TED Package

TED packages contains three major functions, 1) the “run.Ted” function which implements a fully Bayesian inference of tumor microenvironment composition and gene expression, 2) the “learn.embedding.Kcls” and 3) “learn.embedding.withPhiTum” functions which uses Expectation-maximization (EM) to approximate the tumor expression using a linear combination of tumor pathways while conditional on the inferred expression and fraction of non-tumor cells estimated by the deconvolution module. The “learn.embedding.Kcls” function initializes the embedding by running hierarchical clustering over the tumor expression inferred by run.Ted, while “learn.embedding.withPhiTum” initializes using the embedding provided by the user.

In this example we will be deconvolving 169 TCGA-GBM bulk RNA-seq samples using the scRNA-seq dataset from 8 high grade glioma patients as the reference. We will also demonstrate the embedding learning of tumor pathways from TCGA-GBM samples. The "tcga.gbm.example.rdata" contains the pre-prepared normalized scRNA-seq gene expression profile (GEP) matrix and bulk sample matrix over protein-coding genes. In practice, users may also use unnormalized GEP or raw count matrix of individual cells. Genes need not to be aligned between the reference matrix and the bulk matrix. run.Ted will automatically collapse, align the genes on the common subset between reference and bulk, and normalized the scRNA-seq reference.

1) Deconvolve bulk RNA-seq and infer tumor expression

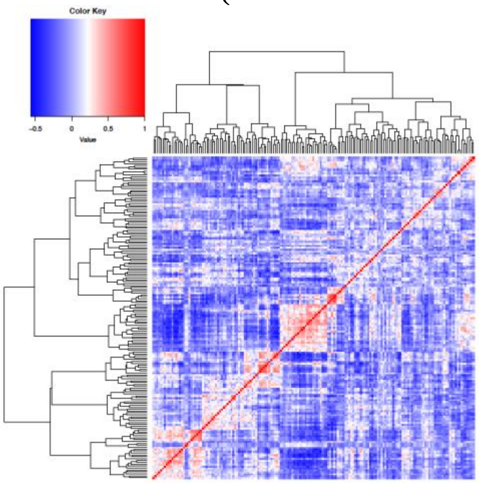
```
> #load TED package
> library(TED)

> #load pre-prepared reference scRNA-seq gene expression matrix and the bulk sample
matrix
> load(system.file("extdata", "tcga.gbm.example.rdata", package="TED"))

# Manually remove ribosomal and mitochondrial genes from the reference matrix, may also
remove genes on the sex chromosomes if samples are collected on different genders.

> ref.norm.filtered <- cleanup.genes(ref.dat= ref.norm,
                                      species="hs",
                                      gene.type=c("RB","chrM","chrX","chrY"),
                                      exp.cells=0) #set exp.cells=0 as this is a normalized
                                      matrix

#assign cell types and subtype labels
> cell.subtype.labels <- rownames(ref.norm)
> cell.type.labels <- cell.subtype.labels
> cell.type.labels[grepl("tumor", cell.type.labels)] <- "tumor"
```

```
 #run BayesPrism
    > tcga.ted <-  run.Ted  (ref.dat = ref.norm.filtered,
                              X= tcga.tumor.pc.NOchrY,
                              cell.type.labels=cell.type.labels,
                              cell.subtype.labels= cell.subtype.labels,
                              tum.key="tumor",
                              input.type="GEP",
                              n.cores=30,
                              outlier.cut=0.05,
                              pdf.name="tcga.tumor")
```

```
#console output:
[1] "processing input..."
[1] "run first sampling"
[1] "inferred cell percentage"
             Tumor myeloid pericyte endothelial tcell oligo
        Min.    0.276   0.003    0.000       0.002 0.000 0.000
        1st Qu. 0.766   0.038    0.007       0.020 0.000 0.006
        Median  0.838   0.076    0.015       0.030 0.000 0.021
        Mean    0.813   0.086    0.033       0.032 0.001 0.035
        3rd Qu. 0.891   0.113    0.032       0.040 0.000 0.043
        Max.    0.974   0.543    0.494       0.100 0.043 0.287
[1] "correct batch effect"
[1] "vst transformation is feasible"
converting counts to integer mode
[1] "run final sampling"
             Tumor myeloid pericyte endothelial tcell oligo
        Min.    0.364   0.000    0.000       0.003 0.000 0.000
        1st Qu. 0.810   0.034    0.004       0.016 0.000 0.002
        Median  0.863   0.069    0.009       0.026 0.000 0.010
        Mean    0.845   0.077    0.023       0.028 0.001 0.026
        3rd Qu. 0.912   0.100    0.023       0.036 0.000 0.032
        Max.    0.981   0.494    0.381       0.101 0.018 0.277
```

```
#to extract output from tcga.ted
> tcga.ted$res$first.gibbs.res$gibbs.theta  #Initial estimates of fraction for all cell subtypes in each bulk sample.
> tcga.ted$res$first.gibbs.res$Znkg #Initial estimates of the mean of posterior read count for each cell subtypes in each bulk sample.
> tcga.ted$res$first.gibbs.res$ theta.merged  #Initial estimates of fraction for all cell types in each bulk sample.
> tcga.ted$res$first.gibbs.res$ Znkg.merged  # the mean of posterior reads in each cell type of
each bulk sample
> tcga.ted$res$ Z.tum.first.gibbs    # the mean count of tumor expression in each bulk sample
> tcga.ted$res$ Zkg.tum.norm # the depth-normalized count of tumor expression in each bulk
sample (the zero count genes adjusted to the same small value for each sample)
> tcga.ted$res$ Zkg.tum.vst # the variance stabilizing transformed count of tumor expression in
each bulk sample (by the vst function in DESeq2)
> tcga.ted$res$phi.env #the batch corrected non-malignant cell expression
> tcga.ted$res$cor.mat #the correlation heatmap of  tumor expressions across bulk samples
> tcga.ted$res$ final.gibbs.theta  # theta: the updated estimates of cell type fraction
```

#The correlation (between 169 TCGA-GBM samples) heatmap generated by run.Ted:



2) Learning embeddings of tumor expression at K=4, with pathways initialized by hierarchical clustering

```
>  tcga.ebd.res.k4 <- learn.embedding.Kcls  (ted.res = tcga.ted,
                                             K.vec = 4,
                                             EM.maxit=50,
                                             n.cores =50)
```

#to extract output from tcga.ebd.res.k4

tcga.ebd.res.k4$ theta.all # The fractions associated with tumor bases (first K.tum columns) and stromal cells.

tcga.ebd.res.k4$ opt.phi.hat.tum # the inferred expression profile of each tumor pathways, referred to as eta in the TED paper

tcga.ebd.res.k4$ log.posterior #log posterior of each EM cycle, if compute.posterior=T (default is not computed)