A toy example of bulk RNA-seq deconvolution through ENIGMA

Weixu Wang

Prerequisites

ENIGMA is a method for deconvoluting bulk RNA-seq matrix into cell type fractions and cell type-specific expression matrices. User could freely explore the cell heterogeneity within the bulk RNA-seq samples, study the differential expression gene, cell type-specific gene co-expression module or differentiation trajectory. In this tutorial, I applied ENIGMA on NSCLC bulk RNA-seq and corresponding FACS RNA-seq collected by Gentles et al to illustrate the main steps of ENIGMA analysis.

Construct reference(signature) matrix

The first step of ENIGMA require signature matrix that represent the unique expression pattern of each cell type. In ENIGMA, we provided the B-mode and S-mode batch effect correction method to correct and generate reference matrix. In this tutorial, we used scRNA-seq and FACS RNA-seq datsets to illustrate these two methods.

S-mode batch effect correction

We downloaded scRNA-seq dataset generated by Lambrechts, D. et al. and used one of its NSCLC patients to generate reference

```
source("/mnt/data1/weixu/HiDe/ENIGMA.R")
```

```
## Loading required package: sva
```

```
## Loading required package: mgcv
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.8-33. For overview type 'help("mgcv-package")'.
```

```
## Loading required package: genefilter
```

```
## Loading required package: BiocParallel
```

```
## Loading required package: purrr
```

```
library(Biobase)
```

```
## Loading required package: BiocGenerics
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which.max, which.min
```

```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
library(scater)
```

```
## Loading required package: SingleCellExperiment
```

```
## Loading required package: SummarizedExperiment
```

```
## Loading required package: MatrixGenerics
```

```
## Loading required package: matrixStats
```

```
##
## Attaching package: 'matrixStats'
```

```
## The following objects are masked from 'package:Biobase':
##
##     anyMissing, rowMedians
```

```
## The following objects are masked from 'package:genefilter':
##
##     rowSds, rowVars
```

```
##
## Attaching package: 'MatrixGenerics'
```

```
## The following objects are masked from 'package:matrixStats':
##
##     colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##     colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##     colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##     colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##     colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##     colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##     colWeightedMeans, colWeightedMedians, colWeightedSds,
##     colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##     rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##     rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##     rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##     rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##     rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##     rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##     rowWeightedSds, rowWeightedVars
```

```
## The following object is masked from 'package:Biobase':
##
##     rowMedians
```

```
## The following objects are masked from 'package:genefilter':
##
##     rowSds, rowVars
```

```
## Loading required package: GenomicRanges
```

```
## Loading required package: stats4
```

```
## Loading required package: S4Vectors
```

```
##
## Attaching package: 'S4Vectors'
```

```
## The following object is masked from 'package:base':
##
##     expand.grid
```

```
## Loading required package: IRanges
```

```
##
## Attaching package: 'IRanges'
```

```
## The following object is masked from 'package:purrr':
##
##     reduce
```

```
## The following object is masked from 'package:nlme':
##
##     collapse
```

```
## Loading required package: GenomeInfoDb
```

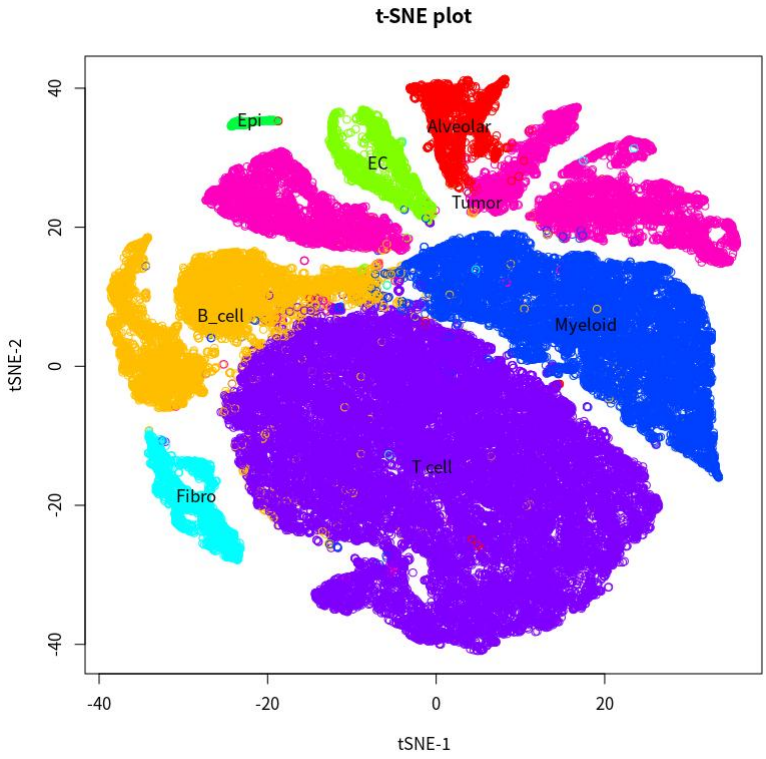
```
## Loading required package: ggplot2
```

```
library(SingleCellExperiment)
library(nnls)
library(pheatmap)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:genefilter':
##
##     area
```

```
dataNSCLC <- readRDS("/mnt/data1/weixu/HiDe/dataNSCLC.rds")
Bulk <- dataNSCLC[[5]]
Tumor <- dataNSCLC[[1]]
Immune <- dataNSCLC[[2]]
Endothelial <- dataNSCLC[[3]]
Fibroblast <- dataNSCLC[[4]]
pheno <- dataNSCLC[[6]]
names(pheno) <- colnames(Tumor)

Bulk_eset <- ExpressionSet(Bulk)
ref_sc <- readRDS("/mnt/data1/weixu/HiDe/ref.rds")
tsne <- pData(ref_sc)[,c(1,2)]

tsne_plot(tsne,pData(ref_sc)[,"main_celltype"])
```



We used the third patients to generate reference, and correct its batch effect with bulk RNA-seq dataset through S-mode correction. To make comparison with ground truth FACS RNA-seq dataset, we removed some of the cell types.

```
ref_sc_sub <- ref_sc[,ref_sc$PatientID %in% "3" == TRUE]
ref_sc_sub <- ref_sc_sub[,ref_sc_sub$CellFromTumor %in% "1"]
#ref_sc_sub$main_celltype[ref_sc_sub$main_celltype %in% c("T cell","B_cell","Myeloid")] <- "Imm
une"
ref_sc_sub <- ref_sc_sub[,ref_sc_sub$main_celltype %in% c("Alveolar","Epi") == FALSE]

## Running S-mode correction
tmp = remove_batch_effect(Bulk_eset,ref_sc_sub,"main_celltype",n_pseudo_bulk=100)
```

```
## Sun Jul  4 22:31:27 2021 generating pseudo bulk...
## Sun Jul  4 22:31:41 2021 do ComBat...
## Found 898 genes with uniform expression within a single batch (all zeros); these will not be
adjusted for batch.
```

```
## Found2batches
```

```
## Adjusting for0covariate(s) or covariate level(s)
```

```
## Standardizing Data across genes
```

```
## Fitting L/S model and finding priors
```

```
## Finding parametric adjustments
```

```
## Adjusting the Data
```

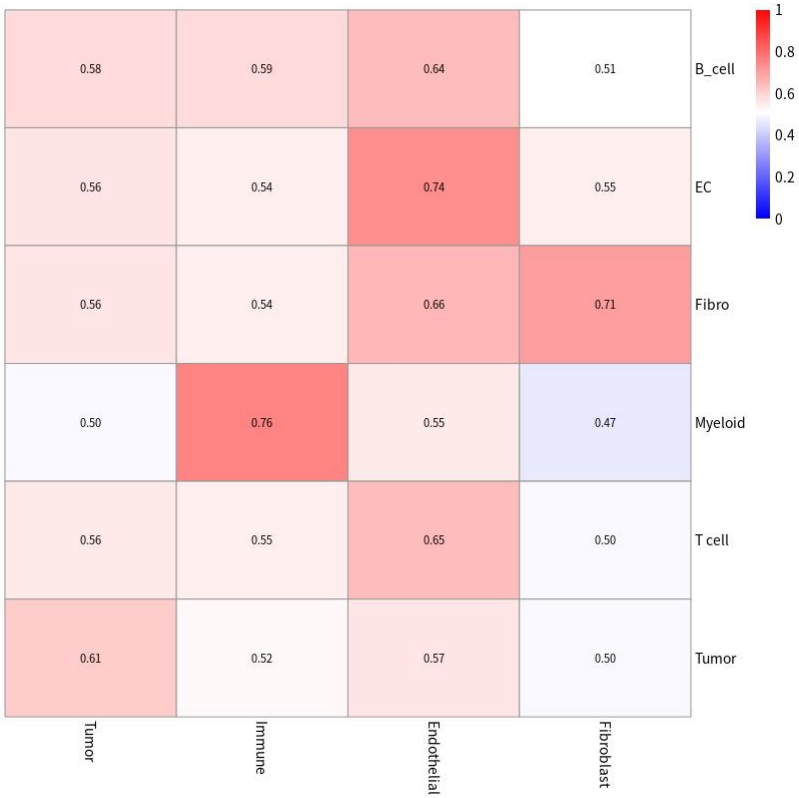
```
## Sun Jul  4 22:31:45 2021 restore reference...
```

```
head(tmp$main_celltype)
```

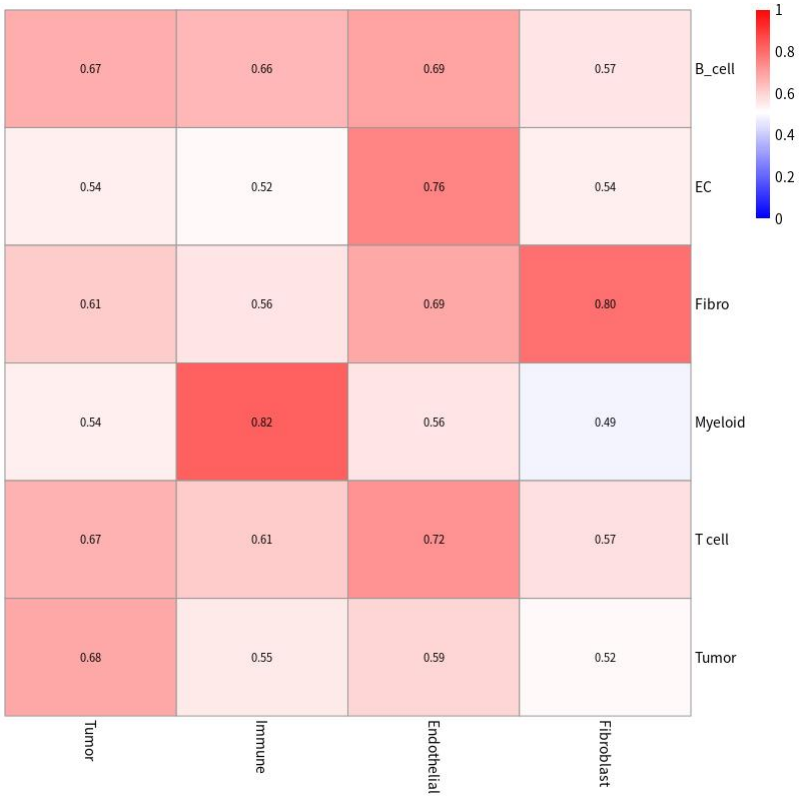
```
##              B_cell          EC        Fibro     Myeloid       T cell    Tumor
## A1BG     69.75826354    3.088783 8.245254e+01  91.5718481 1.028807e+02   0.0000
## A1CF      0.00000000    0.000000 0.000000e+00   0.0000000 0.000000e+00   0.0000
## A2M     207.20450782 2966.978441 3.509105e+03 319.3863088 3.289811e+01 139.2880
## A2ML1     0.00000000    0.000000 0.000000e+00   0.0000000 0.000000e+00   0.0000
## A4GALT    1.72185383   83.113559 1.511588e+02  11.7966276 6.246375e+00  63.9042
## A4GNT     0.02734789    0.000000 8.661414e-02   0.4957382 1.716769e-03   0.0000
```

To validate that after the S-mode correction, the correlation between reference and ground truth CSE expression is increased, we made comparison between reference matrix generate from S-mode correction, and reference matrix generate through directly average expression.

```
# generate ground truth reference
gt <- NULL
for(K in 1:4){
 gt <- cbind(gt, rowMeans(dataNSCLC[[K]]))
}
colnames(gt) <- c("Tumor","Immune","Endothelial","Fibroblast")

ref_profile <- NULL
for(K in names(table(ref_sc_sub$main_celltype))){
 ref_profile <- cbind(ref_profile, rowMeans(exprs(ref_sc_sub)[,ref_sc_sub$main_celltype %in%
K]))
}
colnames(ref_profile) <- names(table(ref_sc_sub$main_celltype))

bk <- c(seq(0,1,by=0.01))
pheatmap(cor(ref_profile[rownames(tmp$main_celltype),],gt[rownames(tmp$main_celltype),]),
         scale = "none",
         color = c(colorRampPalette(colors = c("blue","white"))(length(bk)/2),colorRampPalette
(colors = c("white","red"))(length(bk)/2)),
         breaks=bk,cluster_row=FALSE,cluster_col=FALSE,display_numbers = TRUE,number_color = "b
lack")
```




```
pheatmap(cor(tmp$main_celltype[rownames(tmp$main_celltype),],gt[rownames(tmp$main_celltype),]),
         scale = "none",
         color = c(colorRampPalette(colors = c("blue","white"))(length(bk)/2),colorRampPalette
(colors = c("white","red"))(length(bk)/2)),
         breaks=bk,cluster_row=FALSE,cluster_col=FALSE,display_numbers = TRUE,number_color = "b
lack")
```



User also could use tmp file generated already for saving time

```
tmp <- readRDS("/mnt/data1/weixu/HiDe/tmp.rds")
```

We could observe the increased correlation after the S-mode correlation.

Running B-mode correction

We used FACS RNA-seq to illustrate the B-mode batch effect correction, regarding the expression profile of T3 patients as the the reference, we noted that the FACS RNA-seq is not necessarily to be generated from the same cohort with the mixture(bulk RNA-seq) but could from the independent study.

```
B_ref <- cbind(Tumor[,"T3"],Immune[,"T3"],Endothelial[,"T3"],Fibroblast[,"T3"])
colnames(B_ref) <- c("Tumor","Immune","Endothelial","Fibroblast")

##To make comparison, we select the same genes with S-mode based reference
B_ref <- B_ref[rownames(tmp$main_celltype),]

##Running B-mode batch effect correction
frac <- get_proportion(exprs(Bulk_eset),B_ref)
```

```
## Sun Jul  4 22:31:51 2021 Calculating cell type proportion of bulk samples...
```

```
tmp_B <- B_mode_batch_effect_remove(exprs(Bulk_eset)[rownames(B_ref),],B_ref,frac$theta)
```

```
## Run B-mode to correct batch effect...
##  do Combat...Found 678 genes with uniform expression within a single batch (all zeros); thes
e will not be adjusted for batch.
```

```
## Found2batches
```

```
## Adjusting for0covariate(s) or covariate level(s)
```

```
## Standardizing Data across genes
```

```
## Fitting L/S model and finding priors
```

```
## Finding parametric adjustments
```

```
## Adjusting the Data
```

```
##
##  Done
```

Deconvolute bulk RNA-seq profile through ENIGMA

ENIGMA provides two type of regularized matrix completion methods to deconvolute bulk RNA-seq profile, L2-max norm and trace norm regularization. Before running the algorithm, we need to process the gene expression profile, try to stablize each gene variance, make each gene expression distribution closer to the

gaussian distribution. In our preprocessing_test tutorial, we showed that using sqrt() to transform gene expression profile is more suitable than log() transformation, therefore, we applied sqrt() to transform gene expression matrix for processing input matrix.

L2-max norm

```
frac_s_mode <- get_proportion(exprs(Bulk_eset),tmp$main_celltype)
```

```
## Sun Jul  4 22:31:57 2021 Calculating cell type proportion of bulk samples...
```

```
system.time({ L2maxNorm <- cell_deconvolve(X=as.matrix(sqrt(Bulk[rownames(tmp$main_celltyp
e),])),
                            theta=frac_s_mode$theta,
                            R=as.matrix(sqrt(tmp$main_celltype)),
                            inner_epilson=0.001,
                            alpha=0.8,
                            miu=10000,tao_k=0.01,max.iter=1,max.iter.exp=1000,verbose=TRUE)})
```

```
## Sun Jul  4 22:32:04 2021 Optimizing cell type specific expression profile...
##    Ratio ranges from: 4308564.013017 - 4373999.028274
##    Ratio ranges from: 97001.835711 - 133338.990223
##    Ratio ranges from: 3506.936762 - 4882.304495
##    Ratio ranges from: 144.316742 - 208.485093
##    Ratio ranges from: 7.783688 - 19.143320
##    Ratio ranges from: 0.373249 - 3.606474
##    Ratio ranges from: 0.021416 - 0.857877
##    Ratio ranges from: 0.001477 - 0.213690
##    Ratio ranges from: 0.000118 - 0.054010
##    Ratio ranges from: 0.000010 - 0.013748
##    Ratio ranges from: 0.000001 - 0.003515
##    Ratio ranges from: 0.000000 - 0.000901
## Optimizing cell type proportions...
```

```
##    user  system elapsed
##   6.288   0.100   6.387
```

Important parameters are as follows:

- *X*: The inputted bulk RNA-seq matrix
- *theta*: The inputted cell type fraction matrix
- *R*: The inputted cell type signature(reference) matrix
- *inner_epilson*: This parameter is used to determine the stop condition in CSE updating. Default: 0.001
- *outer_epilson*: This parameter is used to determine the stop condition in whole parameter updating. Default: 0.001
- *alpha*: ENIGMA is a multi-objective optimization problem involve two object function, the distance function between observed bulk RNA-seq and reconstitute RNA-seq generated by weighted combination of CSE, and the distance function beween average CSE expression and cell type reference matrix. The alpha is used to determine weights of these two objects. If the alpha gets larger, the optimization attach greater importance on the the first object.

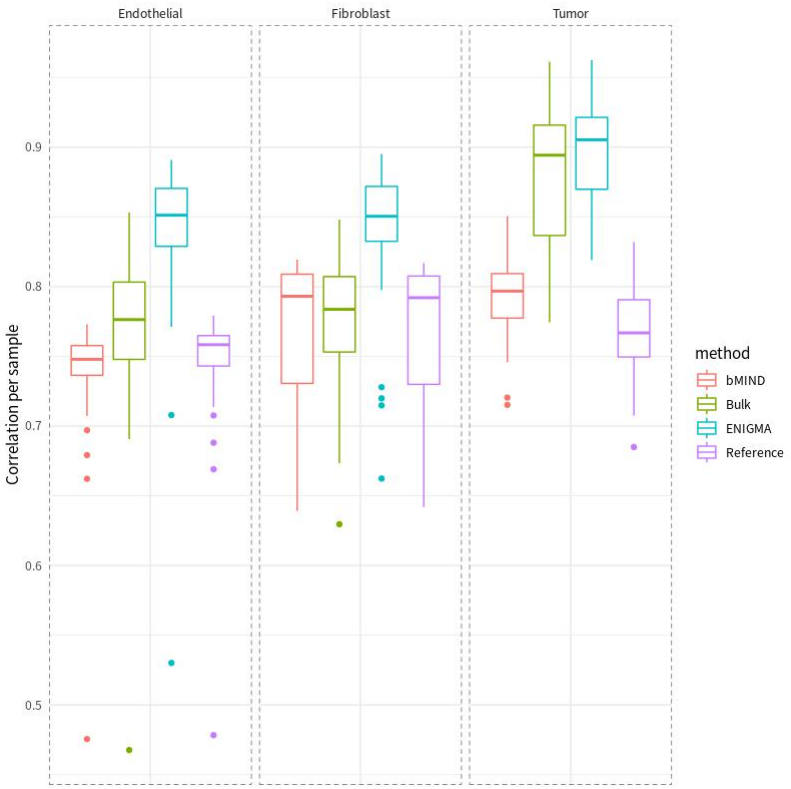
- *tao_k*: The step size of each round of gradient decent
- *max.iter*: ENIGMA could updated the estimation of CSE and cell type fractions through Expectation and Maximization (EM) fashion. In E-step, ENIGMA estimates the cell type fractions through robust linear regression, in M-step, ENIGMA maximize the object function through optimizing CSE with fixed cell type fractions matrix. The max.iter determines the number iterations of EM fashion optimization.
- *max.iter.exp*: the maximum number of iterations in M-step.

To make comparison, we performed deconvolution through bMIND

```
#system.time({deconv = bMIND2(as.matrix(sqrt(Bulk[rownames(tmp$main_celltype),])), frac= frac_s
_mode$theta, profile = sqrt(tmp$main_celltype), noRE = F,ncore=1)})
#The bMIND requires a long time to calculate CSE, you could get the bMIND estimated results dir
ectly as follow
deconv <- readRDS("/mnt/data1/weixu/HiDe/deconv.rds")
```

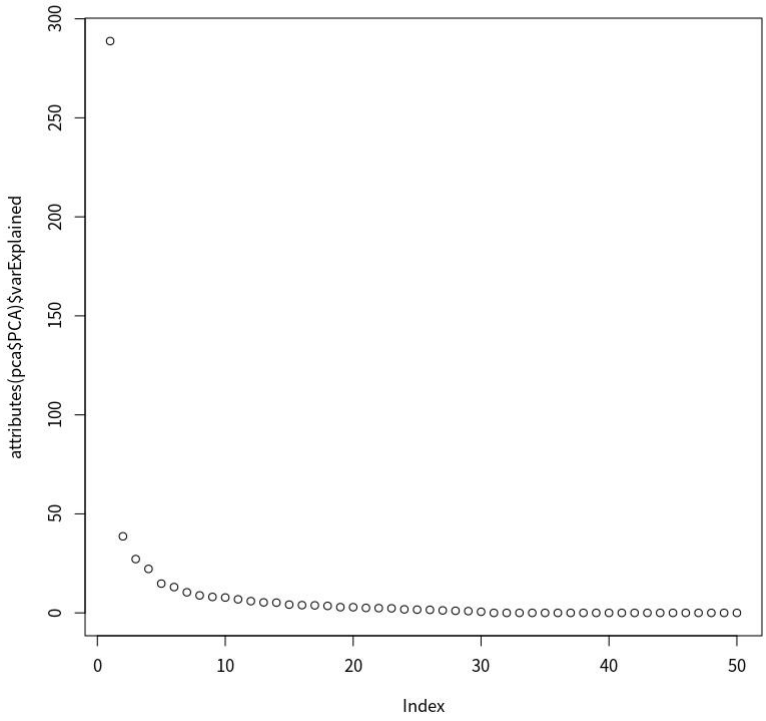
We benchmarked our method with bMIND through comparing their correlation with ground truth cell type-specific expression profile

```
tumor.enigma <- NULL
tumor.bmind <- NULL
tumor.bulk <- NULL
tumor.ref <- NULL
for(i in 1:ncol(L2maxNorm$expr_array[,,6])){
  tumor.enigma <- c(tumor.enigma,cor(L2maxNorm$expr_array[,i,6],Tumor[rownames(tmp$main_celltyp
e),i],method="sp"))
  tumor.bmind <- c(tumor.bmind,cor(deconv$A[,6,i],Tumor[rownames(tmp$main_celltype),i],method
="sp"))
  tumor.bulk <- c(tumor.bulk,cor(Bulk[rownames(tmp$main_celltype),i],Tumor[rownames(tmp$main_ce
lltype),i],method="sp"))
  tumor.ref <- c(tumor.ref,cor(tmp$main_celltype[,6],Tumor[rownames(tmp$main_celltype),i],metho
d="sp"))
}
tumor_vec <- c(tumor.enigma,tumor.bmind,tumor.bulk,tumor.ref)

fibro.enigma <- NULL
fibro.bmind <- NULL
fibro.bulk <- NULL
fibro.ref <- NULL
for(i in 1:ncol(L2maxNorm$expr_array[,,3])){
  fibro.enigma <- c(fibro.enigma,cor(L2maxNorm$expr_array[,i,3],Fibroblast[rownames(tmp$main_ce
lltype),i],method="sp"))
  fibro.bmind <- c(fibro.bmind,cor(deconv$A[,3,i],Fibroblast[rownames(tmp$main_celltype),i],met
hod="sp"))
  fibro.bulk <- c(fibro.bulk,cor(Bulk[rownames(tmp$main_celltype),i],Fibroblast[rownames(tmp$ma
in_celltype),i],method="sp"))
  fibro.ref <- c(fibro.ref,cor(tmp$main_celltype[,3],Fibroblast[rownames(tmp$main_celltype),i],
method="sp"))
}
fibro_vec <- c(fibro.enigma,fibro.bmind,fibro.bulk,fibro.ref)

endo.enigma <- NULL
endo.bmind <- NULL
endo.bulk <- NULL
endo.ref <- NULL
for(i in 1:ncol(L2maxNorm$expr_array[,,2])){
  endo.enigma <- c(endo.enigma,cor(L2maxNorm$expr_array[,i,2],Endothelial[rownames(tmp$main_cel
ltype),i],method="sp"))
  endo.bmind <- c(endo.bmind,cor(deconv$A[,2,i],Endothelial[rownames(tmp$main_celltype),i],meth
od="sp"))
  endo.bulk <- c(endo.bulk,cor(Bulk[rownames(tmp$main_celltype),i],Endothelial[rownames(tmp$mai
n_celltype),i],method="sp"))
  endo.ref <- c(endo.ref,cor(tmp$main_celltype[,2],Endothelial[rownames(tmp$main_celltype),i],m
ethod="sp"))
}
endo_vec <- c(endo.enigma,endo.bmind,endo.bulk,endo.ref)

dat <- data.frame(cor=c(tumor_vec,fibro_vec,endo_vec),celltype=c(rep("Tumor",24*4),rep("Fibrobl
ast",24*4),rep("Endothelial",24*4)),
                  method=rep(c(rep("ENIGMA",24),rep("bMIND",24),rep("Bulk",24),rep("Reference",
24)),3))
ggplot(dat, aes(x=celltype, y=cor, color=method)) +
    geom_boxplot(position=position_dodge(1))+theme_minimal()+labs(y="Correlation per sample")+
facet_grid(~celltype, scales = "free_x") +
    theme(axis.title.x = element_blank(), axis.text.x = element_blank()) +theme(panel.border =
element_rect(size = 0.3, linetype = "dashed", fill = NA))
```

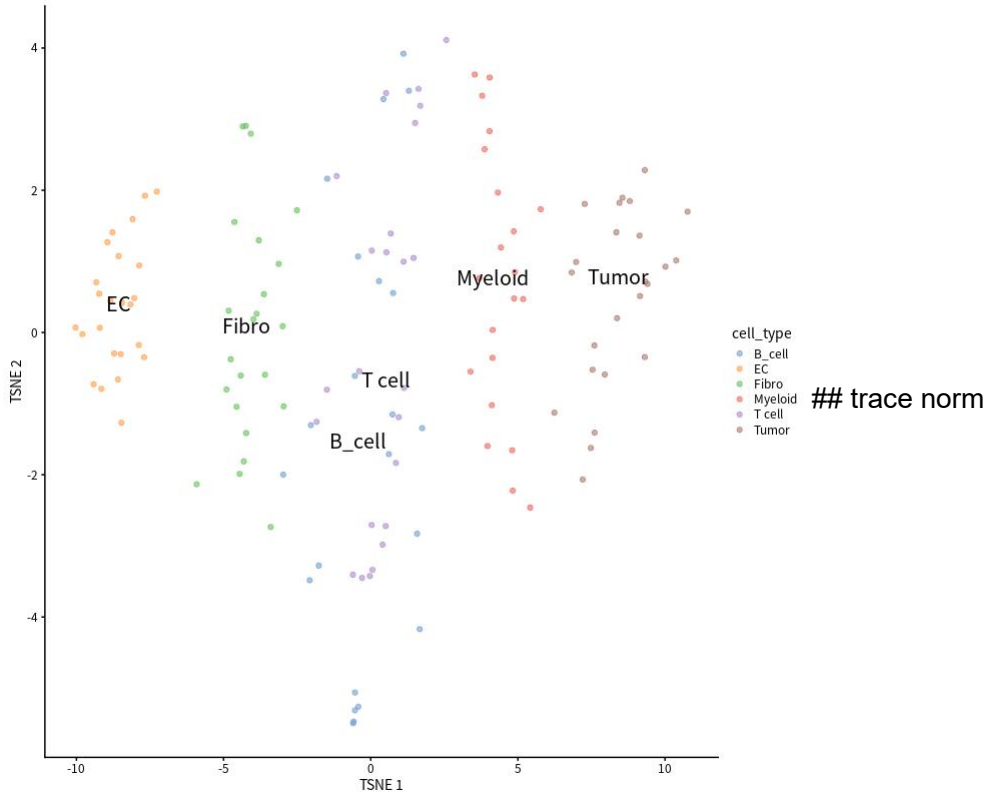


We could notice that CSE produced by ENIGMA show generally higher correlation with each sample than bMIND, bulk or reference matrix. We also could visualize CSE through tsne plot

```
nsclc <- celltype <- NULL
for(i in 1:ncol(tmp$main_celltype)){
 nsclc <- cbind(nsclc,L2maxNorm$expr_array[,frac_s_mode$theta[,i]!=0,i])
 celltype <- c(celltype,rep(colnames(tmp$main_celltype)[i],sum(frac_s_mode$theta[,i]!=0)))
}

sce_nsclc <- SingleCellExperiment(assays = list(logcounts = nsclc))
sce_nsclc$cell_type <- celltype
sce_nsclc <- runPCA(sce_nsclc,scale=TRUE)
pca <- reducedDims(sce_nsclc)
plot(attributes(pca$PCA)$varExplained)
```




```
sce_nsclc <- runTSNE(sce_nsclc)
plotTSNE(sce_nsclc, colour_by="cell_type",text_by="cell_type")
```



trace norm

Finally, we run the trace norm based matrix completion to deconvolute bulk RNA-seq matrix. Compare with L2-max norm, trace norm has more parameters need to tune.

```
frac_s_mode <- get_proportion(exprs(Bulk_eset),tmp$main_celltype)
```

```
## Sun Jul  4 22:32:14 2021 Calculating cell type proportion of bulk samples...
```

```
system.time({ traceNorm <- cell_deconvolve_trace(0 = as.matrix(sqrt(Bulk[rownames(tmp$main_cell
type),])),
                                                 theta=frac_s_mode$theta,
                                                 R=as.matrix(sqrt(tmp$main_celltype)),
                                                 epsilon=0.0001,
                                                 alpha=0.8,beta=200,gamma = 1,
                                                 verbose=TRUE,max.iter = 300)})
```

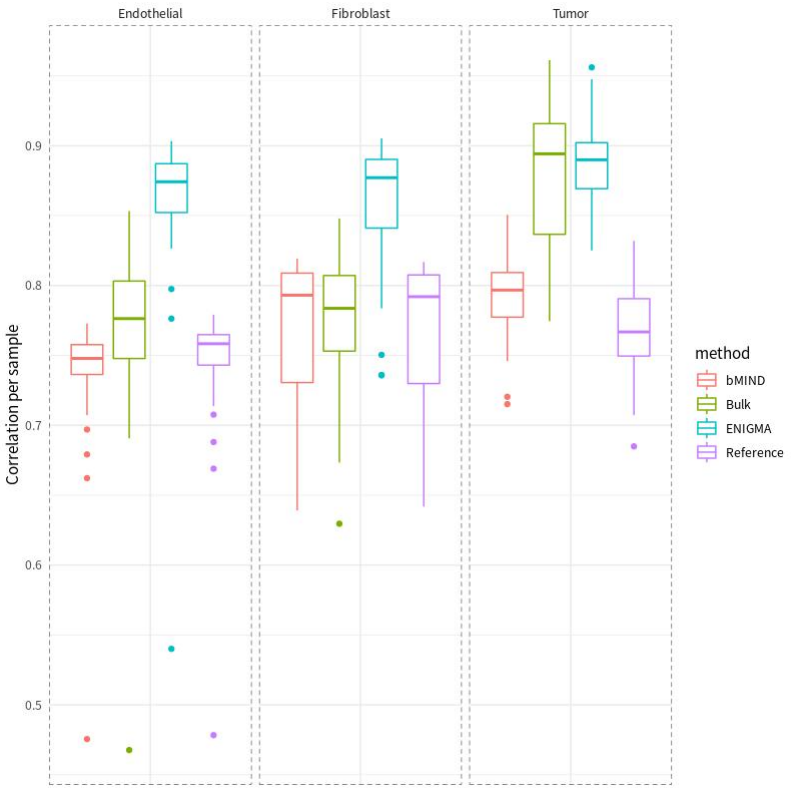
```
##    Ratio ranges from: 0.805922 - 0.965892
##    Loss: part1=0.000000 , part2=11344695.974730 , part3=7237232.789575
##    Ratio ranges from: 73.639006 - 2014.153030
##    Loss: part1=2158073.166194 , part2=20961036.330235 , part3=339980.516676
##    Ratio ranges from: 0.000569 - 0.007547
##    Loss: part1=99274.991273 , part2=10276598.211094 , part3=6402202.409922
##    Ratio ranges from: 0.000470 - 0.004925
##    Loss: part1=88798.471068 , part2=10376107.785325 , part3=5784401.020347
##    Ratio ranges from: 0.000425 - 0.003179
##    Loss: part1=88424.761368 , part2=10445787.673563 , part3=5333910.212645
##    Ratio ranges from: 0.000402 - 0.001748
##    Loss: part1=89860.480840 , part2=10490442.334491 , part3=4957845.521569
##    Ratio ranges from: 0.000366 - 0.001281
##    Loss: part1=95224.183870 , part2=10515724.446478 , part3=4677161.286983
##    Ratio ranges from: 0.000347 - 0.000833
##    Loss: part1=102339.657077 , part2=10525984.509500 , part3=4479239.103538
##    Ratio ranges from: 0.000310 - 0.000589
##    Loss: part1=108119.857471 , part2=10523902.720239 , part3=4307252.349183
##    Ratio ranges from: 0.000287 - 0.000422
##    Loss: part1=113166.517626 , part2=10513120.955735 , part3=4173022.562775
##    Ratio ranges from: 0.000246 - 0.000328
##    Loss: part1=118173.982547 , part2=10496072.994052 , part3=4081639.090133
##    Ratio ranges from: 0.000202 - 0.000254
##    Loss: part1=120405.274989 , part2=10474496.090349 , part3=3996555.384193
##    Ratio ranges from: 0.000183 - 0.000232
##    Loss: part1=122720.112931 , part2=10449815.086683 , part3=3929420.041575
##    Ratio ranges from: 0.000151 - 0.000199
##    Loss: part1=124657.545375 , part2=10422925.388488 , part3=3874374.539958
##    Ratio ranges from: 0.000124 - 0.000191
##    Loss: part1=126147.023087 , part2=10394675.989484 , part3=3831053.565027
##    Ratio ranges from: 0.000106 - 0.000163
##    Loss: part1=127114.479917 , part2=10365686.840152 , part3=3790158.857244
##    Ratio ranges from: 0.000088 - 0.000154
##    Loss: part1=127929.973548 , part2=10336399.667283 , part3=3757595.640819
##    Ratio ranges from: 0.000079 - 0.000148
##    Loss: part1=128621.736499 , part2=10307086.201188 , part3=3730339.843975
##    Ratio ranges from: 0.000072 - 0.000136
##    Loss: part1=129055.993452 , part2=10278010.564341 , part3=3706372.166408
##    Ratio ranges from: 0.000067 - 0.000126
##    Loss: part1=129400.775658 , part2=10249407.494744 , part3=3684279.445880
##    Ratio ranges from: 0.000060 - 0.000121
##    Loss: part1=129638.516745 , part2=10221422.597547 , part3=3665098.523454
##    Ratio ranges from: 0.000055 - 0.000114
##    Loss: part1=129697.054574 , part2=10194131.228764 , part3=3648782.395093
##    Ratio ranges from: 0.000051 - 0.000107
##    Loss: part1=129756.297955 , part2=10167559.381403 , part3=3634070.304197
##    Ratio ranges from: 0.000049 - 0.000103
##    Loss: part1=129651.737867 , part2=10141755.387097 , part3=3620956.200269
##    Ratio ranges from: 0.000046 - 0.000100
##    Loss: part1=129527.912816 , part2=10116763.810510 , part3=3609010.039144
## Converge in 25 steps
```

```
##    user  system elapsed
##  32.300  59.144  15.469
```

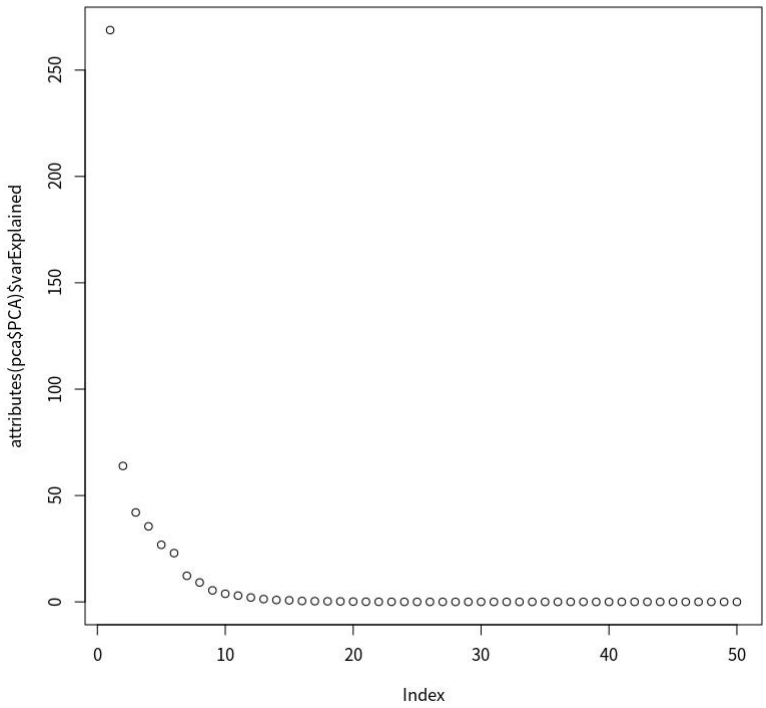
Important parameters are as follows:

- *O*: The inputted bulk RNA-seq matrix
- *beta*: This parameter is used to control the latent dimension of each CSE, if this parameter gets larger, than the latent dimension of each CSE is smaller (lower trace norm value), which means that each sample is more similar with each others. The user need to tune this parameter based on the range of the singular value of the bulk RNA-seq matrix. default setting: 100
- *epsilon*: In trace norm based ENIGMA, the epsilon is not necessarily choose a extremly small value, the number of iteration would influence the latent dimensions of CSE, as each step is performing singular value thresholding. default setting: 0.0001

```
tumor.enigma <- NULL
tumor.bmind <- NULL
tumor.bulk <- NULL
tumor.ref <- NULL
for(i in 1:ncol(traceNorm[,,6])){
  tumor.enigma <- c(tumor.enigma,cor(traceNorm[,i,6],Tumor[rownames(tmp$main_celltype),i],metho
d="sp"))
  tumor.bmind <- c(tumor.bmind,cor(deconv$A[,6,i],Tumor[rownames(tmp$main_celltype),i],method
="sp"))
  tumor.bulk <- c(tumor.bulk,cor(Bulk[rownames(tmp$main_celltype),i],Tumor[rownames(tmp$main_ce
lltype),i],method="sp"))
  tumor.ref <- c(tumor.ref,cor(tmp$main_celltype[,6],Tumor[rownames(tmp$main_celltype),i],metho
d="sp"))
}
tumor_vec <- c(tumor.enigma,tumor.bmind,tumor.bulk,tumor.ref)

fibro.enigma <- NULL
fibro.bmind <- NULL
fibro.bulk <- NULL
fibro.ref <- NULL
for(i in 1:ncol(traceNorm[,,3])){
  fibro.enigma <- c(fibro.enigma,cor(traceNorm[,i,3],Fibroblast[rownames(tmp$main_celltype),i],
method="sp"))
  fibro.bmind <- c(fibro.bmind,cor(deconv$A[,3,i],Fibroblast[rownames(tmp$main_celltype),i],met
hod="sp"))
  fibro.bulk <- c(fibro.bulk,cor(Bulk[rownames(tmp$main_celltype),i],Fibroblast[rownames(tmp$ma
in_celltype),i],method="sp"))
  fibro.ref <- c(fibro.ref,cor(tmp$main_celltype[,3],Fibroblast[rownames(tmp$main_celltype),i],
method="sp"))
}
fibro_vec <- c(fibro.enigma,fibro.bmind,fibro.bulk,fibro.ref)

endo.enigma <- NULL
endo.bmind <- NULL
endo.bulk <- NULL
endo.ref <- NULL
for(i in 1:ncol(traceNorm[,,2])){
  endo.enigma <- c(endo.enigma,cor(traceNorm[,i,2],Endothelial[rownames(tmp$main_celltype),i],m
ethod="sp"))
  endo.bmind <- c(endo.bmind,cor(deconv$A[,2,i],Endothelial[rownames(tmp$main_celltype),i],meth
od="sp"))
  endo.bulk <- c(endo.bulk,cor(Bulk[rownames(tmp$main_celltype),i],Endothelial[rownames(tmp$mai
n_celltype),i],method="sp"))
  endo.ref <- c(endo.ref,cor(tmp$main_celltype[,2],Endothelial[rownames(tmp$main_celltype),i],m
ethod="sp"))
}
endo_vec <- c(endo.enigma,endo.bmind,endo.bulk,endo.ref)

dat <- data.frame(cor=c(tumor_vec,fibro_vec,endo_vec),celltype=c(rep("Tumor",24*4),rep("Fibrobl
ast",24*4),rep("Endothelial",24*4)),
                  method=rep(c(rep("ENIGMA",24),rep("bMIND",24),rep("Bulk",24),rep("Reference",
24)),3))
ggplot(dat, aes(x=celltype, y=cor, color=method)) +
    geom_boxplot(position=position_dodge(1))+theme_minimal()+labs(y="Correlation per sample")+
facet_grid(~celltype, scales = "free_x") +
    theme(axis.title.x = element_blank(), axis.text.x = element_blank()) +theme(panel.border =
element_rect(size = 0.3, linetype = "dashed", fill = NA))
```




```
nsclc <- celltype <- NULL
for(i in 1:ncol(tmp$main_celltype)){
 nsclc <- cbind(nsclc,traceNorm[,frac_s_mode$theta[,i]!=0,i])
 celltype <- c(celltype,rep(colnames(tmp$main_celltype)[i],sum(frac_s_mode$theta[,i]!=0)))
}

sce_nsclc <- SingleCellExperiment(assays = list(logcounts = nsclc))
sce_nsclc$cell_type <- celltype
sce_nsclc <- runPCA(sce_nsclc,scale=TRUE)
pca <- reducedDims(sce_nsclc)
plot(attributes(pca$PCA)$varExplained)
```




```
sce_nsclc <- runTSNE(sce_nsclc)
plotTSNE(sce_nsclc, colour_by="cell_type",text_by="cell_type")
```