دست آورید. هم منابع و هم مشاهدات بدون نویز و هم مشاهدات نویزی را رسم کنید تا ظاهر آنها را ببینید. حال به دید جداسازی کور منابع به مساله نگاه کنید. در واقع فرض می کنیم فقط ماتریس X را داریم و تعداد منابع را هم می دانیم. استراتژی ما این خواهد بود که با ضرب یک ماتریس جدا کننده B در ماتریس X، خروجی هایی تولید کنیم که این خروجی ها تا حد ممکن از هم مستقل باشند.

روش اولی که در کلاس مبتنی بر کمینه سازی D_{KL} بیان شد را پیاده سازی کنید. برای تخمین تابع رتبه از روش MSE استفاده کنید. تخمین را خطی در نظر گرفته و کرنل $k(y) = [\,1 \quad y \quad y^2 \quad y^3 \quad y^4 \quad y^5\,]^T$ را برای تخمین در نظر بگیرید.

ابتدا متناسب با داده‌های داده شده، اقدام به رسم S و X که بردار منابع و بردار مشاهدات ما هستند می‌کنیم:

```
Data_hw8 = load("hw8.mat");


A = Data_hw8.A;
S = Data_hw8.S;
Noise = Data_hw8.Noise;


X = A*S+Noise;


figure()
subplot(2,3,1)
plot(X(1,:))
```

```
grid on
xlabel("Samples")
ylabel("X_1")




subplot(2,3,2)
plot(X(2,:))
grid on
xlabel("Samples")
ylabel("X_2")
title("X & S respectively")
subplot(2,3,3)
plot(X(3,:))
grid on
xlabel("Samples")
ylabel("X_3")
subplot(2,3,4)
plot(S(1,:))
grid on
xlabel("Samples")
ylabel("S_1")
subplot(2,3,5)
plot(S(2,:))
grid on
xlabel("Samples")
ylabel("S_2")
subplot(2,3,6)
plot(S(3,:))
grid on
xlabel("Samples")
ylabel("S_3")
```



شکل 1

قسمت-1:

۱- ماتریس جدا کننده ای که در نهایت به دست آوردید را در ماتریس مخلوط کننده ی اصلی ضرب کنید و حاصل را گزارش کنید. ماتریس حاصل باید نزدیک به یک ماتریس permutation باشد به این معنی که در هر سطر و هر ستون فقط یک مقدار غیر صفر داشته باشد.

از قبل میدونیم که صرف *uncorrelated* بودن منابع مارو به جواب یکتا نمیرسونه!

برای همین منظور به سراغ پاسخی می‌رویم که مربوط به ***استقلال منابع*** است!

اساس قضیه *ICA* مبتنی بر قضیه‌ی *darmoia* هست!

- در این ابزار، نکته‌ای که هست عدم توانایی در جداسازی و تشخیص منابع گوسی هست. چرا که شرط مستقل بودن دو منبع گوسی، *uncorrelated* بودن آن دو هست.

برای معیار استقلال دو متغیر از دیورژانس کالبک *(Kullback–Leibler divergence)* استفاده می‌کنیم:

$$D_{KL}\big(f(x)||g(x)\big) = \int \big(f(x)\log(f(x)/g(x)\,)\,dx$$

می‌دانیم این دیورژانس همواره مثبت است!

$$-D_{KL}(f||g) = \int f(x)\log\left(\frac{g(x)}{f(x)}\right)dx;\ \log(a) \le a - 1;$$

پس داریم:

$$-D_{KL} \le \int f(x)\left(\frac{g(x)}{f(x)} - 1\right)dx = 1 - 1 = 0; \rightarrow D_{KL} \ge 0;$$

پس به این ترتیب این معیار استقلال ما شکل گرفت! حال باید به دنبال تابع هزینه باشیم تا دو یا چند متغیر را به نحوی تعیین کنیم که نسبت به هم بیشترین استقلال را با معیار KL داشته باشند.

تابع هزینه را به صورت زیر تعیین می‌کنیم:

$$f(B) = D_{KL}\left(P_{y1y2\ldots y_M}(y1, y2, \ldots, y_M) \,||P_{y1}(y1)P_{y2}(y2)\ldots P_{y_M}(y_M) \right)$$

و خوب صورت مسئله‌ی بهینه‌سازی ما به فرم زیر درخواهد آمد:

$$\hat{B} = argmin\, f(B)$$

$$s.t.\quad b_i^T b_i = 1, \quad i = 1, \ldots, M = N$$

فرض می‌شود که تعداد منابع با مشاهدات برابر است!

با تعریفی که از آنتروپی به صورت زیر ارائه می‌دهیم، تابع هزینه‌ی ما به صورت زیر درخواهد آمد:

$$H(x) \triangleq \sum_i \left(P_i \log\left(\frac{1}{P_i}\right)\right) = E\left\{\log\left(\frac{1}{P}\right)\right\}$$
$$= \int \left(P_x(x)\log\left(\frac{1}{P_x(x)}\right)\right) dx$$

پس در نتیجه با این تعریف داریم:

$$\boxed{f(B) = \sum_{m=1\ to\ M} \left(H(y_m)\right) - H(y)}$$

$$y = Bx$$

حال برای بهینه‌سازی این تابع هزینه، از روش گرادیان کاهشی استفاده می‌کنیم:

$$B_{new} = B_{old} - \mu \frac{\partial}{\partial B} f$$

$$b_i^T b_i = 1$$

و همچنین برای گرادیان تابع f نسبت به B داریم:

$$\frac{\partial}{\partial B} f = \frac{\partial}{\partial B}\left(\sum_m H(y_m)\right) - \frac{\partial}{\partial B} H(y)$$
$$= E\{\Psi_Y(Y)x^T\} - B^{-T}$$

که در اینجا Ψ همان *score function* توزیع ما هست!

- نکته‌ی مهم عمود بودن سطرهای B بر هم است! باید پس از پیدا کردن هر سطر آن را نرمالایز کنیم بر سطرهای پیشین پیدا شده!

نحوه‌ی پیداکردن مقدار *score function*:*(MSE method)*:

$$\widehat{\Psi}_y(y) = \theta^T K(y) = \theta_1 K_1(y) + \theta_2 K_2(y) + \cdots + \theta_N K_N(y);$$

که در صورت سوال به ما $K(y)$ داده شده است!

$$k(y) = [\,1 \quad y \quad y^2 \quad y^3 \quad y^4 \quad y^5\,]^T$$

$$for\ g(\theta) = E\{\theta^T K(y) K^T(y)\theta\} - 2E\{\theta^T K'(y)\}$$

$$\hat{\theta} = (E\{K(y)K^T(y)\})^{-1} E\{K'(y)\}$$

پس با توجه به $K(y)$ فعلی، $\hat{\theta}$ ما به صورت زیر خواهد بود:

$$\hat{\theta} = \left(E\left\{ \begin{bmatrix} 1 \\ y \\ y^2 \\ y^3 \\ y^4 \\ y^5 \end{bmatrix} [1, y, y^2, y^3, y^4, y^5] \right\} \right)^{-1} E\left\{ \begin{bmatrix} 0 \\ 1 \\ 2y \\ 3y^2 \\ 4y^3 \\ 5y^4 \end{bmatrix} \right\}$$

پس داریم:

$$\hat{\theta} = \left(\begin{bmatrix} 1 & m1 & m2 & m3 & m4 & m5 \\ m1 & m2 & m3 & m4 & m5 & m6 \\ m2 & m3 & m4 & m5 & m6 & m7 \\ m3 & m4 & m5 & m6 & m7 & m8 \\ m4 & m5 & m6 & m7 & m8 & m9 \\ m5 & m6 & m7 & m8 & m9 & m10 \end{bmatrix} \right)^{-1} E\left\{ \begin{bmatrix} 0 \\ 1 \\ 2y \\ 3y^2 \\ 4y^3 \\ 5y^4 \end{bmatrix} \right\}$$

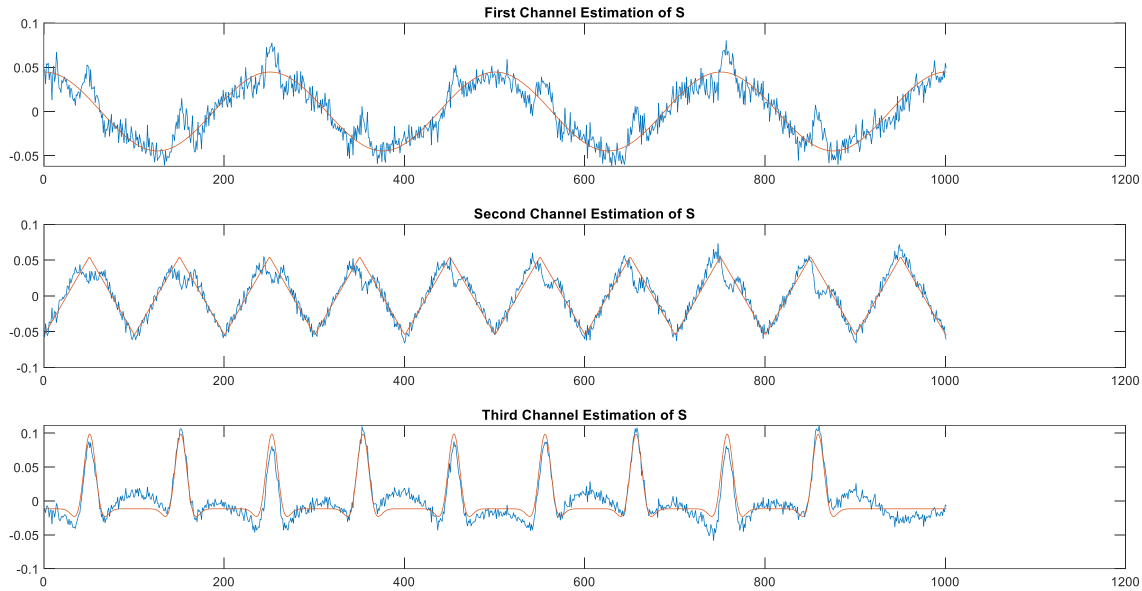
که در آن m_i نمایانگر ممان مرتبه‌ی iام است.

قسمت-2:

۲- ابهام ترتیب و همچنین ابهام scale منابع را برطرف کرده به این معنی که انرژی منابع تخمین زده شده را مساوی انرژی منابع اصلی کنید. منابع تخمین زده شده را روی منابع اصلی رسم کنید. سپس مقدار خطای زیر را گزارش کنید.

$$E = \frac{\|\hat{S} - S\|_F^2}{\|S\|_F^2}$$

پس از پیاده‌سازی و رفع ابهام دامنه و permutation داریم:



شکل 2

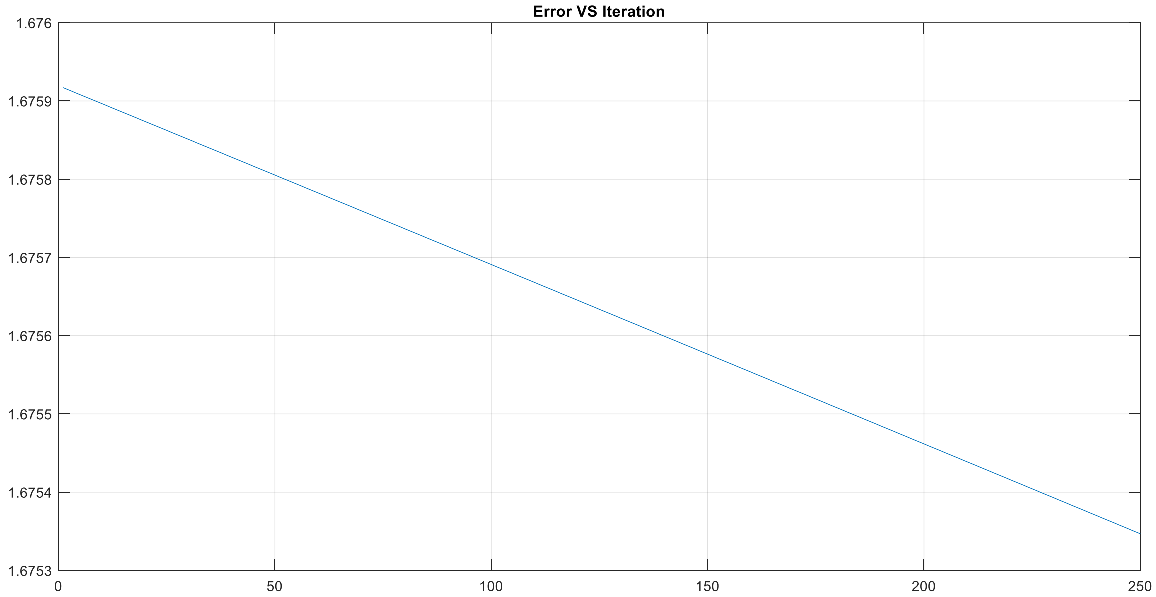
خطای این مورد به میزان زیر در نهایت رسیده است:

```
Y_True = [-y(3,:) ; -y(2,:) ; y(1,:)   ] ;
% Calc True Error:
Error_FInal = norm(Y_True  - S)/norm(S);
disp("Final Error AFter Removing Permutaion : "+Error_FInal)
Final Error AFter Removing Permutaion : 0.46328
```

قسمت-3:

۳- نمودار همگرایی (تابع هدف بر حسب شماره ی iteration) را رسم کنید.

نمودار خطا بدون رفع ابهام دامنه و ترتیب در فرآیند اصلاح سطرهای B به صورت زیر است:



شکل 3

و همینطور در نهایت کدی که پیاده‌سازی شد به صورت زیر است:

```
% Whitening:
[U , Gamma] = eig(X*X');
W = Gamma^(-0.5);
Z = W*U'*X;








B    = randn(size(inv(A)));
[B,~]    = eig(B*B');  % Enforce Orthogonality
y = B*Z;


Num_of_Channels = size(y);
Score_Func_y = zeros(size(y));
K_y = zeros(Num_of_Channels(1,1) , M,length(y));
```

```
mu = 1e1;


IterMax = 250;
Error_Iter_deflate = zeros(1,IterMax);
MIN_ERR = inf;
for p=1:IterMax



    for i=1:length(B) % Each Row
         % Estimation of Score Function:
         [Theta_hat , Score_Func_y ] = Theta_Calc_Kernel(y(i,:));
         StepSize = ( Score_Func_y*Z' )/length(Z) ;
        %  StepSize = normalize(StepSize,2,"norm");
         B(i,:)        = B(i,:) - mu*StepSize  ;
         B(i,:)        =  ( eye(size(B)) - B(1:i-1,:)'*B(1:i-1,:)   )* B(i,:)';  %
Orthogonality
         B(i,:)  = B(i,:)/norm(B(i,:));
    end
    y = B*Z;


 % Remove Permutation:
%  [Errors_ICA,y]  = Perm_AMP_Disamb(y,S);
 Errors_ICA = norm(y-S)/norm(S);
 Error_Iter_deflate(p) = min(Errors_ICA);


 if(MIN_ERR > Error_Iter_deflate(p))
     y_Best = y;
     Index_Best = p;
     MIN_ERR = Error_Iter_deflate(p);
 end


end
```

این روش با سفید کردن صرفا عبارت دوم stepsize *را حذف می‌کند چرا که پس از سفیدسازی این ماتریس* B *ما مجبور است یک ماتریس* orthonormal *باشد. پس دترمینان آن همواره عدد ثابتی است و مشتق دترمینان آن برابر 0 است.*

در نهایت مقدار نهایی گزارش شده برای ماتریس مخلوط کننده‌ی نهایی برابر:

```
disp(abs(B*W*A))
    0.6384    1.1808    1.4166
    0.1072    0.5190    0.2491
    1.2456    0.5540    0.0590
```

است که در آن در یک سطر سعی شده فقط یک مقدار بیشینه بوده و بقیه از آن بسیار کمتر باشند.

پایان