Bargaje 2017 dataset K4 Collection Time

Load the Data

```
%https://github.com/yutongo/QuanTC/blob/master/Example/QuanTC_SCC.pdf

cd 'F:\projects\BioTIP\result\Bargaje2017_EOMES\QuanTC-modified'

addpath('Input')

data = load('Bargaje2017_log2.data.txt');
prodata = data';

fid = fopen('Bargaje2017_gene_name.txt');
gene_name = textscan(fid, '%s');
progene_name = gene_name{1,1};
fclose(fid);

fid = fopen('Bargaje2017_CollectionTime.txt');
procell_label = textscan(fid, '%s');
procell_label = procell_label{1,1};
true_label = procell_label';
fclose(fid);

fid = fopen('Bargaje2017_Consensus.Cluster.txt');
procell_cell_identity = textscan(fid, '%s');
procell_cell_identity = procell_cell_identity{1,1};
fclose(fid);

file_dir = 'Output/k4_4_time';
```

Consensus clustering

If choosing SC3 package to do the consensus clustering on the processed data, we can get the following cell-cell similarity matrix M:

```
M = csvread('Bargaje2017_cell-cell.csv');
```

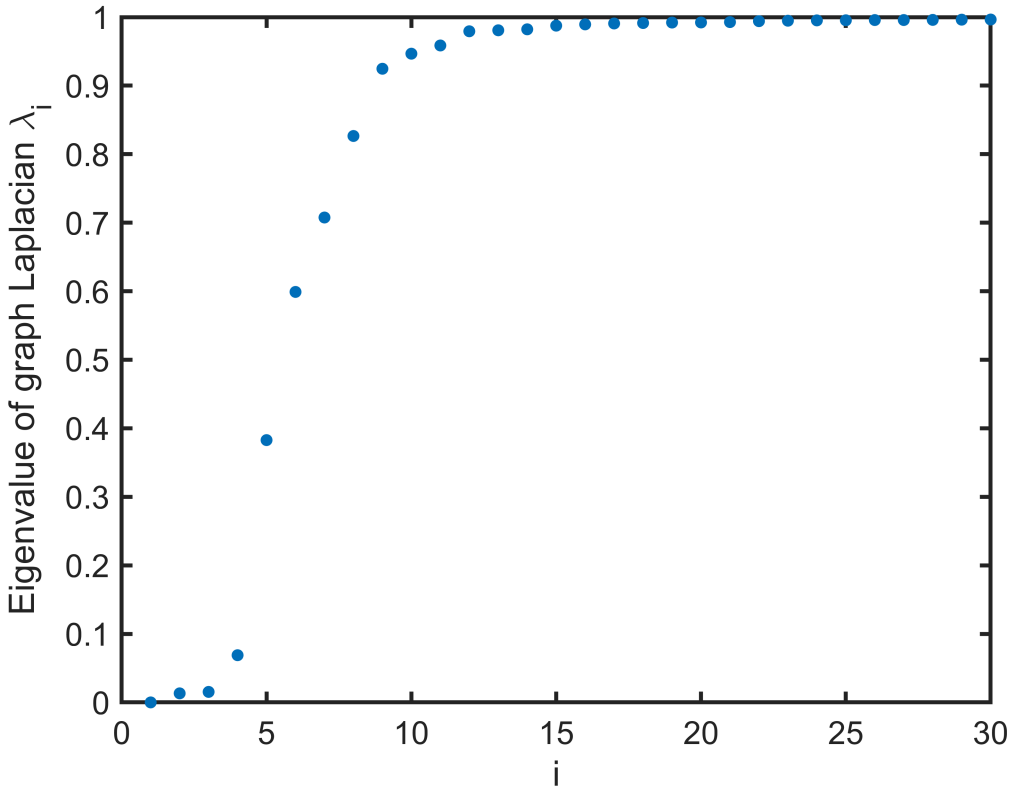
Number of clusters

The number of clusters is estimated by analyzing the largest gap of sorted eigenvalues of symmetric normalized graph Laplacian:

```
[eigenvalues] = plot_eigen_gap(M);
```

```
Number of cluster based on zero eigenvalues & Largest gap
     1     4
```

```
file_name = strcat(file_dir, '/k_selection.pdf');
saveas(gca,file_name)
```




```
No_cluster = 4;
```

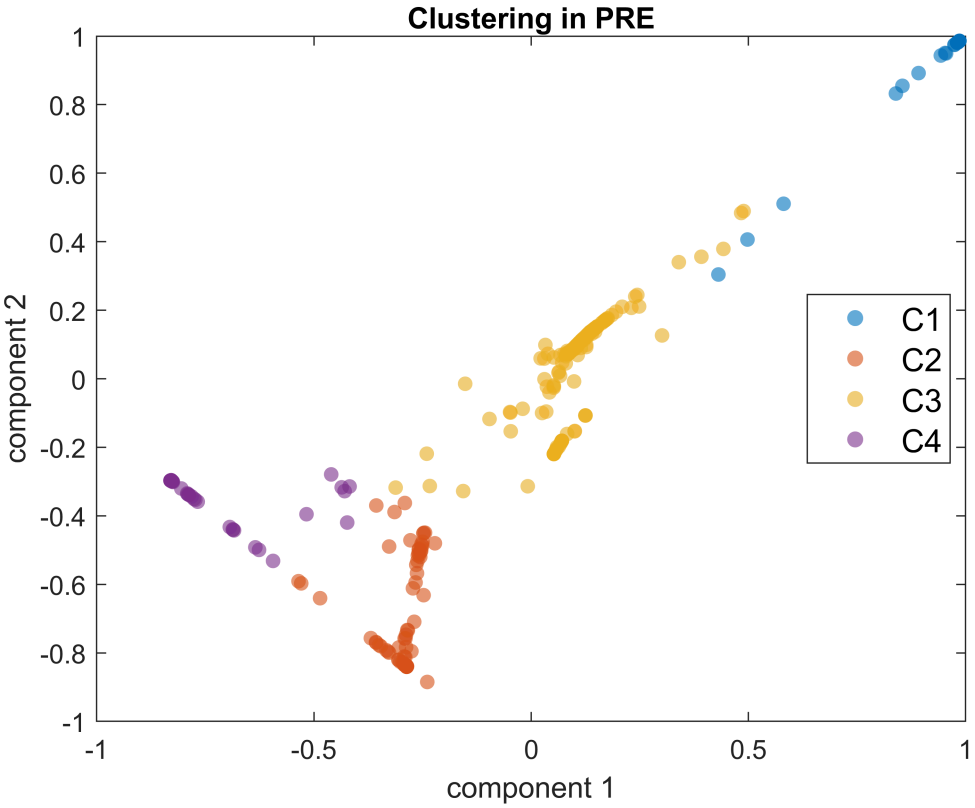
Run QuanTC

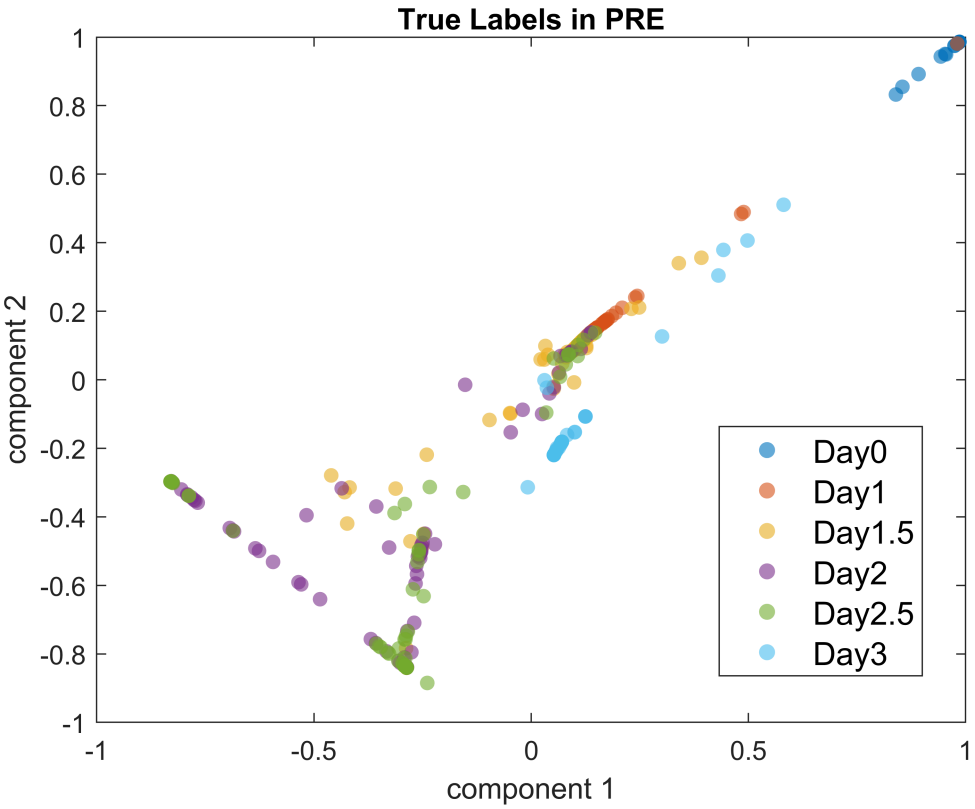
- **Soft clustering**: get the likelihoods of cells belonging to each cluster based on symmetric non-negative matrix factorization of M
- **CPI**: compute CPI value of each cell, use TC_cut to select cell with higher CPI values to be TC

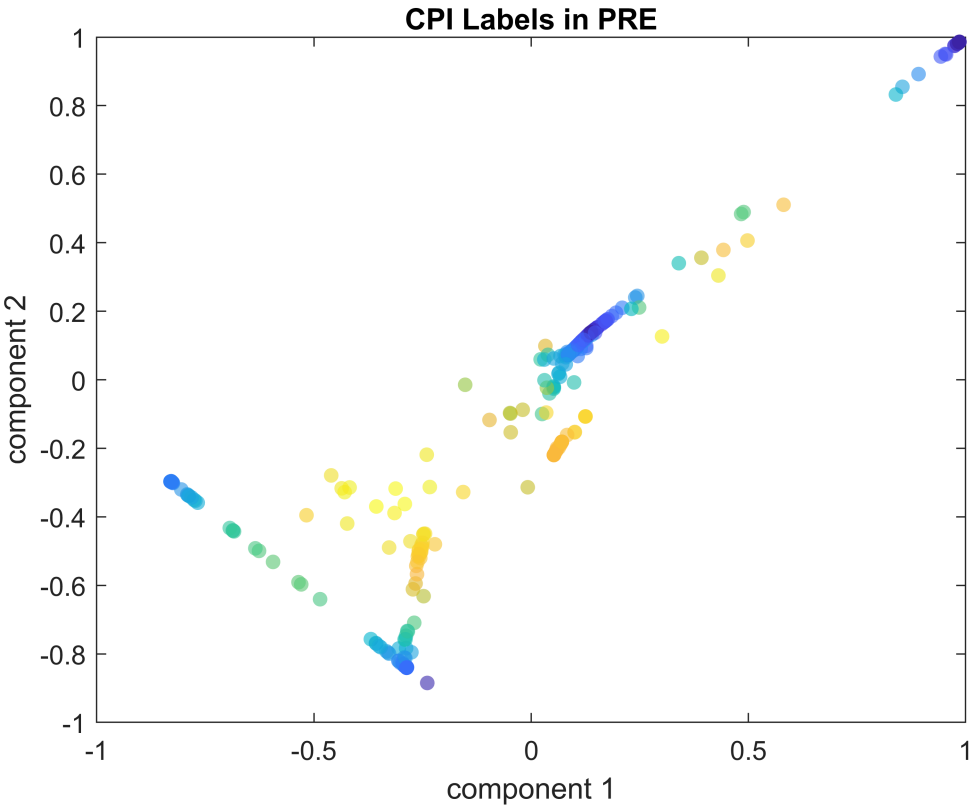
```
TC_cut = 0.34;
%[result] = run_QuanTC(prodata,M,No_cluster,TC_cut);
% saving result space
%save 'Output/k4_4_time/result.mat' result -mat

load('Output/k4_4_time/result.mat')
```

- **2d-visualization**: cells are visualized through the probabilistic regularized embedding (PRE) approach

```
label_legend =  {'Day0', 'Day1', 'Day1.5', 'Day2', 'Day2.5', 'Day3'};
cell_visu_PRE(result,procell_label, label_legend, file_dir)
```





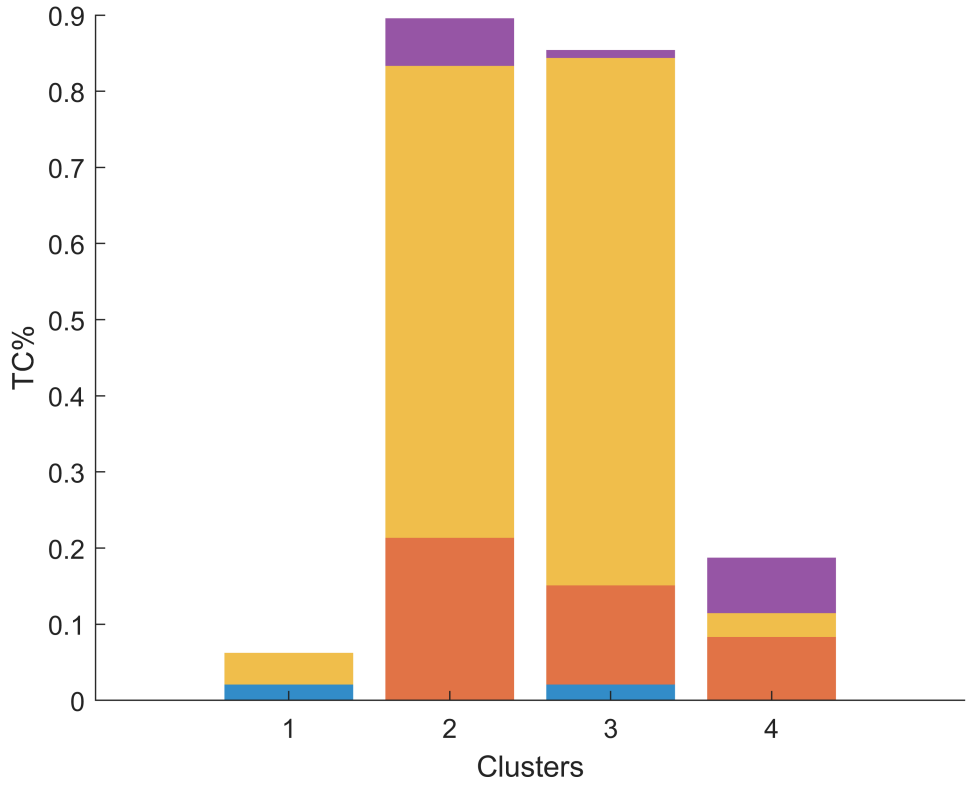


Transition trajectory

- **Start/ end cluster**: choose non-ICS based on the percentage of TC associated with each cluster relative to the total number of TC

```
pTC(result,No_cluster, file_dir)
```

```
mycolor = 13×3
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     0     0     0
     ⋮
```



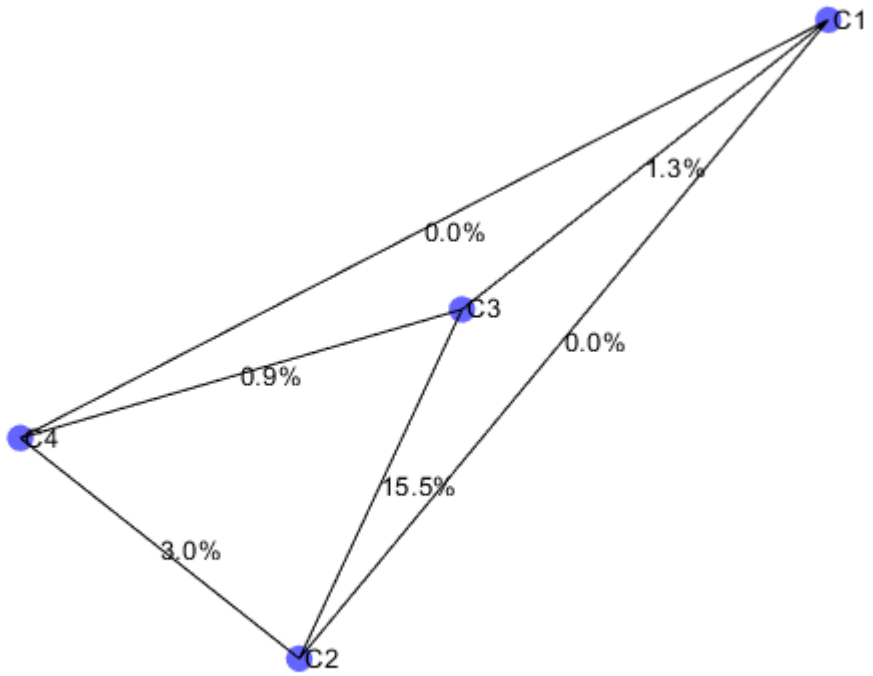
- **Potential transition trajectories**: list the percetage of cells over whole cell population to choose the potential transition trajectories

```
start_cluster = 3; %choose starting cluster based on the above graph
%plot TC% between clusters among all the cells
[path,ordered_cell] = traj(result,start_cluster,No_cluster);
```

```
trajectory: 3  2  4, percentage of cells involved: 0.71905
trajectory: 3  4, percentage of cells involved: 0.19591
```

```
fid = strcat(file_dir, '/TC%_btwn_clusters_C',num2str(start_cluster),'start.pdf');
```

```
saveas(gca,fid)
```



Finding cluster marker genes and the transition genes that mark transition from one trajectory

```
lam2 = 10;
path_num = 1;
[marker_gene,transition_gene] = markers(result,prodata,M,path{path_num},ordered_cell{path_num},
```

```
ans = 0.9753
Consistency of clustering: 0.97532
```

marker_gene = 4×1 cell

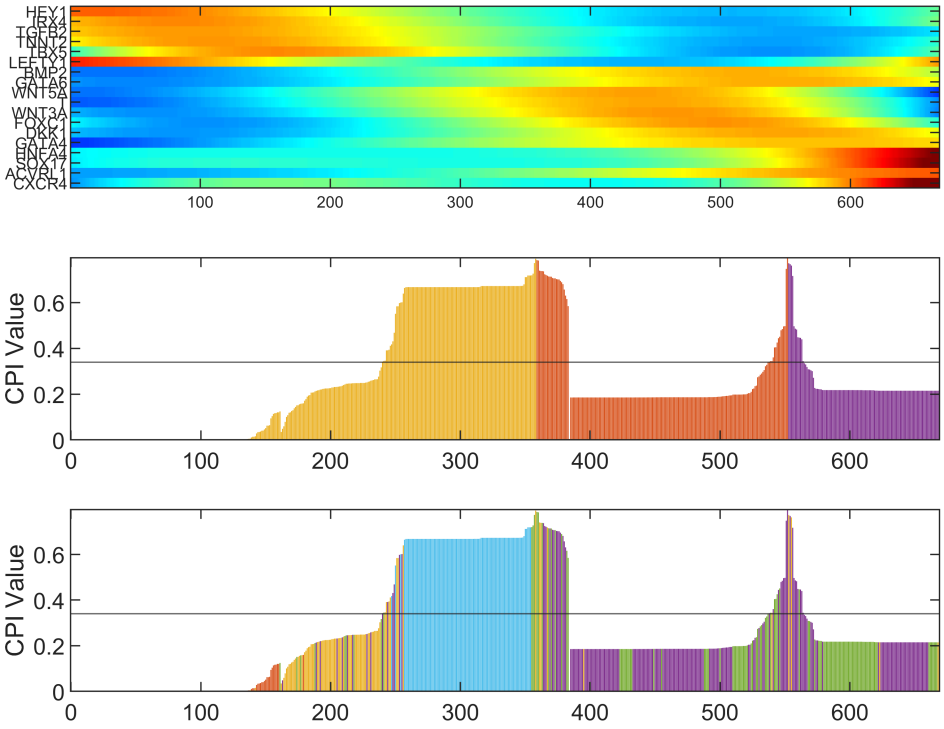
	1
1	20×1 double
2	18×1 double
3	13×1 double
4	15×1 double

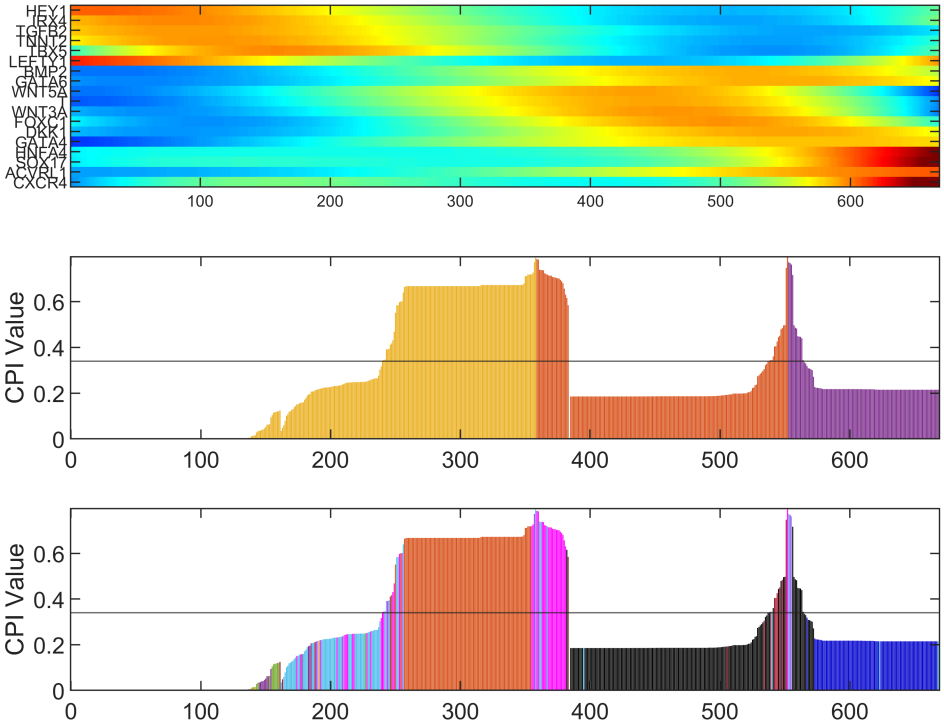
transition_gene = 2×2 cell

	1	2
1	12×1 double	49
2	79	[]

Heatmap

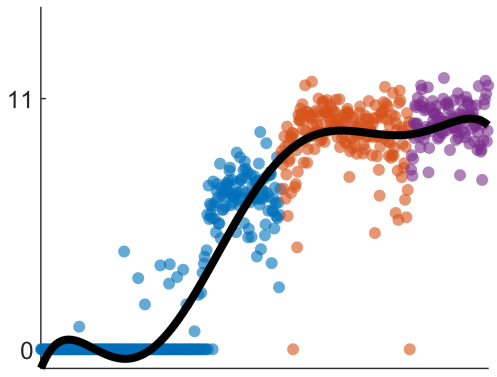
```
a = 5; b = 3; %plot top a marker genes and top 2b transition genes
figTitle='Heatmap';
[gene_plot,gene_cluster] = heatmap(prodata,marker_gene,transition_gene,path{path_num}, ...
ordered_cell{path_num},progene_name,a,b, file_dir, result, TC_cut, path_num, procell_label, fig
figTitle='Heatmap2';
[gene_plot,gene_cluster] = heatmap(prodata,marker_gene,transition_gene,path{path_num}, ...
ordered_cell{path_num},progene_name,a,b, file_dir, result, TC_cut, path_num, procell_cell_ident
```





Plot transition genes along the transiton trajectory

```
trans_gene_plot = transition_gene{1,1}(1);
tran_gene_plot(result,prodata,trans_gene_plot,path{path_num},ordered_cell{path_num},progene_nam
```




```
% output the transition genes by Holly
fid = strcat(file_dir, '\transition_gene.txt');
writecell(transition_gene, fid);

% output the transition gene names by Holly
fid = fopen(strcat(file_dir, '\transition_gene_name1.1.txt'),'wt');
fprintf(fid, '%s\n', progene_name{transition_gene{1,1}});
```

```
fclose(fid);
fid = fopen(strcat(file_dir, '\transition_gene_name1.2.txt'),'wt');
fprintf(fid, '%s\n', progene_name{transition_gene{1,2}});
fclose(fid);
fid = fopen(strcat(file_dir, '\transition_gene_name2.1.txt'),'wt');
fprintf(fid, '%s\n', progene_name{transition_gene{2,1}});
fclose(fid);


% output the cpi by Holly
fid = strcat(file_dir, '/C',num2str(start_cluster),'start_cpi_cells.txt');
writematrix(result{3}, fid);
fid = strcat(file_dir, '/C',num2str(start_cluster),'start_procell_label.txt');
writecell(procell_label, fid);
fid = strcat(file_dir, '/C',num2str(start_cluster),'start_ordered_cell.txt');
writematrix(ordered_cell{1,1}, fid);

% order the Bargaje2017_Consensus.Cluster
fid = strcat(file_dir, '/C',num2str(start_cluster),'start_procell_cell_identity.txt');
writecell(procell_cell_identity, fid);


% save working space
fid = strcat(file_dir, '\k4_4_time_var_xy.mat');
save(fid)
```