Tutorial for R package scDeconv

Yu Liu

2022-09-03

## Introduction

Many DNA methylation (DNAm) data are from tissues composed of various cell types; hence, cell deconvolution methods are needed to infer their cell compositions accurately. However, a bottleneck for DNAm data is the lack of cell-type-specific DNAm references. On the other hand, scRNA-seq data are being accumulated rapidly with various cell type transcriptomic signatures characterized, and also, many paired bulk RNA-DNAm data are publicly available currently. Hence, I developed the R package *scDeconv* to use these resources to solve the reference deficiency problem of DNAm data and deconvolve them from scRNA-seq data in a trans-omics manner.

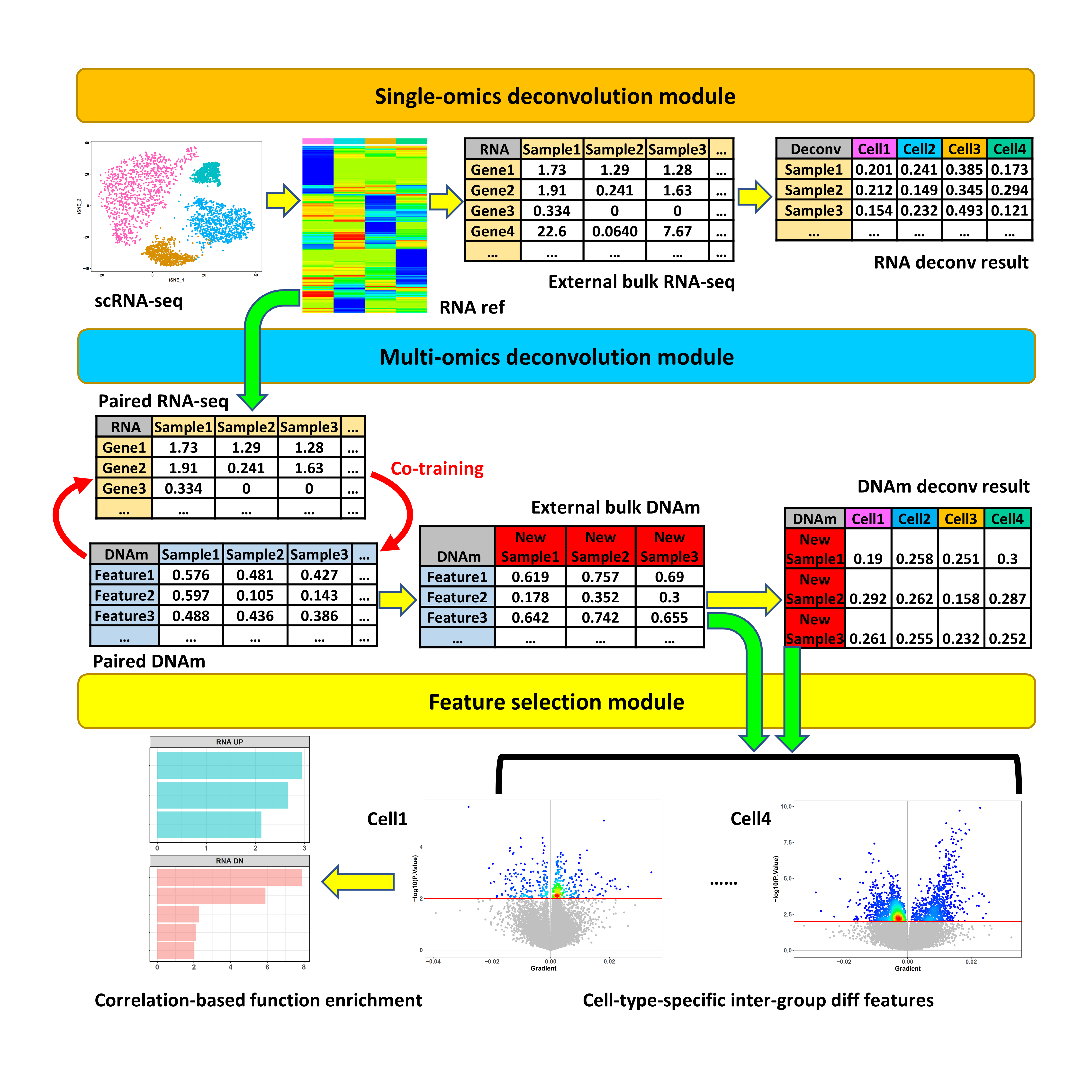
It assumes that paired samples have similar cell compositions. So the cell content information deconvolved from the scRNA-seq and paired RNA data can be transferred to the paired DNAm samples. Then an ensemble model is trained to fit these cell contents with DNAm features and adjust the paired RNA deconvolution in a co-training manner. Finally, the model can be used on other bulk DNAm data to predict their relative cell type abundances. If given an appropriate paired dataset, *scDeconv* can also deconvolve other omics, such as ATAC-seq data. Furthermore, the package also contains other functions, such as identifying cell-type-specific inter-group differential features from bulk DNAm data.

## Package overview

The package has three modules. The first is a single-omics deconvolution module. Its function scRef can construct cell transcriptomic references from scRNA-seq data. Next, the reference is transferred to the function refDeconv. It performs a recursion process to solve a constrained linear model to deconvolve bulk RNA data with the RNA reference.

The second module is a multi-omics deconvolution module. Its function epDeconv deconvolves bulk DNAm microarray data via an RNA reference. It also needs another dataset with paired RNA-DNAm data. Because of the pairing, I assume the cell contents of the RNA and DNAm samples are similar. Hence, the deconvolution results from the RNA reference and RNA samples can be shared with the paired DNAm samples to train a cell contents prediction model with DNAm features.

The third module is a feature selection module. It accepts the deconvolution results from the former modules. Then, it uses them to identify cell-type-specific inter-group differential features from the bulk data.



## Citation

Liu Y. [*scDeconv*: an R package to deconvolve bulk DNA methylation data with scRNA-seq data and paired bulk RNA-DNA methylation data.](https://academic.oup.com/bib/article-abstract/23/3/bbac150/6572659?redirectedFrom=fulltext) *Brief Bioinform*. 2022 May 13;23(3):bbac150. doi: 10.1093/bib/bbac150. PMID: 35453146; PMCID: PMC9271220.

## Installation

The code of *scDeconv* is freely available at <https://github.com/yuabrahamliu/scDeconv>.

The following commands can be used to install this R package via GitHub.

library(devtools)  
  
install\_github('yuabrahamliu/scDeconv')

It can also be installed via Bioconductor.

if (!require("BiocManager", quietly = TRUE))  
 install.packages("BiocManager")  
   
# The following initializes usage of Bioc devel  
BiocManager::install(version = 'devel')  
  
BiocManager::install("scDeconv")

## Data preparation

We will use the data that accompany with *scDeconv* package in this tutorial. They contain a *Seurat* object generated from scRNA-seq data and preprocessed by the R package *Seurat*. It is a subset of a human placenta scRNA-seq dataset in ArrayExpress, with experiment code E-MTAB-6701 (droplet-based data), and covers 1388 cells and 21737 genes. We will deconvolve 4 main placental cell types in it, including extravillous trophoblasts (EVTs), fibroblasts (FBs), Hofbauer cells (HBs), and villous cytotrophoblasts (VCTs). Among them, EVT and VCT are epithelial trophoblasts with similar origins, while HB cells are fetal macrophages in the placenta. The cell type information is contained in the metadata of the *Seurat* object. In addition, this *Seurat* object also has the gene read count data and normalized data. We will use it to generate an RNA deconvolution reference via *scDeconv*.

On the other hand, the data that need to be deconvolved are bulk DNAm data with 311 human placenta samples and 18626 probes. They are collected from 9 different GEO datasets based on the platforms of Illumina 27K and 450K and have gone through preprocessing with batch difference adjusted and the shared high-quality probes retained.

To deconvolve these DNAm data with the scRNA-seq data, *scDeconv* also needs a paired bulk RNA-bulk DNAm dataset to fulfill the trans-omics deconvolution, which is also in the accompanying data, with 48 human placenta samples. Its RNA part contains 22188 genes, and its DNAm part contains 18626 probes, the same as the ones in the DNAm data to be deconvolved. Because batch difference exists between the paired DNAm data and the ones to be deconvolved, they have been adjusted with the ComBat function in the R package *sva*, using the paired DNAm set as the reference batch.

The paired RNA data are from the platform Affymetrix Human Gene 1.0 ST Array, and the gene expression values are library size normalized values with log2 transformation. In contrast, that in the DNAm datasets are beta values. The samples in both the paired dataset and the external DNAm dataset to be deconvolved can be divided into 2 groups. One is the normal sample group, and the other is the disease group with a preeclampsia pregnancy complication. This information and the original GEO datasets of the samples can be found in the metadata frame coupled with this package.

Now, attach *scDeconv* to the R session and take a look at these data.

library(scDeconv)  
  
scRNA <- system.file('extdata', 'scRNAseqdat.rds', package = 'scDeconv')  
scRNA <- readRDS(scRNA)  
  
pRNA <- system.file('extdata', 'pairedRNAdat.rds', package = 'scDeconv')  
pRNA <- readRDS(pRNA)  
  
pDNAm <- system.file('extdata', 'pairedDNAmdat.rds', package = 'scDeconv')  
pDNAm <- readRDS(pDNAm)  
  
externalDNAm <- system.file('extdata', 'externalDNAmdat.rds', package = 'scDeconv')  
externalDNAm <- readRDS(externalDNAm)  
  
DNAmpd <- system.file('extdata', 'DNAmpd.rds', package = 'scDeconv')  
DNAmpd <- readRDS(DNAmpd)

The summary or beginning parts of these data are shown below.

#The scRNA-seq data  
scRNA  
#> An object of class Seurat   
#> 21737 features across 1388 samples within 1 assay   
#> Active assay: RNA (21737 features, 0 variable features)  
head(scRNA@meta.data)  
#> orig.ident nCount\_RNA nFeature\_RNA Fetus location  
#> FCA7196226\_CATGCCTGTCCCTTGT FCA7196226 23559 4736 F27 Placenta  
#> FCA7196226\_AGCGGTCAGCTGCAAG FCA7196226 8203 3194 F27 Placenta  
#> FCA7196226\_AATCCAGCATTGGCGC FCA7196226 18224 3602 F27 Placenta  
#> FCA7196226\_GAACATCTCTTGTACT FCA7196226 18266 4077 F27 Placenta  
#> FCA7196226\_AGATCTGGTTGCCTCT FCA7196226 17689 4169 F27 Placenta  
#> FCA7196226\_CTCGAGGAGGGCACTA FCA7196226 6314 2796 F27 Placenta  
#> annotation  
#> FCA7196226\_CATGCCTGTCCCTTGT VCT  
#> FCA7196226\_AGCGGTCAGCTGCAAG VCT  
#> FCA7196226\_AATCCAGCATTGGCGC VCT  
#> FCA7196226\_GAACATCTCTTGTACT VCT  
#> FCA7196226\_AGATCTGGTTGCCTCT VCT  
#> FCA7196226\_CTCGAGGAGGGCACTA VCT

#The paired RNA microarray data  
pRNA[1:6,1:6]  
#> GSM1940495 GSM1940496 GSM1940499 GSM1940500 GSM1940501 GSM1940502  
#> A1BG 7.376765 7.413560 7.285113 7.352853 7.488182 7.300325  
#> A1CF 5.522374 5.537723 5.463703 5.671208 5.581101 5.413397  
#> A2M 11.622694 11.120252 11.230030 11.327457 11.541988 11.442168  
#> A2ML1 5.994014 6.101948 5.982931 6.033332 6.075942 6.036454  
#> A3GALT2 6.150027 6.216505 6.046596 6.320198 6.233631 6.119112  
#> A4GALT 7.752668 8.059492 7.935501 7.676856 7.830030 7.920985

#The paired DNAm data  
pDNAm[1:6,1:6]  
#> GSM1940495 GSM1940496 GSM1940499 GSM1940500 GSM1940501 GSM1940502  
#> cg00000292 0.73358449 0.59264315 0.59496274 0.63138095 0.66699830 0.67046169  
#> cg00002426 0.69265004 0.51667233 0.47864586 0.45470146 0.51774379 0.59661450  
#> cg00003994 0.16837433 0.12105939 0.22729146 0.22437398 0.32077141 0.24005318  
#> cg00007981 0.04251201 0.02582752 0.03510695 0.03843307 0.04369758 0.03930951  
#> cg00008493 0.41044308 0.32724144 0.39277370 0.30326993 0.48137703 0.48915648  
#> cg00008713 0.05892651 0.04065540 0.06138317 0.08263494 0.09078148 0.08007630

#The external DNAm data to be deconvolved  
externalDNAm[1:6,1:6]  
#> GSM788417 GSM788419 GSM788420 GSM788421 GSM788414 GSM788415  
#> cg00000292 0.65961366 0.67591141 0.65709651 0.66077820 0.66847653 0.67436406  
#> cg00002426 0.53516824 0.53883284 0.53683120 0.53990206 0.53804637 0.53543344  
#> cg00003994 0.17674229 0.16432771 0.16631494 0.16803355 0.16416337 0.16852233  
#> cg00007981 0.03553198 0.02934814 0.03189098 0.02765783 0.02768899 0.02764183  
#> cg00008493 0.43619912 0.42998023 0.43761396 0.44973257 0.46281094 0.45935308  
#> cg00008713 0.07431056 0.06367040 0.06764089 0.06218142 0.05267028 0.05397539

#The metadata for the paired samples  
head(subset(DNAmpd, type == 'paired'))  
#> sampleid Samplegroup Gestwk dataset type  
#> 1 GSM1940495 Preeclampsia 37 GSE98224 paired  
#> 2 GSM1940496 Preeclampsia 29 GSE98224 paired  
#> 3 GSM1940499 Preeclampsia 35 GSE98224 paired  
#> 4 GSM1940500 Preeclampsia 31 GSE98224 paired  
#> 5 GSM1940501 Preeclampsia 29 GSE98224 paired  
#> 6 GSM1940502 Preeclampsia 37 GSE98224 paired  
table(subset(DNAmpd, type == 'paired')$Samplegroup)  
#>   
#> Control Preeclampsia   
#> 18 30  
#The metadata for the external samples to be deconvolved  
head(subset(DNAmpd, type == 'external'))  
#> sampleid Samplegroup Gestwk dataset type  
#> 49 GSM788417 Control 8 GSE31781 external  
#> 50 GSM788419 Control 8 GSE31781 external  
#> 51 GSM788420 Control 8 GSE31781 external  
#> 52 GSM788421 Control 9 GSE31781 external  
#> 53 GSM788414 Control 12 GSE31781 external  
#> 54 GSM788415 Control 12 GSE31781 external  
table(subset(DNAmpd, type == 'external')$Samplegroup)  
#>   
#> Control Preeclampsia   
#> 240 71

## RNA reference generation and bulk RNA data deconvolution

We will first use the scRNA-seq and paired RNA data to construct an RNA deconvolution reference. It can be achieved via the function scRef in the package.

We provide the scRNA-seq data to scRef via its parameter Seuratobj and set another parameter targetcelltypes as c('EVT', 'FB', 'HB', 'VCT'), meaning these 4 cell types in the scRNA-seq data will be covered to generate the reference. The parameter celltypecolname is set as “annotation”, indicating that the cell type information for every single cell is stored in the column “annotation” of the scRNA-seq metadata frame.

Because the first step of reference making is to synthesize several pseudo-bulk RNA-seq samples for each cell type from the scRNA-seq data, the parameter pseudobulknum is used to set how many such samples will be made for each cell type, and we set it as 100 here, meaning each cell type will get 100 pseudo-bulk RNA samples via sampling from the scRNA-seq data, and the 4 cell types will totally get 100\*4 = 400 such samples.

It should be noted that for large scRNA-seq data, this pseudo-bulk sample synthesis step of scRef can be very slow. In this case, I suggest using another function, prepseudobulk, in the package to conduct this step first. Then save the output and transfer it to scRef via its parameter pseudobulkdat to construct the reference. It is because the same scRNA-seq data may be needed to deconvolve various bulk RNA or DNAm datasets, so saving the pseudo-bulk samples can avoid repeating their synthesis for different deconvolution tasks.

Then, the synthesized samples will go through several steps to get the final reference, while if the bulk RNA data need to be deconvolved (the paired RNA data pRNA here) is provided to the parameter targetdat, a batch adjustment step will be included to remove the batch difference or platform difference between the scRNA-seq and the bulk RNA data. If no data is provided to it, this step will be skipped. Because the values in pRNA are log2 transformed values, we set the parameter targetlogged as TRUE.

#Generate the reference with scRef directly   
#refres <- scRef(Seuratobj = scRNA,   
# targetcelltypes = c('EVT', 'FB', 'HB', 'VCT'),   
# celltypecolname = 'annotation',   
# pseudobulknum = 100,   
# targetdat = pRNA,   
# targetlogged = TRUE)  
  
#Or use the function prepseudobulk to synthesize the pseudo-bulk data first,   
#and then transfer the pseudo-bulk data to scRef to generate the reference,   
#which is more suggested   
pseudobulk <- prepseudobulk(Seuratobj = scRNA,   
 targetcelltypes = c('EVT', 'FB', 'HB', 'VCT'),   
 celltypecolname = 'annotation',   
 pseudobulknum = 100)  
  
refres <- scRef(Seuratobj = scRNA,   
 targetcelltypes = c('EVT', 'FB', 'HB', 'VCT'),   
 celltypecolname = 'annotation',   
 pseudobulkdat = pseudobulk,   
 targetdat = pRNA,   
 targetlogged = TRUE)  
  
#Save the synthesized pseudo-bulk dataset so that it can be used repetitively   
#for different tasks  
#saveRDS(pseudobulk, 'pseudobulk.rds')

The result refres is a list containing 2 slots. The one named “ref” is the RNA reference generated.

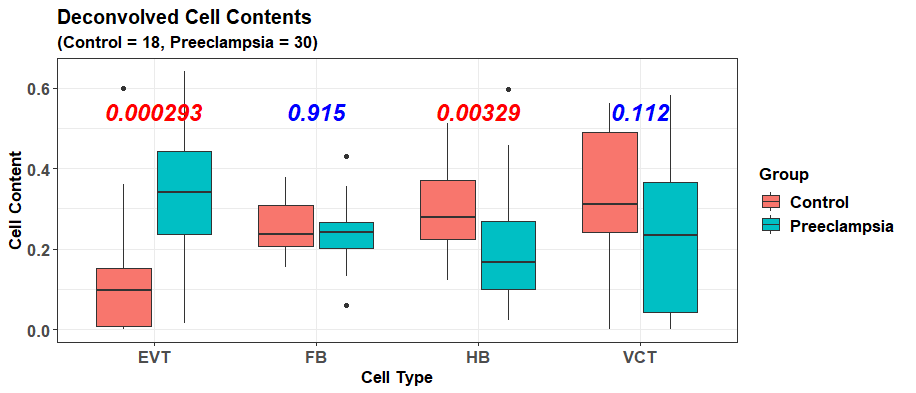
head(refres$ref)  
#> EVT FB HB VCT  
#> ZNF490 15.350183 14.510841 14.510841 14.510841  
#> CP 21.066156 20.387364 20.387364 20.387364  
#> ABCC2 14.769732 14.266494 14.266494 14.266494  
#> SULF1 12.160773 12.160773 16.972445 12.160773  
#> PADI1 26.177981 10.876944 6.304278 4.723937  
#> TLR8 8.157669 7.737533 25.150155 8.579287

The other slot named “targetnolog” is the adjusted bulk RNA data to be deconvolved, and the values in it are non-log transformed values.

refres$targetnolog[1:6,1:6]  
#> GSM1940495 GSM1940496 GSM1940499 GSM1940500 GSM1940501 GSM1940502  
#> ZNF490 18.416815 13.522850 14.930505 10.946805 13.748425 8.916959  
#> CP 0.000000 0.000000 149.774824 0.000000 0.000000 0.000000  
#> ABCC2 14.707393 7.098110 7.169431 16.389123 12.370262 10.784374  
#> SULF1 9.567271 11.716441 34.076833 2.233104 9.400352 3.464205  
#> PADI1 3.824240 7.766002 10.525161 15.920500 18.052632 3.175283  
#> TLR8 7.326088 14.035631 16.531961 4.667870 4.065535 15.483962

After getting the RNA reference and the adjusted bulk RNA data, they can be transferred to the function refDeconv in the package to deconvolve the bulk RNA data with the RNA reference. The obtained RNA reference refres$ref and the adjusted bulk RNA data refres$targetnolog can be transferred to the parameters ref and targetdat, and because the values of refres$targetnolog are non-log transformed, the parameter targetlogged is set as FALSE. Another parameter rescale can be set as TRUE so that the sum of all the deconvolved cell contents for each sample will be 1.

rnares <- refDeconv(ref = refres$ref,   
 targetdat = refres$targetnolog,   
 targetlogged = FALSE,   
 resscale = TRUE,   
 plot = TRUE,   
 pddat = subset(DNAmpd, type == 'paired'))



#Result for the bulk RNA microarray data  
head(rnares)  
#> EVT FB HB VCT  
#> GSM1940495 0.33828645 0.1931764 0.22473700 0.2438002  
#> GSM1940496 0.34555565 0.2599395 0.17158354 0.2229213  
#> GSM1940499 0.38832639 0.2543652 0.35730844 0.0000000  
#> GSM1940500 0.38614136 0.2566712 0.02217642 0.3350110  
#> GSM1940501 0.26575362 0.3056125 0.11986884 0.3087651  
#> GSM1940502 0.05083727 0.2008974 0.34875269 0.3995126

It is noteworthy that the result of refDeconv is very accurate and outperforms all other popular RNA deconvolution algorithms. The details can be found in my original publication for *scDeconv* (Liu Y. [*scDeconv*: an R package to deconvolve bulk DNA methylation data with scRNA-seq data and paired bulk RNA-DNA methylation data.](https://academic.oup.com/bib/article-abstract/23/3/bbac150/6572659?redirectedFrom=fulltext) *Brief Bioinform*. 2022 May 13;23(3):bbac150. doi: 10.1093/bib/bbac150. PMID: 35453146; PMCID: PMC9271220).

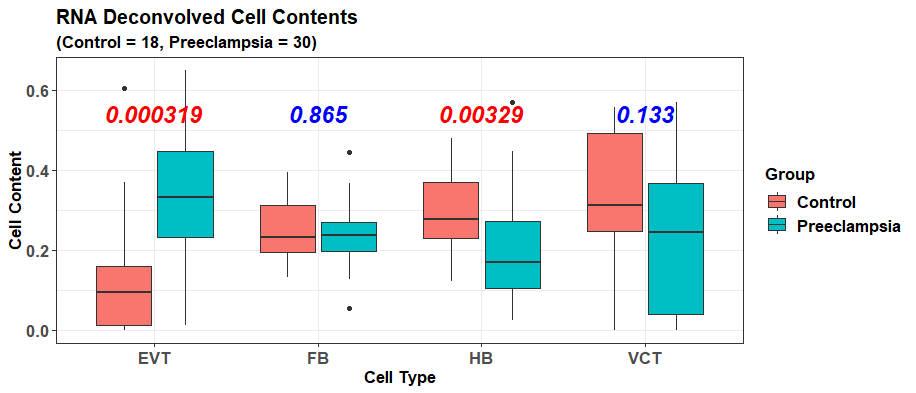
## Bulk DNAm data deconvolution with RNA reference

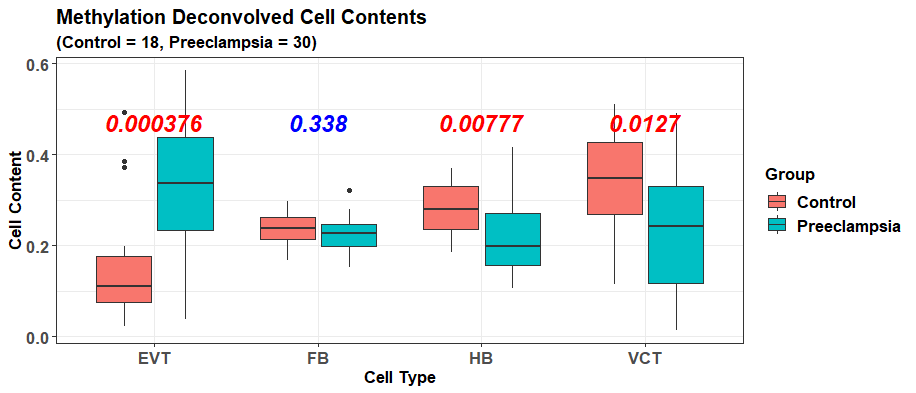
After getting the RNA reference, we use it to deconvolve the external DNAm data externalDNAm via the function epDeconv. It also needs the paired bulk RNA-bulk DNAm dataset, and for the RNA part, it is the adjusted bulk RNA data returned by scRef, and we provide it to the parameter rnamat, while for the DNAm part, we provide the data pDNAm to methylmat. Because the values in the adjusted RNA data are non-log transformed values, we set the parameter rnamatlogged as FALSE.

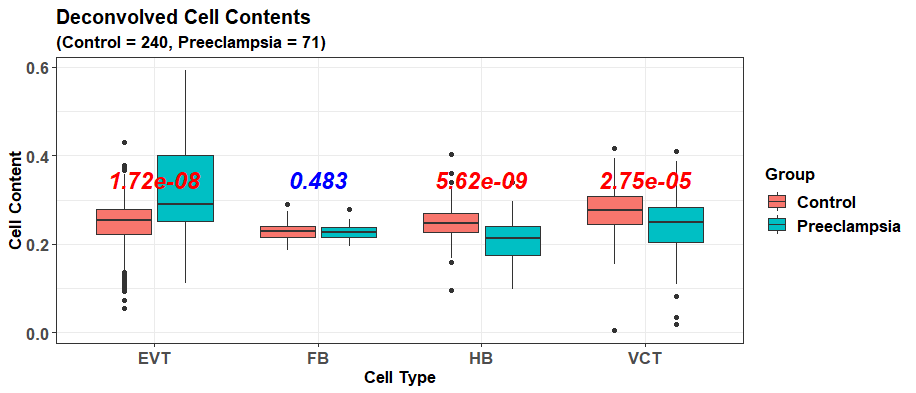
To deconvolve externalDNAm with the RNA reference refres$ref, epDeconv will construct an ensemble model on the paired RNA and DNAm data in a co-training manner and then use this model to predict the cell contents for externalDNAm. For the number of base learners of the ensemble, it is defined by the parameter learnernum, and we set it as 10 here. While because we want the 4 cell contents deconvolved can have a sum of 1 for each DNAm sample, we set the parameter resscale as TRUE.

If we want box plots to show the deconvolution results for the paired RNA, paired DNAm, and external DNAm data, we can set the parameter plot as TRUE. At the same time, we can provide the metadata frame of the paired samples to pddat, and that of the external samples to targetmethylpddat, so that the sample group information can be transferred to the function, and the cell content difference can also be shown in the plot.

dnamres <- epDeconv(rnaref = refres$ref,   
 rnamat = refres$targetnolog,   
 rnamatlogged = FALSE,   
   
 methylmat = pDNAm,   
 learnernum = 10,   
 resscale = TRUE,   
   
 targetmethyldat = externalDNAm,   
 plot = TRUE,   
 pddat = subset(DNAmpd, type == 'paired'),   
 targetmethylpddat = subset(DNAmpd, type == 'external'))

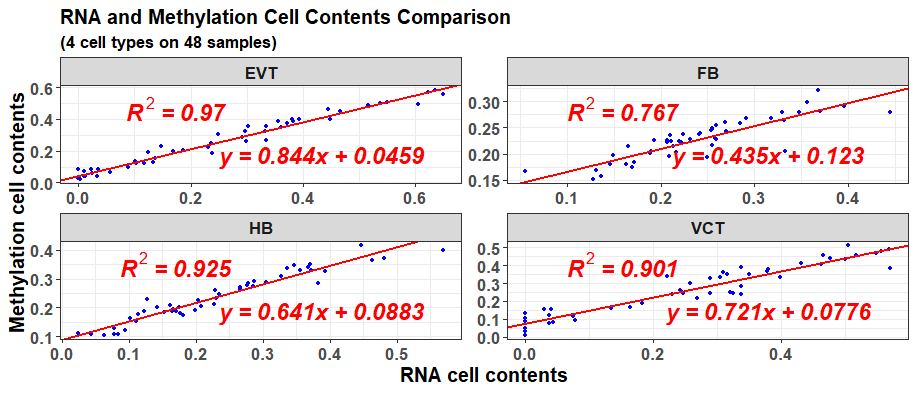






From the box plots, we can see that in all the 3 datasets, preeclampsia samples have a much higher EVT cell content than normal samples, while their HB and VCT cells are largely reduced.

In addition, 4 scatter plots are also generated for the cell types as plot is set as TRUE, and they compare the deconvolution results between the paired RNA data and the paired DNAm data. Because epDeconv constructs the model based on the assumption that paired samples have similar cell compositions, the results predicted by it should show a high correlation between the RNA and DNAm data.



The concrete values of the deconvolution results can be seen from the result dnamres. Its slots “rnacellconts” and “methylcellconts” contain the results for the paired data, while the slot “methyltargetcellconts” contains the results for the external DNAm data.

#Result for the paired RNA microarray data  
head(dnamres$rnacellconts)  
#> EVT FB HB VCT  
#> GSM1940495 0.33367213 0.1931336 0.22831875 0.2465819  
#> GSM1940496 0.33135710 0.2586181 0.16554716 0.2454366  
#> GSM1940499 0.39360023 0.2416980 0.37026306 0.0000000  
#> GSM1940500 0.38089687 0.2550985 0.02477827 0.3369090  
#> GSM1940501 0.24784508 0.3179029 0.11330019 0.3205049  
#> GSM1940502 0.05616103 0.2068742 0.35558339 0.3797657

#Result for the paired DNAm data  
head(dnamres$methylcellconts)  
#> EVT FB HB VCT  
#> GSM1940495 0.26947568 0.2267247 0.2616417 0.24215789  
#> GSM1940496 0.32627289 0.2291105 0.1934914 0.25112522  
#> GSM1940499 0.39789689 0.2396586 0.3487757 0.01366883  
#> GSM1940500 0.40208801 0.2484923 0.1127943 0.23662538  
#> GSM1940501 0.30573949 0.2670643 0.1772920 0.24990424  
#> GSM1940502 0.06658484 0.2262086 0.3310046 0.37620193

#Result for the external DNAm data  
head(dnamres$methyltargetcellcounts)  
#> EVT FB HB VCT  
#> GSM788417 0.2817118 0.2298730 0.2091309 0.2792843  
#> GSM788419 0.3122537 0.2109420 0.2037660 0.2730383  
#> GSM788420 0.2960548 0.2214043 0.2059084 0.2766325  
#> GSM788421 0.2852733 0.2101249 0.1959775 0.3086243  
#> GSM788414 0.2978430 0.2068120 0.1911144 0.3042306  
#> GSM788415 0.3099221 0.2017879 0.1769487 0.3113413

The trained ensemble model is also in dnamres; its slot “modellist” is the base learners of the model, while “normweights” is the base learner weights for the ensemble.

If externalDNAm is not provided to epDeconv, it will not influence the ensemble model training on the paired data, and the same model can still be returned. Then, externalDNAm can be transferred with the epDeconv result to the function methylpredict to predict the external sample cell contents.

externalcellconts <- methylpredict(model = dnamres,   
 targetmethyldat = externalDNAm,   
 resscale = TRUE,   
 adjustminus = TRUE)  
  
head(externalcellconts)  
#> EVT FB HB VCT  
#> GSM788417 0.2817118 0.2298730 0.2091309 0.2792843  
#> GSM788419 0.3122537 0.2109420 0.2037660 0.2730383  
#> GSM788420 0.2960548 0.2214043 0.2059084 0.2766325  
#> GSM788421 0.2852733 0.2101249 0.1959775 0.3086243  
#> GSM788414 0.2978430 0.2068120 0.1911144 0.3042306  
#> GSM788415 0.3099221 0.2017879 0.1769487 0.3113413

In addition, *scDeconv* also contains other useful functions such as celldiff, which selects cell-type-specific inter-group differential features from bulk data; and enrichwrapper, which annotates differential DNAm feature function using a correlation-based method. To make this tutorial simple and clear, they will not be covered here, but the users can explore them via the help documents and the original publication for this package.

My original publication on *scDeconv* contains more examples of its functions and is recommended to read (Liu Y. [*scDeconv*: an R package to deconvolve bulk DNA methylation data with scRNA-seq data and paired bulk RNA-DNA methylation data.](https://academic.oup.com/bib/article-abstract/23/3/bbac150/6572659?redirectedFrom=fulltext) *Brief Bioinform*. 2022 May 13;23(3):bbac150. doi: 10.1093/bib/bbac150. PMID: 35453146; PMCID: PMC9271220).

## Session information

The session information for this vignette is as follows.

sessionInfo()  
#> R version 4.0.5 (2021-03-31)  
#> Platform: x86\_64-w64-mingw32/x64 (64-bit)  
#> Running under: Windows 10 x64 (build 19042)  
#>   
#> Matrix products: default  
#>   
#> locale:  
#> [1] LC\_COLLATE=English\_United States.1252   
#> [2] LC\_CTYPE=English\_United States.1252   
#> [3] LC\_MONETARY=English\_United States.1252  
#> [4] LC\_NUMERIC=C   
#> [5] LC\_TIME=English\_United States.1252   
#>   
#> attached base packages:  
#> [1] stats graphics grDevices utils datasets methods base   
#>   
#> other attached packages:  
#> [1] Seurat\_4.0.1 SeuratObject\_4.0.0 scDeconv\_1.0.0   
#>   
#> loaded via a namespace (and not attached):  
#> [1] plyr\_1.8.6 igraph\_1.2.6   
#> [3] lazyeval\_0.2.2 splines\_4.0.5   
#> [5] BiocParallel\_1.24.1 listenv\_0.8.0   
#> [7] scattermore\_0.7 GenomeInfoDb\_1.26.7   
#> [9] ggplot2\_3.3.3 scater\_1.18.6   
#> [11] sva\_3.38.0 digest\_0.6.27   
#> [13] foreach\_1.5.1 htmltools\_0.5.1.1   
#> [15] viridis\_0.6.0 fansi\_0.4.2   
#> [17] magrittr\_2.0.1 memoise\_2.0.0   
#> [19] tensor\_1.5 cluster\_2.1.1   
#> [21] ROCR\_1.0-11 limma\_3.46.0   
#> [23] annotate\_1.68.0 globals\_0.14.0   
#> [25] matrixStats\_0.58.0 spatstat.sparse\_2.0-0   
#> [27] colorspace\_2.0-0 blob\_1.2.1   
#> [29] ggrepel\_0.9.1 xfun\_0.22   
#> [31] dplyr\_1.0.5 crayon\_1.4.1   
#> [33] RCurl\_1.98-1.3 jsonlite\_1.7.2   
#> [35] genefilter\_1.72.1 spatstat.data\_2.1-0   
#> [37] iterators\_1.0.13 survival\_3.2-10   
#> [39] zoo\_1.8-9 glue\_1.4.2   
#> [41] polyclip\_1.10-0 gtable\_0.3.0   
#> [43] zlibbioc\_1.36.0 XVector\_0.30.0   
#> [45] leiden\_0.3.7 DelayedArray\_0.16.3   
#> [47] BiocSingular\_1.6.0 shape\_1.4.5   
#> [49] future.apply\_1.7.0 SingleCellExperiment\_1.12.0  
#> [51] BiocGenerics\_0.36.0 abind\_1.4-5   
#> [53] scales\_1.1.1 pheatmap\_1.0.12   
#> [55] edgeR\_3.32.1 DBI\_1.1.1   
#> [57] miniUI\_0.1.1.1 Rcpp\_1.0.7   
#> [59] viridisLite\_0.4.0 xtable\_1.8-4   
#> [61] reticulate\_1.19 spatstat.core\_2.1-2   
#> [63] rsvd\_1.0.5 bit\_4.0.4   
#> [65] stats4\_4.0.5 glmnet\_4.1-1   
#> [67] htmlwidgets\_1.5.3 httr\_1.4.2   
#> [69] RColorBrewer\_1.1-2 ellipsis\_0.3.2   
#> [71] ica\_1.0-2 farver\_2.1.0   
#> [73] XML\_3.99-0.6 pkgconfig\_2.0.3   
#> [75] scuttle\_1.0.4 uwot\_0.1.10   
#> [77] deldir\_1.0-6 locfit\_1.5-9.4   
#> [79] utf8\_1.2.1 labeling\_0.4.2   
#> [81] tidyselect\_1.1.2 rlang\_1.0.1   
#> [83] reshape2\_1.4.4 later\_1.1.0.1   
#> [85] AnnotationDbi\_1.52.0 munsell\_0.5.0   
#> [87] tools\_4.0.5 cachem\_1.0.4   
#> [89] cli\_2.4.0 generics\_0.1.0   
#> [91] RSQLite\_2.2.5 ggridges\_0.5.3   
#> [93] evaluate\_0.14 stringr\_1.4.0   
#> [95] fastmap\_1.1.0 yaml\_2.2.1   
#> [97] goftest\_1.2-2 org.Hs.eg.db\_3.12.0   
#> [99] knitr\_1.34 bit64\_4.0.5   
#> [101] fitdistrplus\_1.1-3 purrr\_0.3.4   
#> [103] RANN\_2.6.1 sparseMatrixStats\_1.2.1   
#> [105] pbapply\_1.4-3 future\_1.21.0   
#> [107] nlme\_3.1-152 mime\_0.10   
#> [109] compiler\_4.0.5 beeswarm\_0.3.1   
#> [111] plotly\_4.9.3 png\_0.1-7   
#> [113] spatstat.utils\_2.3-0 tibble\_3.1.0   
#> [115] stringi\_1.5.3 lattice\_0.20-41   
#> [117] Matrix\_1.3-2 vctrs\_0.3.8   
#> [119] pillar\_1.7.0 lifecycle\_1.0.1   
#> [121] spatstat.geom\_2.3-2 lmtest\_0.9-38   
#> [123] BiocNeighbors\_1.8.2 RcppAnnoy\_0.0.18   
#> [125] data.table\_1.14.0 cowplot\_1.1.1   
#> [127] bitops\_1.0-6 irlba\_2.3.3   
#> [129] httpuv\_1.5.5 patchwork\_1.1.1   
#> [131] GenomicRanges\_1.42.0 R6\_2.5.0   
#> [133] promises\_1.2.0.1 KernSmooth\_2.23-18   
#> [135] gridExtra\_2.3 vipor\_0.4.5   
#> [137] IRanges\_2.24.1 parallelly\_1.25.0   
#> [139] codetools\_0.2-18 MASS\_7.3-53.1   
#> [141] assertthat\_0.2.1 SummarizedExperiment\_1.20.0  
#> [143] sctransform\_0.3.2 S4Vectors\_0.28.1   
#> [145] GenomeInfoDbData\_1.2.4 mgcv\_1.8-34   
#> [147] parallel\_4.0.5 beachmat\_2.6.4   
#> [149] grid\_4.0.5 rpart\_4.1-15   
#> [151] tidyr\_1.1.3 DelayedMatrixStats\_1.12.3   
#> [153] rmarkdown\_2.16 MatrixGenerics\_1.2.1   
#> [155] Rtsne\_0.15 Biobase\_2.50.0   
#> [157] shiny\_1.6.0 ggbeeswarm\_0.6.0