



به نام خدا



دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

**BSS**

گزارش تمرین ۱۰

سالار صفردوست

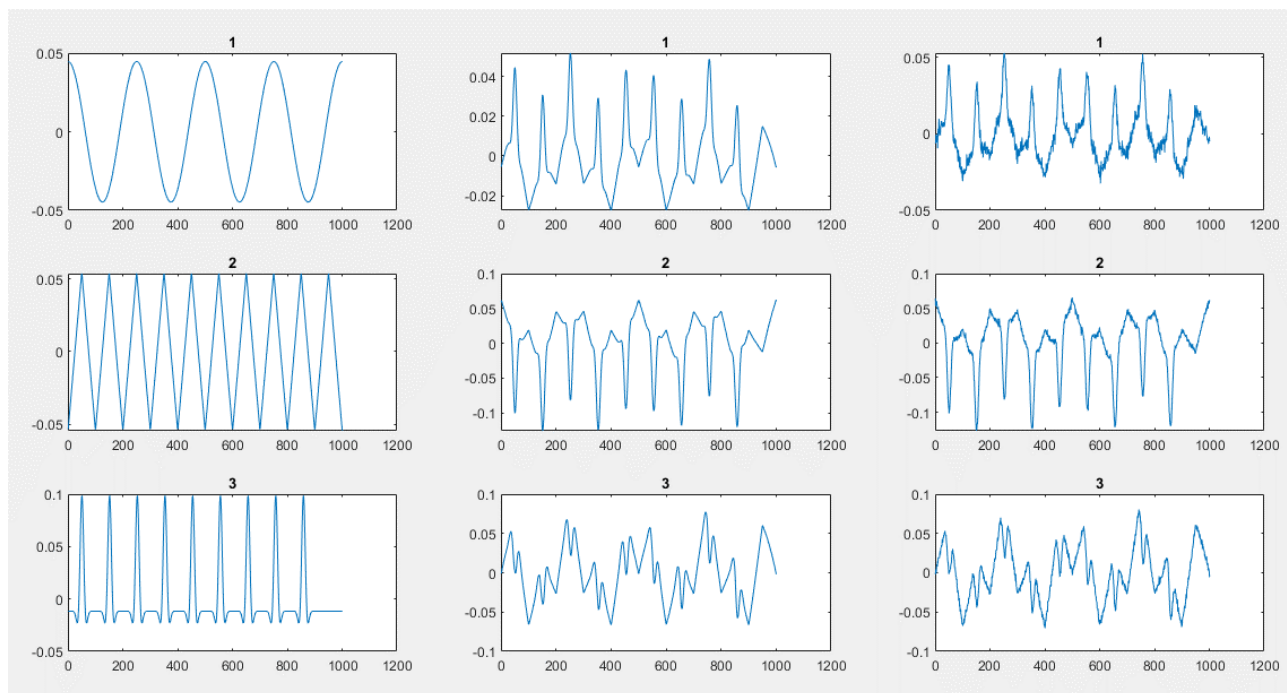
۸۱۰۱۹۹۴۵۰

۱۴۰۲/۰۳/۳۰

```

3 %% Preprocessing
4
5 load hw10.mat
6
7 X_raw = A*S;
8 X = A*S+Noise;
9
10 figure
11 for i = 1:size(X,1)
12     subplot(3,3,3*i-2)
13     plot(S(i,:))
14     title(i)
15     subplot(3,3,3*i-1)
16     plot(X_raw(i,:))
17     title(i)
18     subplot(3,3,3*i)
19     plot(X(i,:))
20     title(i)
21 end
22
23 [M,~] = size(X);
24 B = normr(rand(M,M)*2-1);
25

```



```

1 function [B,S_hat,Similarity_Cell] = ICA_Kurt(Z,miu,Threshold,B)
2
3     [M,T] = size(Z);
4
5     Similarity_Cell = cell(M,1);
6
7
8     for m = 1:M
9         Similarity = [];
10        Similarity(1) = Threshold;
11        while Similarity(end) <= Threshold
12            y = B(m,:)*Z;
13
14            kurt = mean(y.^4)-3*(mean(y.^2).^2);
15            df_db = sign(kurt)*( (Z*(y.^3).') ./ T - 3*(B(m,:)) ) ;
16
17            b = B(m,:) + miu*df_db;
18            if m~=1
19                b = ((eye(M)-B(1:m-1,:).'*B(1:m-1,:))*b.').';
20            end
21            b = normr(b);
22            Similarity = [Similarity,B(m,:)*b.'];
23            B(m,:) = b;
24        end
25        Similarity_Cell{m,1} = Similarity;
26    end
27
28    S_hat = B*Z;
29
30 end
31

```

```

26 %% Question 1_1
27
28 miu = 0.1;
29 Threshold = 1-1e-10;
30
31 [U,L] = eig(X*X. ');
32 W = L^-(1/2) * U. ';
33 Z = W*X;
34
35 [B,S_hat,Similarity_Cell] = ICA_Kurt(20*Z,miu,Threshold,B);
36
37 Final_Matrix = B*W*A;
38

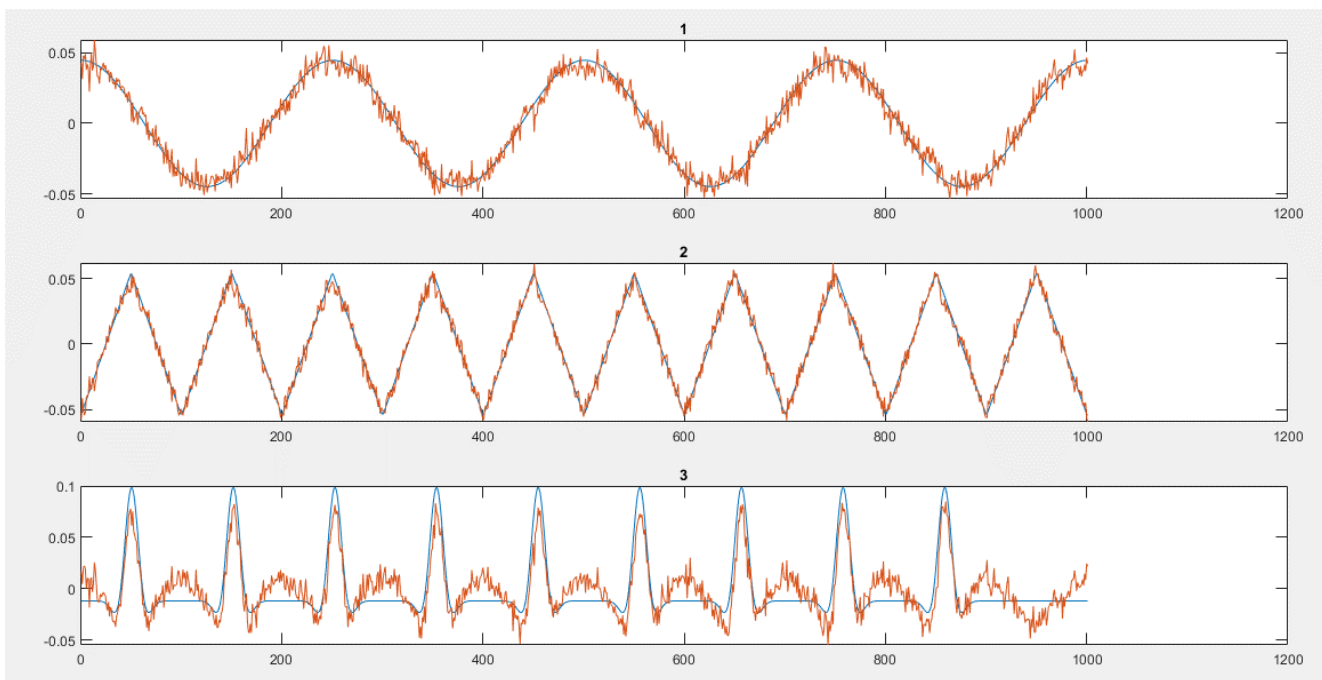
```

Variables - Final_Matrix				
3x3 double				
	1	2	3	
1	-0.9897	-0.0690	0.0745	
2	-0.0132	1.0202	-0.0570	
3	-0.0431	0.5378	-1.1376	

```

39 %% Question 1_2
40
41 S_hat = Scale_Permutation_Recovery(S,S_hat);
42
43 figure
44 for n = 1:size(S,1)
45     subplot(3,1,n)
46     plot(S(n,:),hold on,plot(S_hat(n,:)));
47     title(n)
48 end
49
50 Error1 = (norm(S-S_hat,'fro')/norm(S,'fro'))^2;
51

```



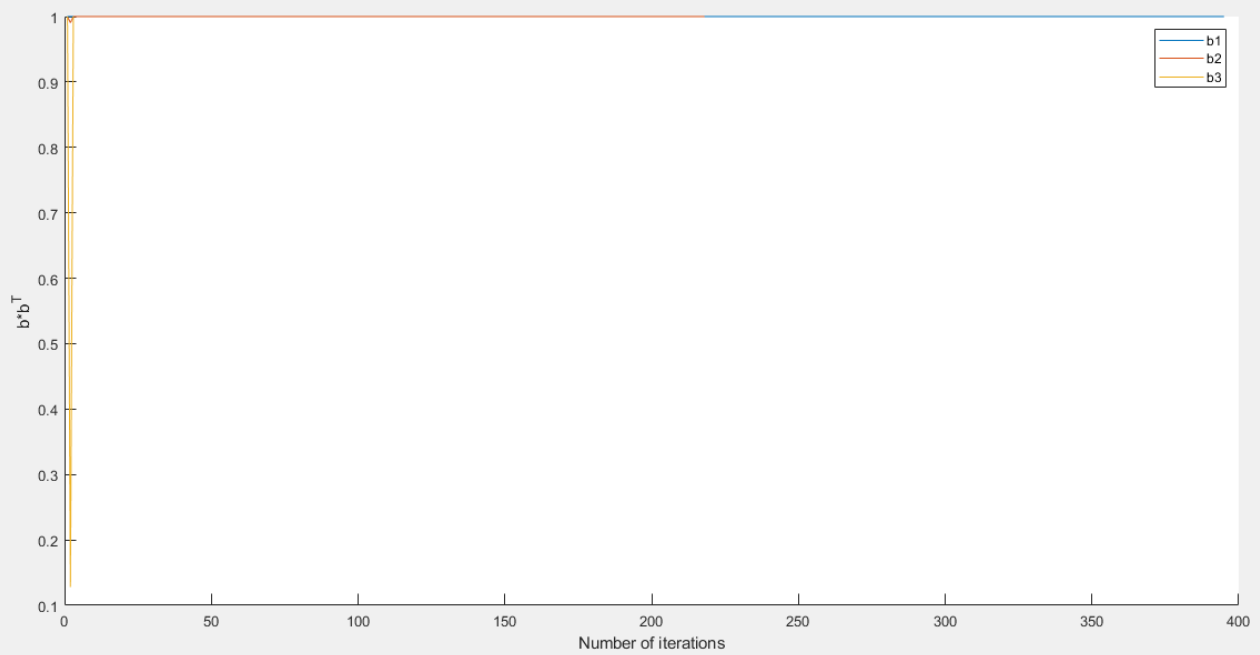
Error1

0.1152

```

52 %% Question 1_3
53
54 figure
55 hold on
56 for m = 1:numel(Similarity_Cell)
57     plot(Similarity_Cell{m,1})
58 end
59 legend('b1','b2','b3')
60 xlabel('Number of iterations')
61 ylabel('b*b^{T}')
62

```



```

1 function [B,S_hat,Similarity_Cell] = ICA_FP(Z,Threshold,B)
2
3     [M,T] = size(Z);
4
5     Similarity_Cell = cell(M,1);
6
7
8     for m = 1:M
9         Similarity = [];
10        Similarity(1) = Threshold;
11        while Similarity(end) <= Threshold
12            y = B(m,:)*Z;
13
14            df_db = ((Z*(y.^3).')).' / T - 3*(B(m,:));
15
16            b = df_db;
17            if m~=1
18                b = ((eye(M)-B(1:m-1,:)).'*B(1:m-1,:))*b.').';
19            end
20            b = normr(b);
21            Similarity = [Similarity,abs(B(m,:)*b.')]';
22            B(m,:) = b;
23        end
24        Similarity_Cell{m,1} = Similarity;
25    end
26
27    S_hat = B*Z;
28
29 end

```

```

63 %% Question 2_1
64
65 Threshold = 1-1e-10;
66
67 [U,L] = eig(X*X. ');
68 W = L^-(1/2) * U. ';
69 Z = W*X;
70
71 [B,S_hat,Similarity_Cell] = ICA_FP(20*Z,Threshold,B);
72
73 Final_Matrix = B*W*A;
74

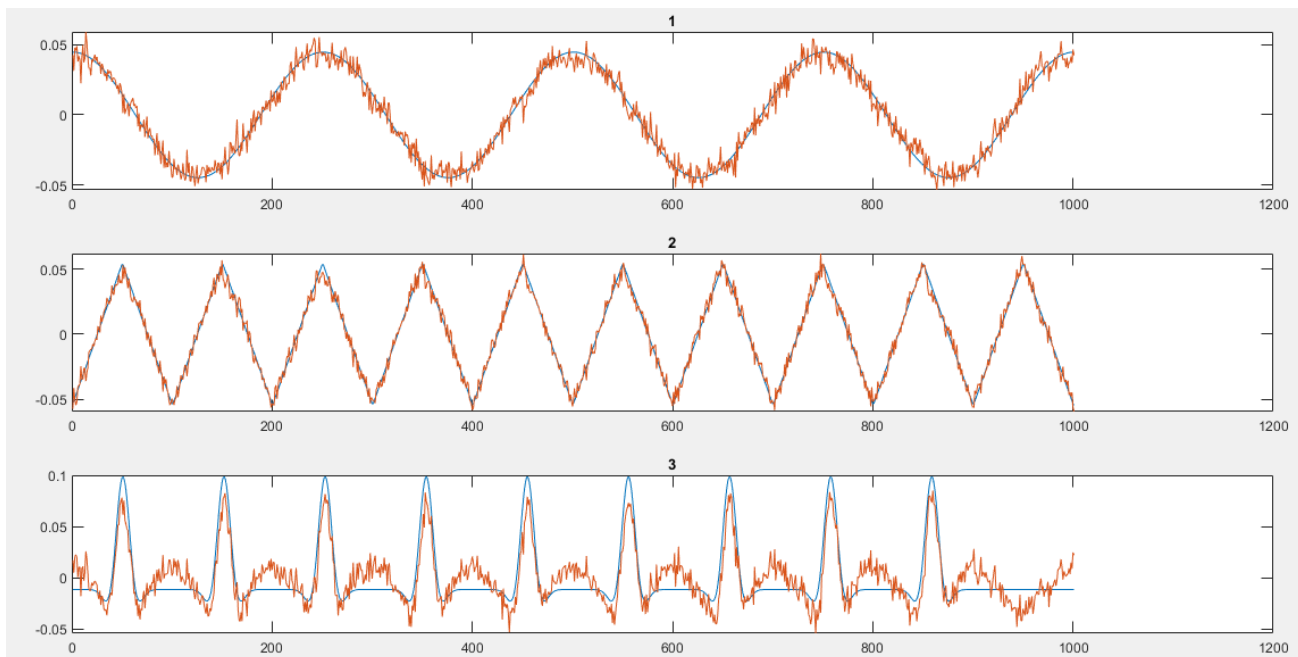
```

Variables - Final_Matrix				
3x3 double				
	1	2	3	
1	0.9897	0.0697	-0.0750	
2	0.0137	-1.0204	0.0574	
3	-0.0435	0.5373	-1.1376	

```

75 %% Question 2_2
76
77 S_hat = Scale_Permutation_Recovery(S,S_hat);
78
79 figure
80 for n = 1:size(S,1)
81     subplot(3,1,n)
82     plot(S(n,:),hold on,plot(S_hat(n,:));
83     title(n)
84 end
85
86 Error2 = (norm(S-S_hat,'fro')/norm(S,'fro'))^2;
87

```



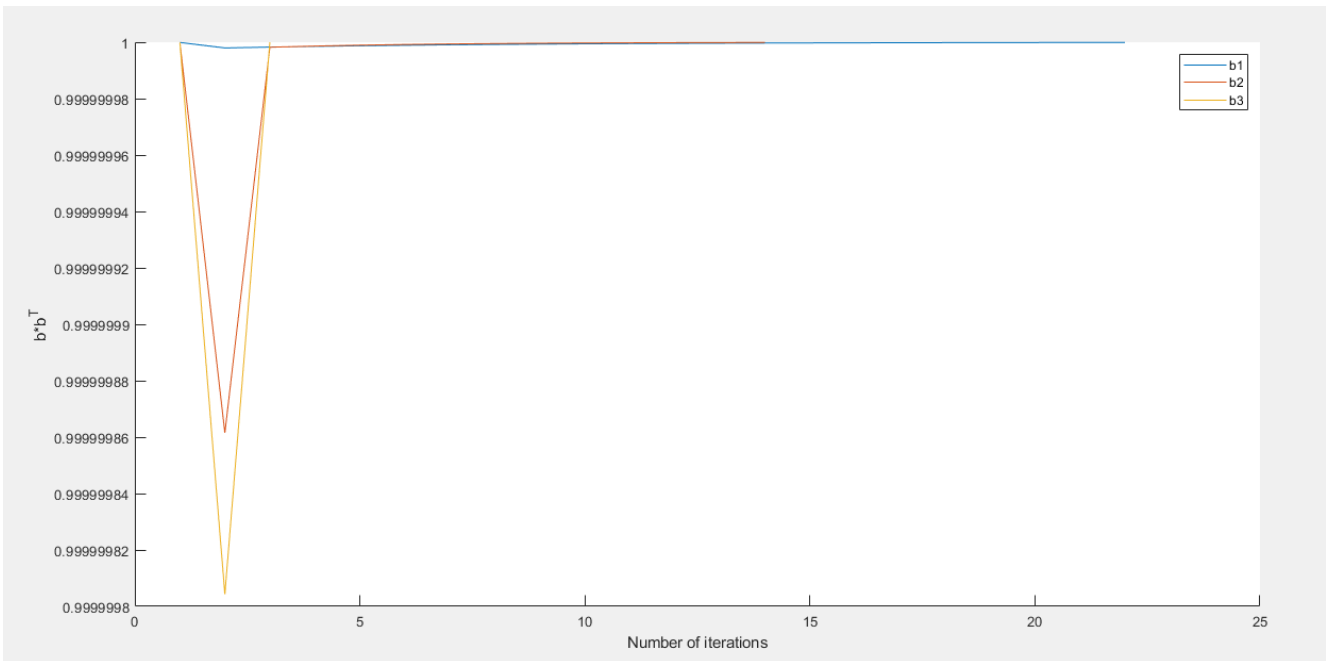
Error2

0.1151

```

88 %% Question 2_3
89
90 figure
91 hold on
92 for m = 1:numel(Similarity_Cell)
93     plot(Similarity_Cell{m,1})
94 end
95 legend('b1','b2','b3')
96 xlabel('Number of iterations')
97 ylabel('b*b^{T}')
98

```





```

1  function [B,S_hat,Similarity_Cell] = ICA_Kurt_G(Z,miu,Threshold,B)
2
3  [M,T] = size(Z);
4
5  Similarity_Cell = cell(M,1);
6
7
8  for m = 1:M
9      Similarity = [];
10     Similarity(1) = Threshold;
11     while Similarity(end) <= Threshold
12         y = B(m,:)*Z;
13
14         g_y = y.*exp(-y.^2./2);
15
16         df_db = (( Z*(g_y)'/T )).';
17
18         b = B(m,:) + miu*df_db;
19         if m~=1
20             b = ((eye(M)-B(1:m-1,:).'*B(1:m-1,:))*b.').';
21         end
22         b = normr(b);
23         Similarity = [Similarity,B(m,:)*b.'];
24         B(m,:) = b;
25     end
26     Similarity_Cell{m,1} = Similarity;
27 end
28
29 S_hat = B*Z;
30
31 end

```

```

99 %% Question 3_1
100
101 miu = 0.1;
102 Threshold = 1-1e-10;
103
104 [U,L] = eig(X*X. ');
105 W = L^-(1/2) * U. ';
106 Z = W*X;
107
108 [B,S_hat,Similarity_Cell] = ICA_Kurt_G(20*Z,miu,Threshold,B);
109
110 Final_Matrix = B*W*A;
111

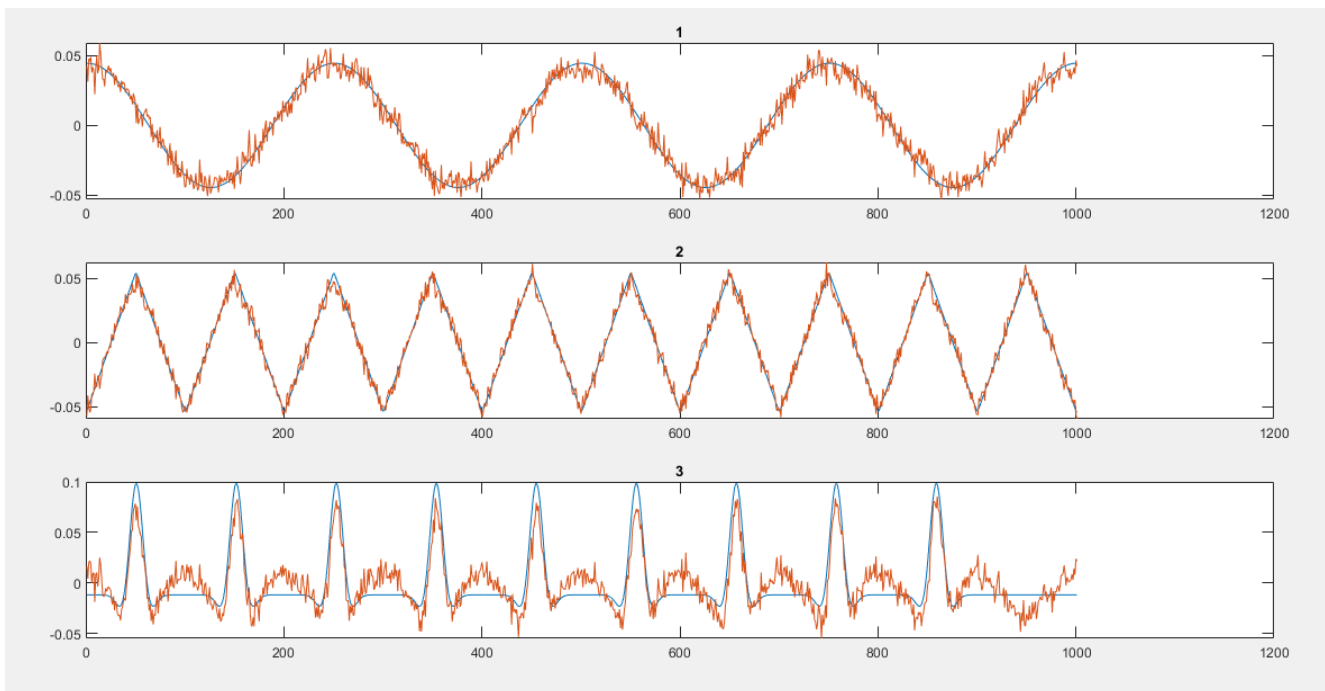
```

Variables - Final_Matrix			
3x3 double			
	1	2	3
1	0.9899	0.0658	-0.0689
2	0.0129	-1.0237	0.0643
3	-0.0382	0.5315	-1.1376

```

112 %% Question 3_2
113
114 S_hat = Scale_Permutation_Recovery(S,S_hat);
115
116 figure
117 for n = 1:size(S,1)
118     subplot(3,1,n)
119     plot(S(n,:),hold on,plot(S_hat(n,:));
120     title(n)
121 end
122
123 Error3 = (norm(S-S_hat,'fro')/norm(S,'fro'))^2;
124

```



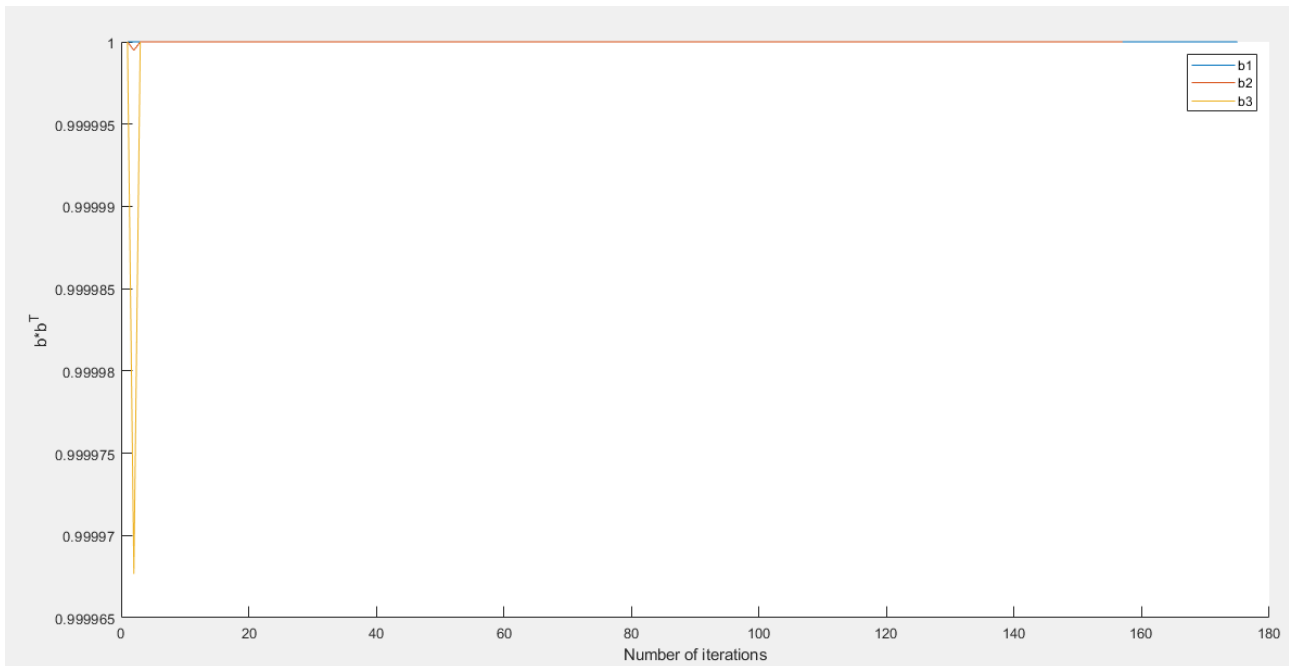
Error3

0.1135

```

125 %% Question 3_3
126
127 figure
128 hold on
129 for m = 1:numel(Similarity_Cell)
130     plot(Similarity_Cell{m,1})
131 end
132 legend('b1','b2','b3')
133 xlabel('Number of iterations')
134 ylabel('b*b^{T}')
135

```



```

1 function [B,S_hat,Similarity_Cell] = ICA_FP_G(Z,Threshold,B)
2
3     [M,T] = size(Z);
4
5     Similarity_Cell = cell(M,1);
6
7
8     for m = 1:M
9         Similarity = [];
10        Similarity(1) = Threshold;
11        while Similarity(end) <= Threshold
12            y = B(m,:)*Z;
13
14            g_y = y.*exp(-y.^2./2);
15            g_prime_y = exp(-y.^2/2) - y.*g_y;
16
17            df_db = ((Z*(g_y).')/T).' + mean(g_prime_y).*B(m,:);
18
19            b = df_db;
20            if m~=1
21                b = ((eye(M)-B(1:m-1,:).'*B(1:m-1,:))*b.').';
22            end
23            b = normr(b);
24            Similarity = [Similarity,abs(B(m,:)*b.')]';
25            B(m,:) = b;
26        end
27        Similarity_Cell{m,1} = Similarity;
28    end
29
30    S_hat = B*Z;
31
32    end

```

```

136 %% Question 4_1
137
138 Threshold = 1-1e-10;
139
140 [U,L] = eig(X*X. ');
141 W = L^-(1/2) * U. ';
142 Z = W*X;
143
144 [B,S_hat,Similarity_Cell] = ICA_FP_G(20*Z,Threshold,B);
145
146 Final_Matrix = B*W*A;
147

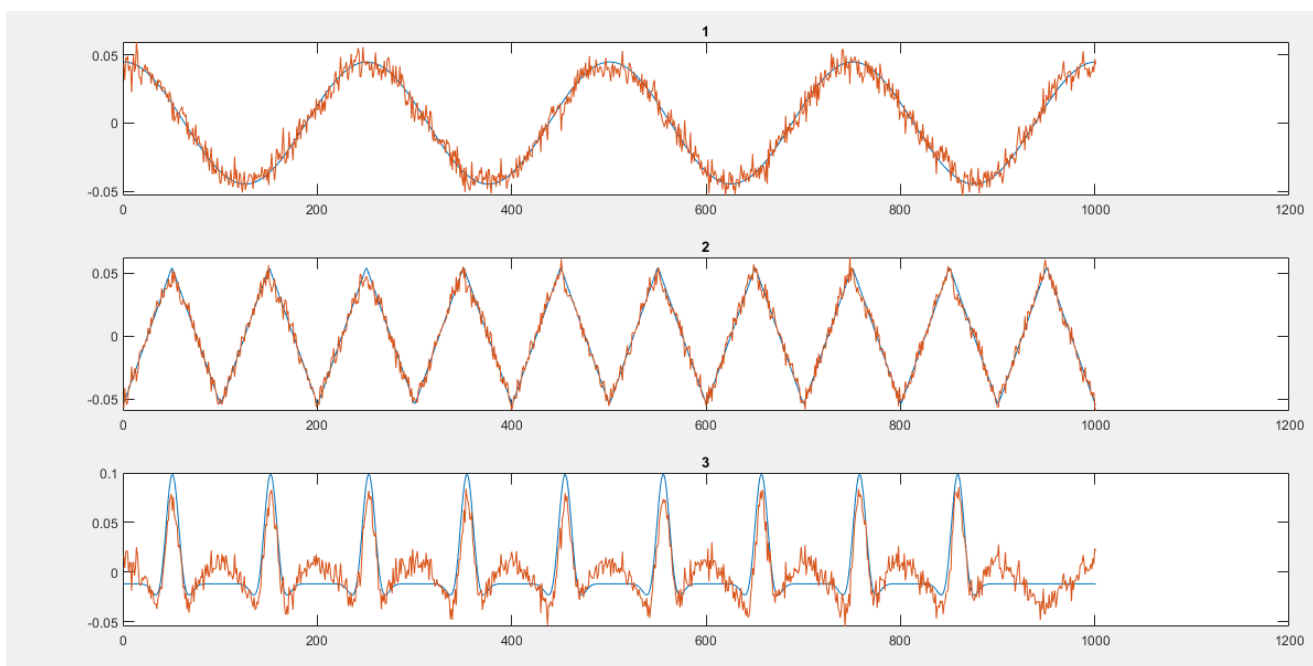
```

Variables - Final_Matrix			
3x3 double			
	1	2	3
1	0.9900	0.0644	-0.0671
2	0.0123	-1.0244	0.0657
3	-0.0367	0.5305	-1.1376

```

148 %% Question 4_2
149
150 S_hat = Scale_Permutation_Recovery(S,S_hat);
151
152 figure
153 for n = 1:size(S,1)
154     subplot(3,1,n)
155     plot(S(n,:),hold on,plot(S_hat(n,:));
156     title(n)
157 end
158
159 Error4 = (norm(S-S_hat,'fro')/norm(S,'fro'))^2;
160

```



Error4

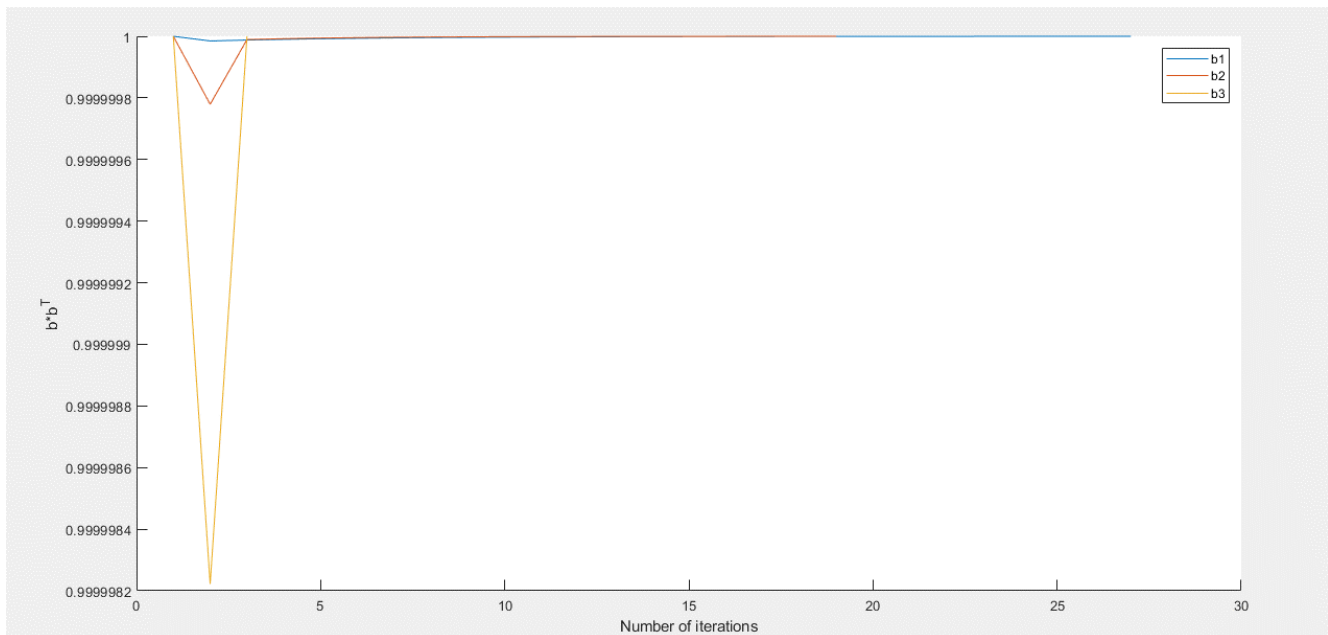
0.1132





\* در این متد در خط ۱۷ تابع مطابق جزوه باید دو تا عبارت اکسپکتهیشن از هم کم می شدند، اما در عمل اینکار موجب بدتر شدن پاسخ خروجی می گشت و به صورت تجربی مثبت و منفی علامت آن عوض گردید.

```

161 %% Question 4_3
162
163 figure
164 hold on
165 for m = 1:numel(Similarity_Cell)
166     plot(Similarity_Cell{m,1})
167 end
168 legend('b1','b2','b3')
169 xlabel('Number of iterations')
170 ylabel('b*b^{T}')

```



 Error1	0.1149
 Error2	0.1150
 Error3	0.1135
 Error4	0.1132

تمامی ۴ روش با اینیشال و Threshold یکسان اجرا شدند تا نتایج عادلانه باشد.

همانگونه که مشخص روش‌ها با جلو رفتن خطای کمتری پیدا می‌کنند و در نتیجه Fast ICA نهایی روشی است که از سایرین بهتر جلو رفته است.

در سرعت همگرایی نیز روش‌های FP بسیار سریعتر از روش‌های GP می‌باشند و سرعت همگرایی به شکل زیر می‌باشد (مجموع ایتريشن تمامی bها به طور تقریبی)

$$GP(600) < GP\ Outlier(310) < FP\ Outlier(Fast\ ICA)(50) < FP(40)$$

البته سرعت همگرایی در هر بار ران کردن تغییر می‌یابد ولی معمولاً FPها از GP سریعتر همگرا می‌شوند.