example for scRNAwrapper V1.0.0

import data

```r
library(scRNAwrapper)
```

```
## Loading required package: Seurat
```

```
## Loading required package: SingleR
```

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

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

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

```
## 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, which.max, which.min
```

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

```
##
## Attaching package: 'S4Vectors'
## The following object is masked from 'package:base':
##
##     expand.grid

## Loading required package: IRanges

## Loading required package: GenomeInfoDb

## Loading required package: Biobase

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

## Loading required package: DelayedArray

## Loading required package: matrixStats

##
## Attaching package: 'matrixStats'

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

## Loading required package: BiocParallel

##
## Attaching package: 'DelayedArray'

## The following objects are masked from 'package:matrixStats':
##
##     colMaxs, colMins, colRanges, rowMaxs, rowMins, rowRanges

## The following objects are masked from 'package:base':
##
##     aperm, apply, rowsum

##
## Attaching package: 'SummarizedExperiment'

## The following object is masked from 'package:Seurat':
##
##     Assays

## Loading required package: scater

## Loading required package: SingleCellExperiment
## Loading required package: ggplot2

## Loading required package: ComplexHeatmap

## Loading required package: grid

## ========================================
## ComplexHeatmap version 2.2.0
## Bioconductor page: http://bioconductor.org/packages/ComplexHeatmap/
## Github page: https://github.com/jokergoo/ComplexHeatmap
## Documentation: http://jokergoo.github.io/ComplexHeatmap-reference
##
## If you use it in published research, please cite:
## Gu, Z. Complex heatmaps reveal patterns and correlations in multidimensional
##   genomic data. Bioinformatics 2016.
## ========================================

## Loading required package: magrittr

## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:matrixStats':
##
##     count

## The following object is masked from 'package:Biobase':
##
##     combine

## The following objects are masked from 'package:GenomicRanges':
##
##     intersect, setdiff, union

## The following object is masked from 'package:GenomeInfoDb':
##
##     intersect

## The following objects are masked from 'package:IRanges':
##
##     collapse, desc, intersect, setdiff, slice, union

## The following objects are masked from 'package:S4Vectors':
##
##     first, intersect, rename, setdiff, setequal, union

## The following objects are masked from 'package:BiocGenerics':
##
##     combine, intersect, setdiff, union
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
sample<-readRDS("~/work/2020.1.7/data/srt.final.RDS")
sample<-subset(sample,ident=0:3)
```

auto annotate seurat cluster

using seu.singler(), annotations for each cluster is returned. you can use your own data as refernce or default HumanPrimaryCellAtlasData

Default usage

```
anno<-seu.singler(sample)
```

```
## snapshotDate(): 2019-10-22

## see ?SingleR and browseVignettes('SingleR') for documentation

## downloading 0 resources

## loading from cache

## see ?SingleR and browseVignettes('SingleR') for documentation

## downloading 0 resources

## loading from cache
```

when you have your own reference

```
ref_data<-read.csv("/home/skye/work/2020.1.7/data/metacells.BrainCellAtlas_Saunders_version_2018.04.0
rownames(ref_data)<-ref_data$X
ref_data<-ref_data[,-1]
ref_data<-as.matrix(ref_data)
anno_own<-seu.singler(sample,ref = ref_data,ref.count = TRUE,method = "cluster")
```

plot heatmap to present gene expression profiles in each clusters

in this step, you can simply plot heatmap for your seurat object, with some parameters, you can choose markers to plot or make your graph more informative

simply usage

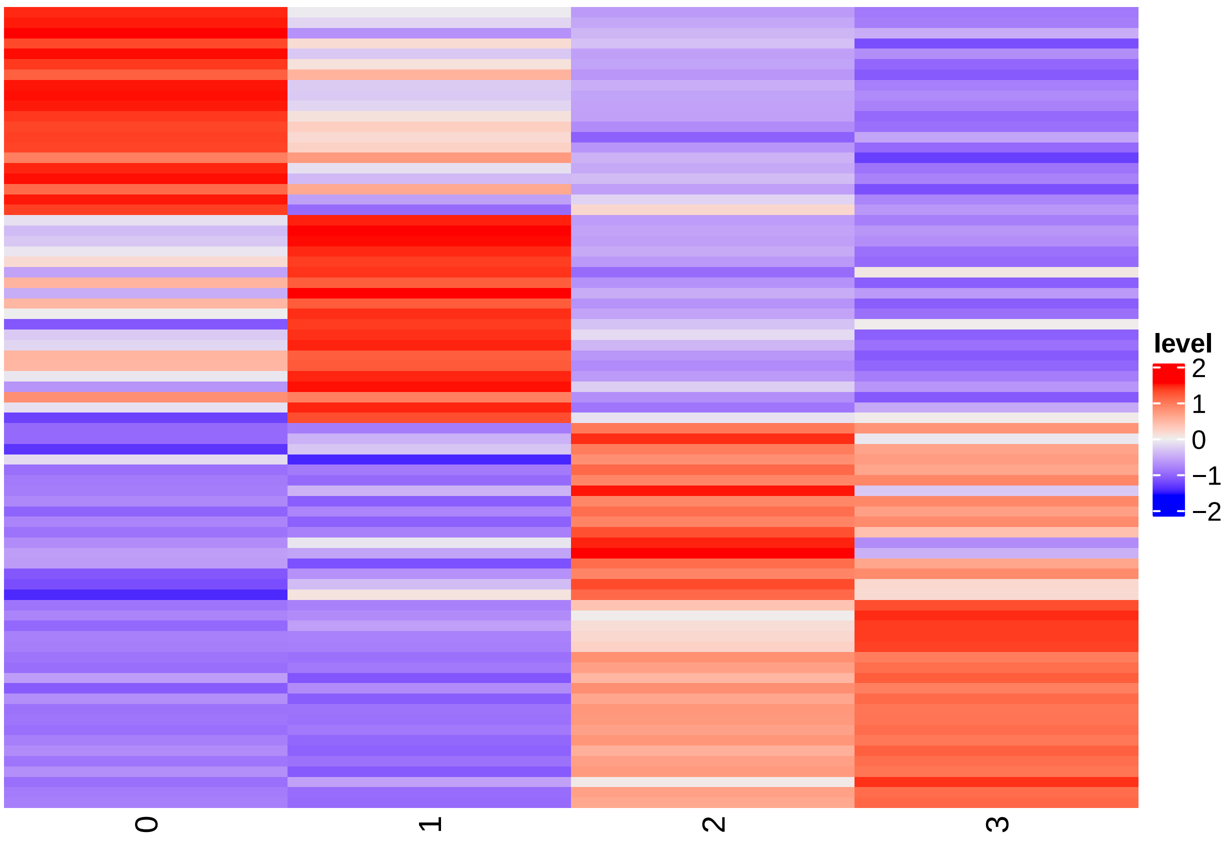
```
plot.heatmap(sample)
```

```
## Calculating cluster 0
```

```
## Calculating cluster 1
```

```
## Calculating cluster 2
```

```
## Calculating cluster 3
```



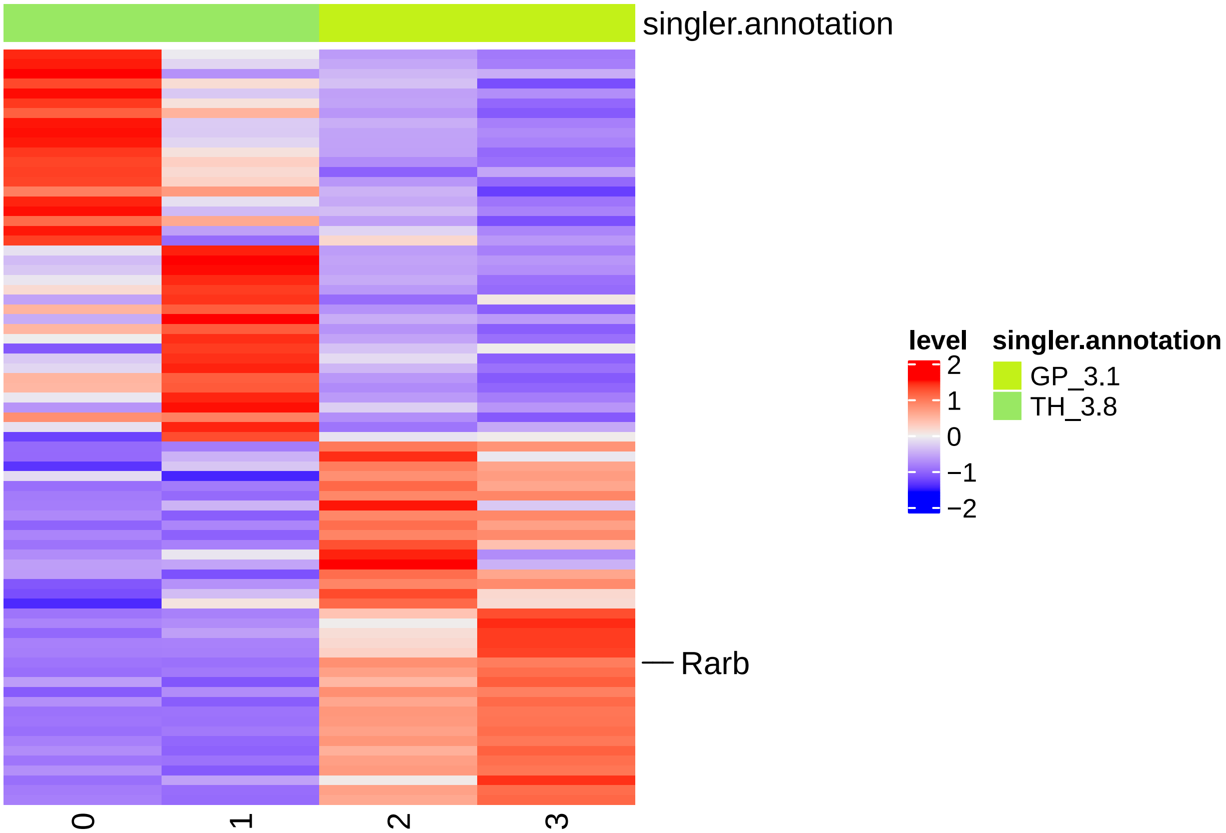
plot

```
## make cluster classification file (can have 1 or more classification type)
anno_own_class<-as.data.frame(anno_own$singler.annotation)
## anno_own_class should be in the same order as seuratObject idents
colnames(anno_own_class)<-"singler.annotation"
plot.heatmap(sample,gene.mark = "Rarb",anno.col = anno_own_class )
```

```
## Calculating cluster 0
## Calculating cluster 1

## Calculating cluster 2

## Calculating cluster 3
```


singler.annotation
Rarb
level
singler.annotation
GP_3.1
TH_3.8