

human_pbmc_seurat

April 13, 2022

1 install iterbi

```
[1]: devtools::install_github("leezx/iterbi", dependencies = F)
```

Using github PAT from envvar GITHUB_PAT

Downloading GitHub repo leezx/iterbi@HEAD

```
checking for file '/private/var/folders/fp/bzk0_6352kl60dq4fjw
nx2z80000gn/T/RtmpIilJqX/remotes1060c2358ab75/leezx-
iterbi-100d0ec/DESCRIPTION' (377ms)
preparing 'iterbi':
checking DESCRIPTION meta-information
checking for LF line-endings in source and make
files and shell scripts
checking for empty or unneeded
directories
building 'iterbi_0.4.9.tar.gz'
```

1.1 load packages

```
[2]: library(iterbi)
library(Seurat)
```

Bioconductor version '3.10' is out-of-date; the current release version '3.14' is available with R version '4.1'; see <https://bioconductor.org/install>

Registered S3 method overwritten by 'enrichplot':

```
method          from
fortify.enrichResult DOSE
```

2 download seurat dataset

```
[3]: # download Seurat test data
GetTestData()
```

3 Seurat standard analysis

```
[4]: # Load the PBMC dataset
seuset <- Read10X(data.dir = "filtered_gene_bc_matrices/hg19/")
# Initialize the Seurat object with the raw (non-normalized data).
seuset <- CreateSeuratObject(counts = seuset, project = "pbmc3k", min.cells = 3,
  min.features = 200)
seuset
```

Warning message:

"Feature names cannot have underscores ('_'), replacing with dashes ('-')"

An object of class Seurat

13714 features across 2700 samples within 1 assay

Active assay: RNA (13714 features)

```
[5]: seuset[["percent.mt"]] <- PercentageFeatureSet(seuset, pattern = "^MT-")
```

```
[6]: seuset <- subset(seuset, subset = nFeature_RNA > 200 & nFeature_RNA < 2500 &
  percent.mt < 5)
```

```
[7]: seuset <- NormalizeData(seuset)
seuset <- FindVariableFeatures(seuset, selection.method = "vst", nfeatures = 2000)
seuset <- ScaleData(seuset, features = rownames(seuset))
```

Centering and scaling data matrix

```
[8]: seuset <- RunPCA(seuset, features = VariableFeatures(object = seuset), verbose = F)
```

```
[9]: # # error from latest Matrix version
# # Error in validObject(.Object): invalid class "Graph" object: superclass
# # "Mnumeric" not defined in the
# # environment of the object's class
# packageVersion("Matrix")
```

```
# remove.packages("Matrix")
# packageurl <- "https://cran.r-project.org/src/contrib/Archive/Matrix/Matrix_1.
  ↪3-2.tar.gz"
# install.packages(packageurl, repos=NULL, type="source")
# packageVersion("Matrix")
# restart R kernel
```

```
[10]: seuset <- FindNeighbors(seuset, dims = 1:10)
```

Computing nearest neighbor graph

Computing SNN

```
[11]: seuset <- FindClusters(seuset, resolution = 0.5, verbose = F)
seuset <- RunUMAP(seuset, dims = 1:10, verbose = F)
```

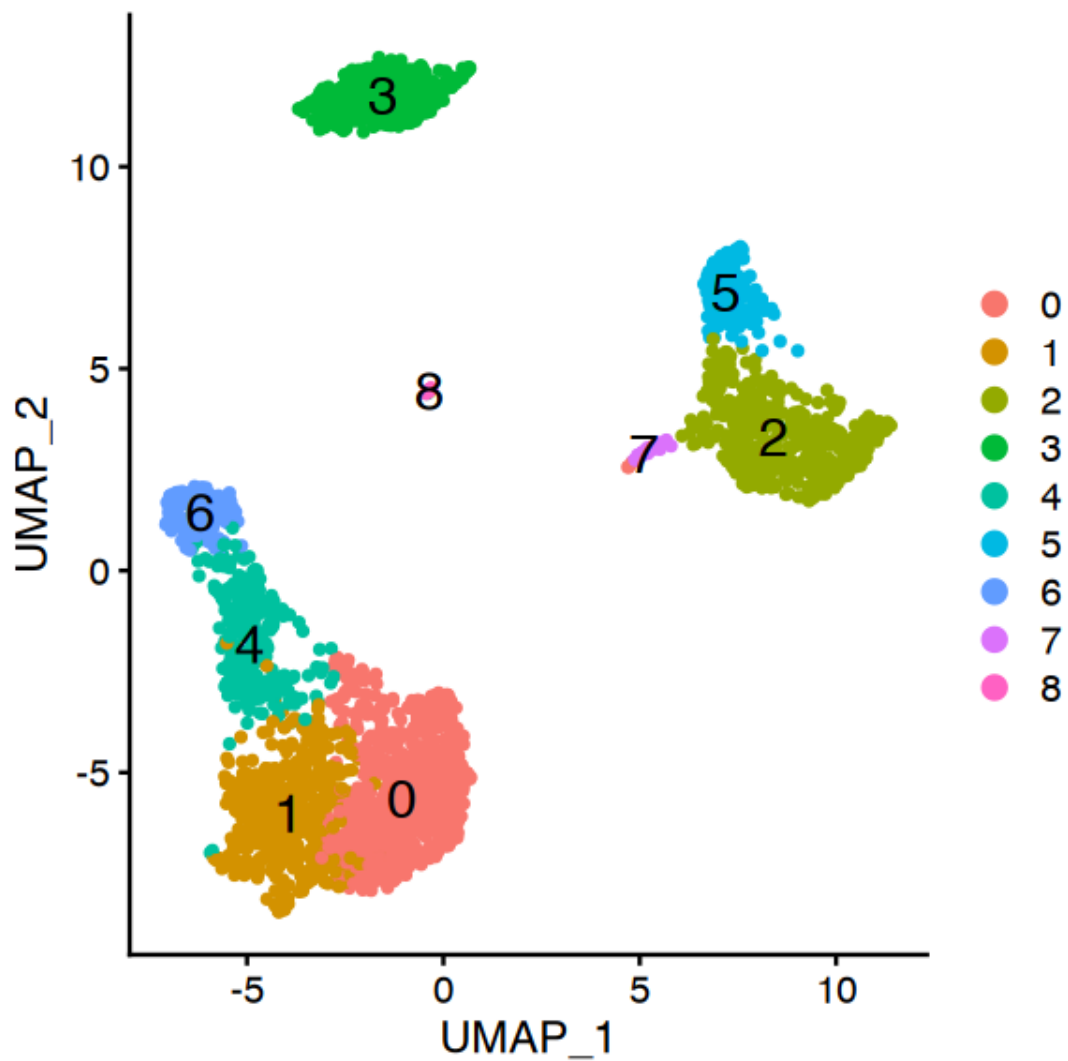
Warning message:

"The default method for RunUMAP has changed from calling Python UMAP via reticulate to the R-native UWOT using the cosine metric
To use Python UMAP via reticulate, set umap.method to 'umap-learn' and metric to 'correlation'
This message will be shown once per session"

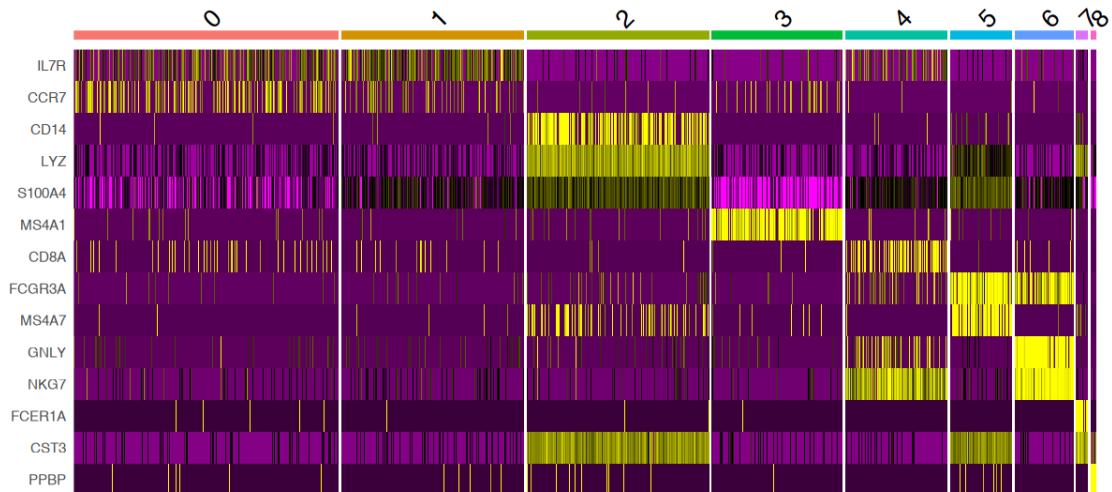
```
[12]: options(repr.plot.width=5, repr.plot.height=5)
DimPlot(seuset, reduction = "umap", label = TRUE, pt.size = 1, label.size = 6)
```

Warning message:

"Using `as.character()` on a quosure is deprecated as of rlang 0.3.0.
Please use `as_label()` or `as_name()` instead.
This warning is displayed once per session."



```
[13]: # show known markers
options(repr.plot.width=10, repr.plot.height=5)
DoHeatmap(seuset, features = c(
  "IL7R", "CCR7", "CD14", "LYZ", "IL7R", "S100A4", "MS4A1", "CD8A",
  "FCGR3A", "MS4A7", "GNLY", "NKG7", "FCER1A", "CST3", "PPBP")) + NoLegend()
```

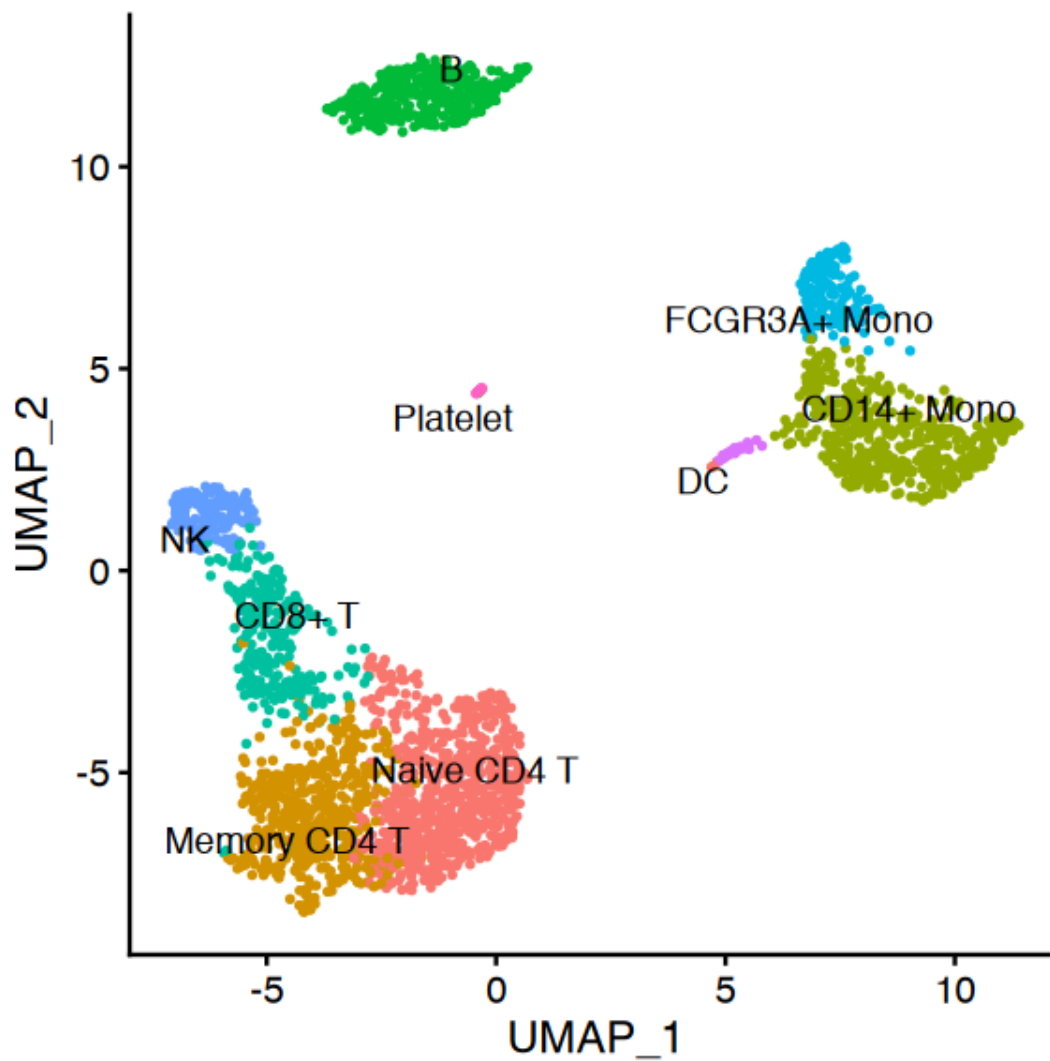


```
[14]: # 0 0 "Naive CD4+ T"
      # 1 2 "Memory CD4+"
      # 2 1 "CD14+ Mono"
      # 3 4 "B"
      # 4 3 "CD8+ T"
      # 5 5 "FCGR3A+ Mono"
      # 6 6 "NK"
      # 7 7 "DC"
      # 8 8 "Platelet"
```

```
[15]: new.cluster.ids <- c("Naive CD4 T", "Memory CD4 T", "CD14+ Mono", "B", "CD8+ T",
      "FCGR3A+ Mono",
      "NK", "DC", "Platelet")
```

```
[16]: names(new.cluster.ids) <- levels(seuset)
      seuset <- RenameIdents(seuset, new.cluster.ids)
```

```
[17]: options(repr.plot.width=5, repr.plot.height=5)
      DimPlot(seuset, reduction = "umap", label = TRUE, pt.size = 0.5, repel = T) +
      NoLegend()
```



```
[18]: # save the manual annotation
seuset$annotation <- seuset@active.ident
```

```
[19]: table(seuset$annotation)
```

Naive CD4 T	Memory CD4 T	CD14+ Mono	B	CD8+ T	FCGR3A+ Mono
697	483	480	344	271	162
NK	DC	Platelet			
155	32	14			

4 iterbi

- the index system in iterbi: L(x)_(y)
- L, level, the depth (x) of bifurcation, start from 0
- y is the cluster index at level x, start from 1

```
[3]: print(load("pbmc.seurat.Rdata"))
```

```
[1] "seuset"
```

4.1 iteratively bifurcated clustering

```
[4]: iterbi.result <- RunIterbi(seuset)
```

We are now at L0

Bifurcating L0_1 ...

Successfully split L0_1 to L1_1 and L1_2

We are now at L1

Bifurcating L1_1 ...

Successfully split L1_1 to L2_1 and L2_2

Bifurcating L1_2 ...

Successfully split L1_2 to L2_3 and L2_4

We are now at L2

Bifurcating L2_1 ...

Successfully split L2_1 to L3_1 and L3_2

Bifurcating L2_2 ...

Bifurcating L2_3 ...

Successfully split L2_3 to L3_5 and L3_6

Bifurcating L2_4 ...

only have 35 cells, set it as an end node

We are now at L3

```

Bifurcating L3_1 ...

Bifurcating L3_5 ...

Successfully split L3_5 to L4_9 and L4_10

Bifurcating L3_2 ...

Bifurcating L3_6 ...

only have 13 cells, set it as an end node

We are now at L4

Bifurcating L4_9 ...

Bifurcating L4_10 ...

We are now at L5

Bifurcating stopped! No more clusters can be split

```

```
[5]: save(iterbi.result, file = "pbmc.raw.iterbi.result.Rdata")
```

4.2 order clusters

```
[6]: iterbi.result <- OrderCluster(seuset, iterbi.result)
```

```
switch 3 4 at level L2
```

```
switch 1 2 at level L3
```

4.3 rename clusters

```
[7]: iterbi.result <- RenameIterbi(iterbi.result)
```

```
The following `from` values were not present in `x`: L0_1, L2_2, L2_3, L3_5,
L4_9
```

```
The following `from` values were not present in `x`: L2_2, L2_4, L3_2, L3_1,
L3_6, L4_10, L4_9
```

```
The following `from` values were not present in `x`: L2_2, L2_4, L3_2, L3_1,
L3_6, L4_10, L4_9
```

```
The following `from` values were not present in `x`: L0_1, L1_2, L2_2, L2_4,
```


L3_2, L3_6, L4_10

The following `from` values were not present in `x`: L0_1, L1_1, L2_1, L2_3, L3_1, L3_5, L4_9

4.4 marker details

classification of marker - expression level changed marker - have very high expression pct in both target and background - uniquely expressed marker in one cluster - uniquely expressed marker in multiple cluster

```
[15]: iterbi.result$marker_chain <- AddMarkerExpressionPct(seuset, iterbi.  
      ↪result$cellMeta, iterbi.result$marker_chain)
```

4.5 GO annotation (optional)

```
[9]: # # remove duplicate markers, each marker only has one best matched cluster  
      # iterbi.marker.chain.rmDup <- RemoveDuplicatedMarker(iterbi.  
      ↪result$marker_chain)
```

```
[10]: iterbi.GO.anno <- IterbiEnrichGO(iterbi.result$marker_chain, organism = "hs",  
      ↪pvalueCutoff = 0.05, min_count = 3)
```

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

'select()' returned 1:many mapping between keys and columns

```
[16]: table(iterbi.GO.anno$cluster)
```

L1_1	L1_2	L2_1	L2_3	L3_1	L3_2	L3_6	L4_5
10	22	14	4	15	4	4	17

4.6 write iterbi result into seurat

```
[17]: seuset <- WriteIterbiIntoSeurat(seuset, iterbi.result)
```

```
[18]: seuset@assays$iterbi$GO_chain <- iterbi.GO.anno
```

```
[19]: save(seuset, file = "pbmc.seurat.Rdata")
```

4.7 visualization

```
[17]: print(load("pbmc.seurat.Rdata"))
```

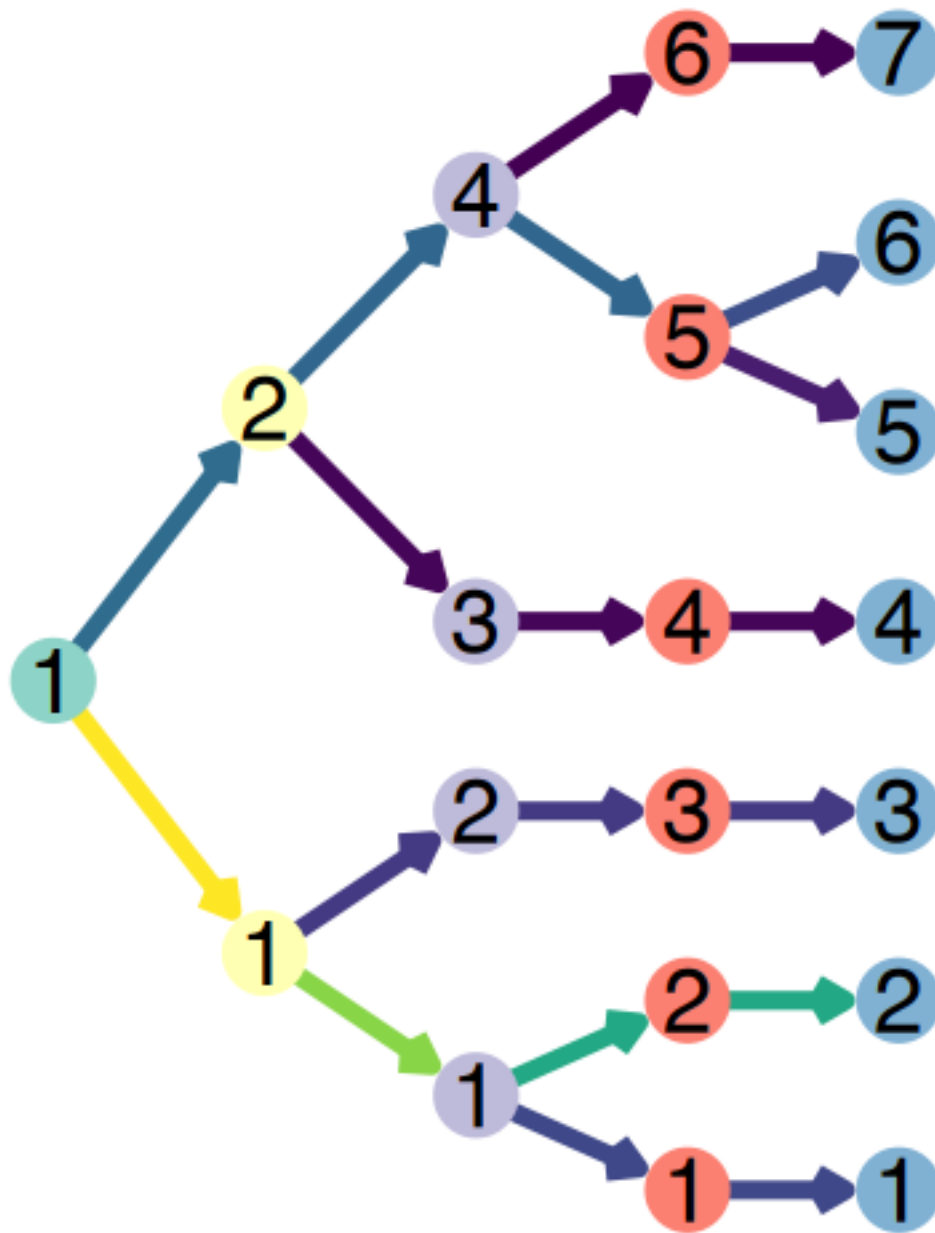
```
[1] "seuset"
```

```
[20]: iterbi.cellMeta <- seuset@assays$iterbi[["cellMeta"]]  
      iterbi.marker.chain <- seuset@assays$iterbi[["marker_chain"]]  
      iterbi.bifucation <- seuset@assays$iterbi[["bifucation"]]
```

```
[21]: go.marker.chain <- seuset@assays$iterbi[["GO_chain"]]
```

4.7.1 binary tree

```
[22]: options(repr.plot.width=3, repr.plot.height=4)  
      DrawIterbiClusterTree(seuset, iterbi.cellMeta)
```

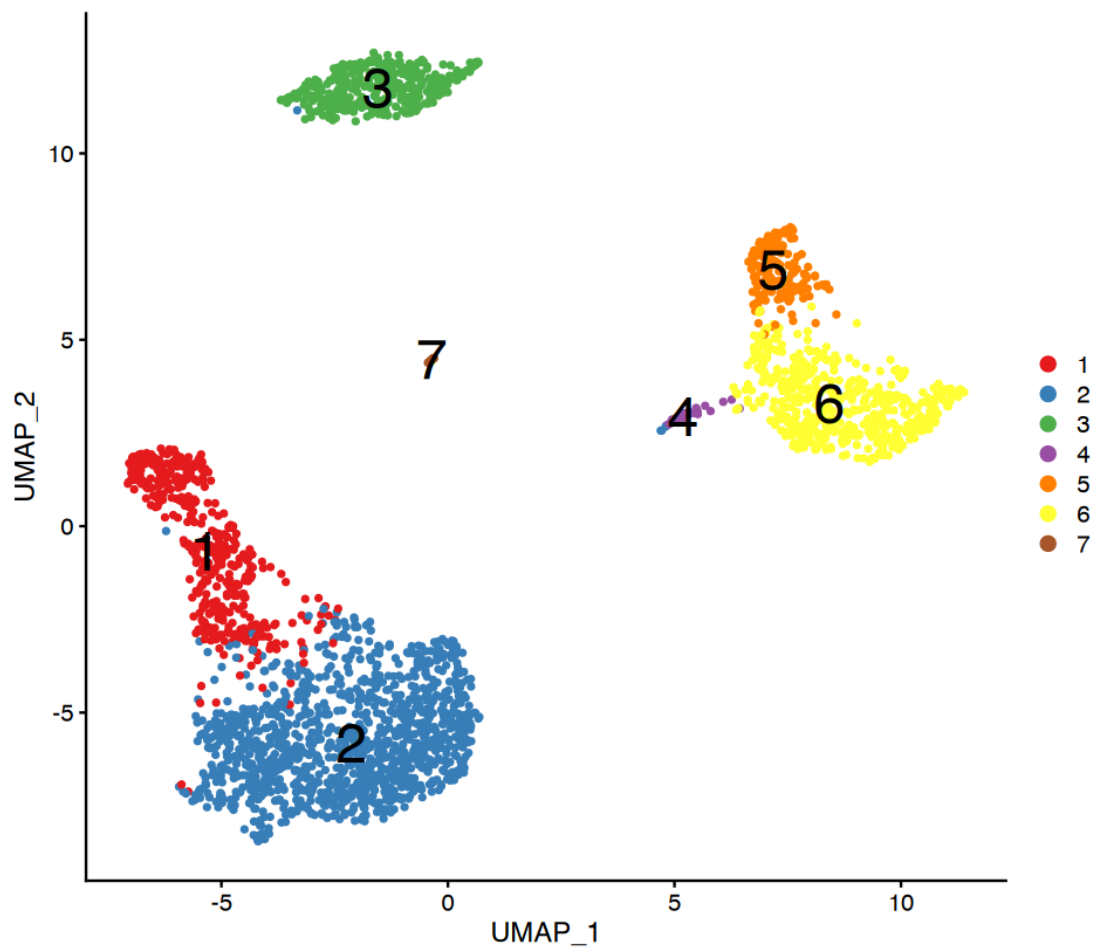


4.7.2 UMAP or tSNE

```
[23]: library(ggplot2)
```

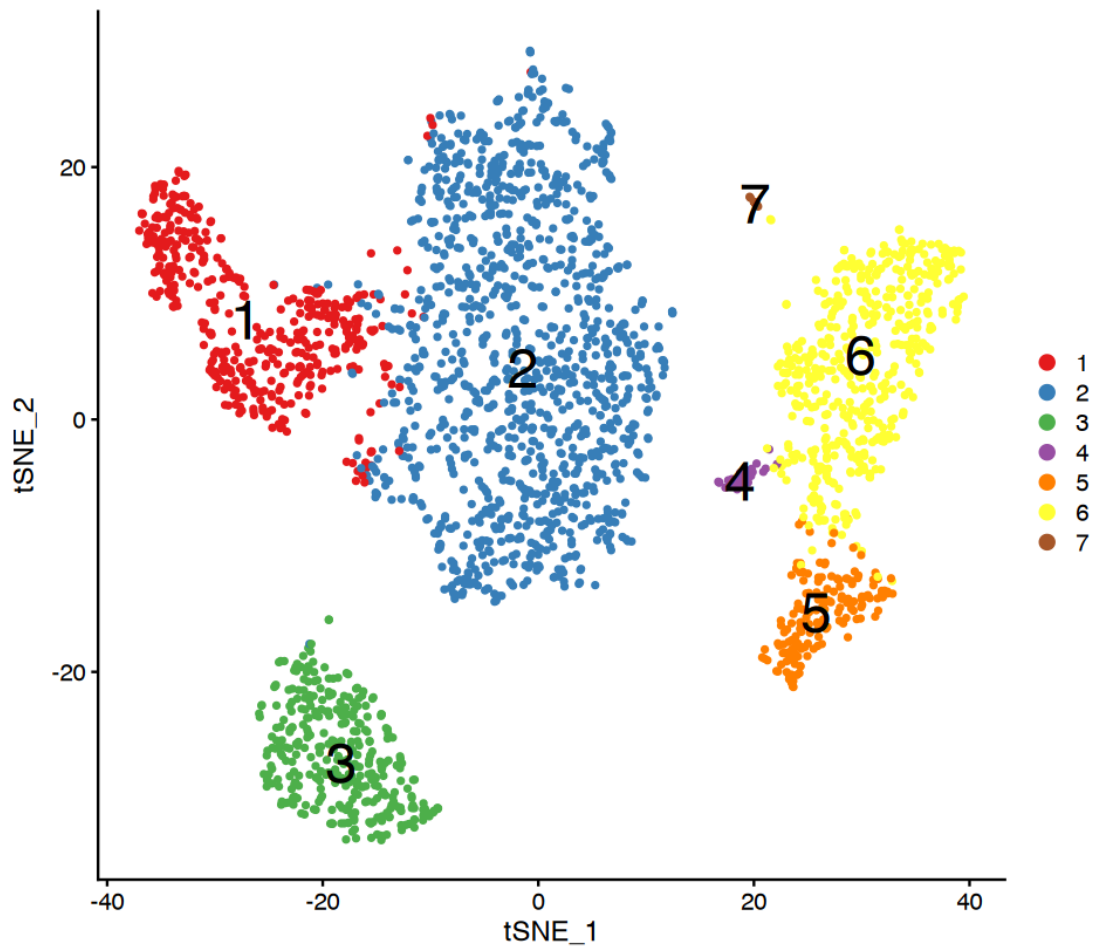
```
[24]: # similar group have continous number
options(repr.plot.width=8, repr.plot.height=7)
DimPlot(seuset, reduction = "umap", group.by = "L4", pt.size = 1, label = T,
  ↪label.size = 10) +
```

```
scale_color_manual(values=IterbiColors())
```



```
[25]: seuset <- RunTSNE(seuset, dims = 1:10)
```

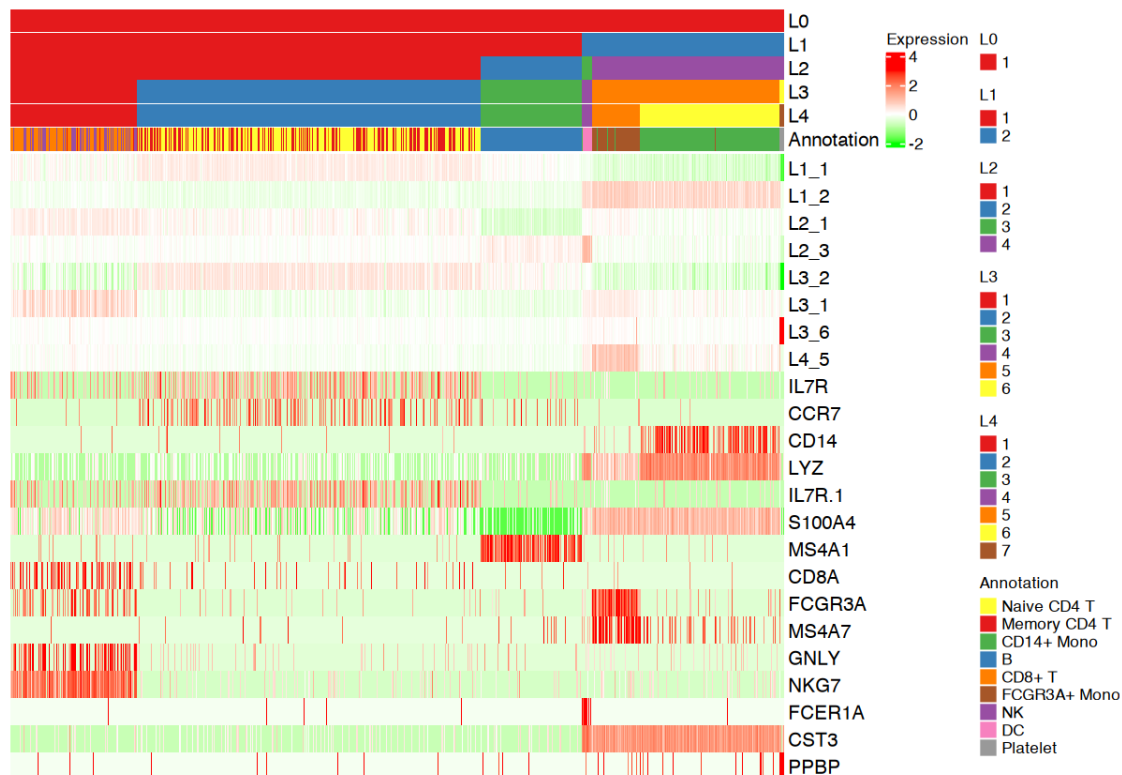
```
[26]: options(repr.plot.width=8, repr.plot.height=7)
DimPlot(seuset, reduction = "tsne", group.by = "L4", pt.size = 1, label = T,
  ↪label.size = 10) +
  scale_color_manual(values=IterbiColors())
```



4.7.3 heatmap

```
[27]: known_markers <- c(
  ↪ c("IL7R", "CCR7", "CD14", "LYZ", "IL7R", "S100A4", "MS4A1", "CD8A", "FCGR3A", "MS4A7", "GNLY", "NKG7",
    ↪ "FCER1A", "CST3", "PPBP")
```

```
[28]: options(repr.plot.width=10, repr.plot.height=7)
DrawMarkerChainHeatmap(seuset, iterbi.cellMeta, iterbi.marker.chain,
  ↪ compare_anno="annotation",
    known_markers=known_markers)
```



4.7.4 dotplot

```
[29]: library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

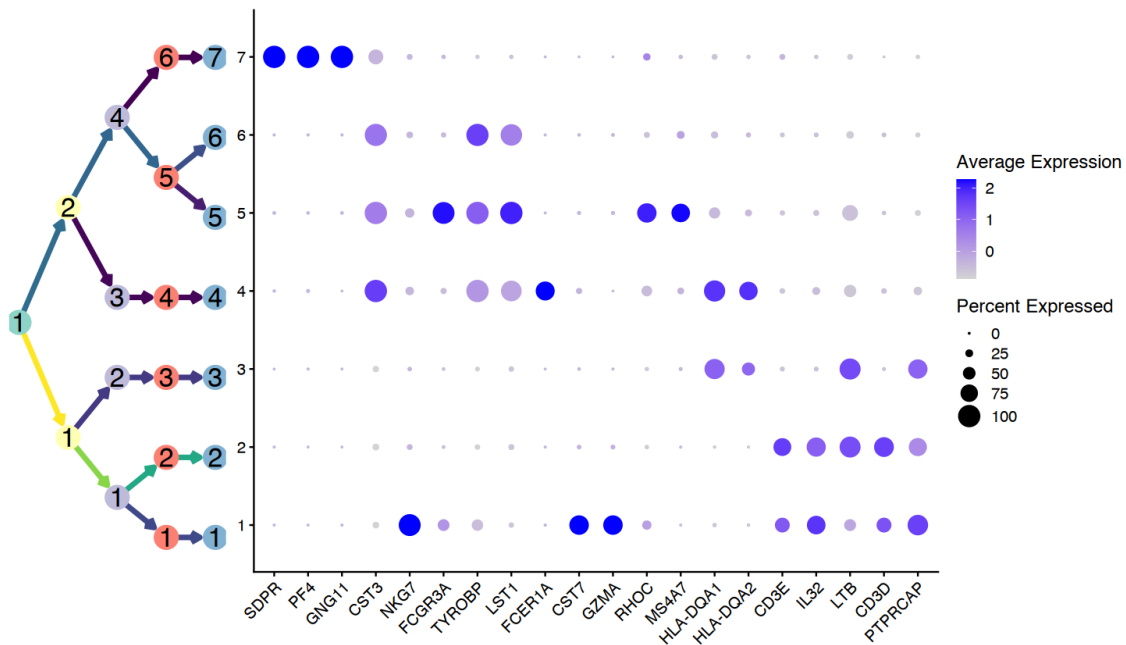
intersect, setdiff, setequal, union

```
[30]: options(repr.plot.width=12, repr.plot.height=7)
       DrawMarkerChainDotplot(seuset, iterbi.marker.chain)
```



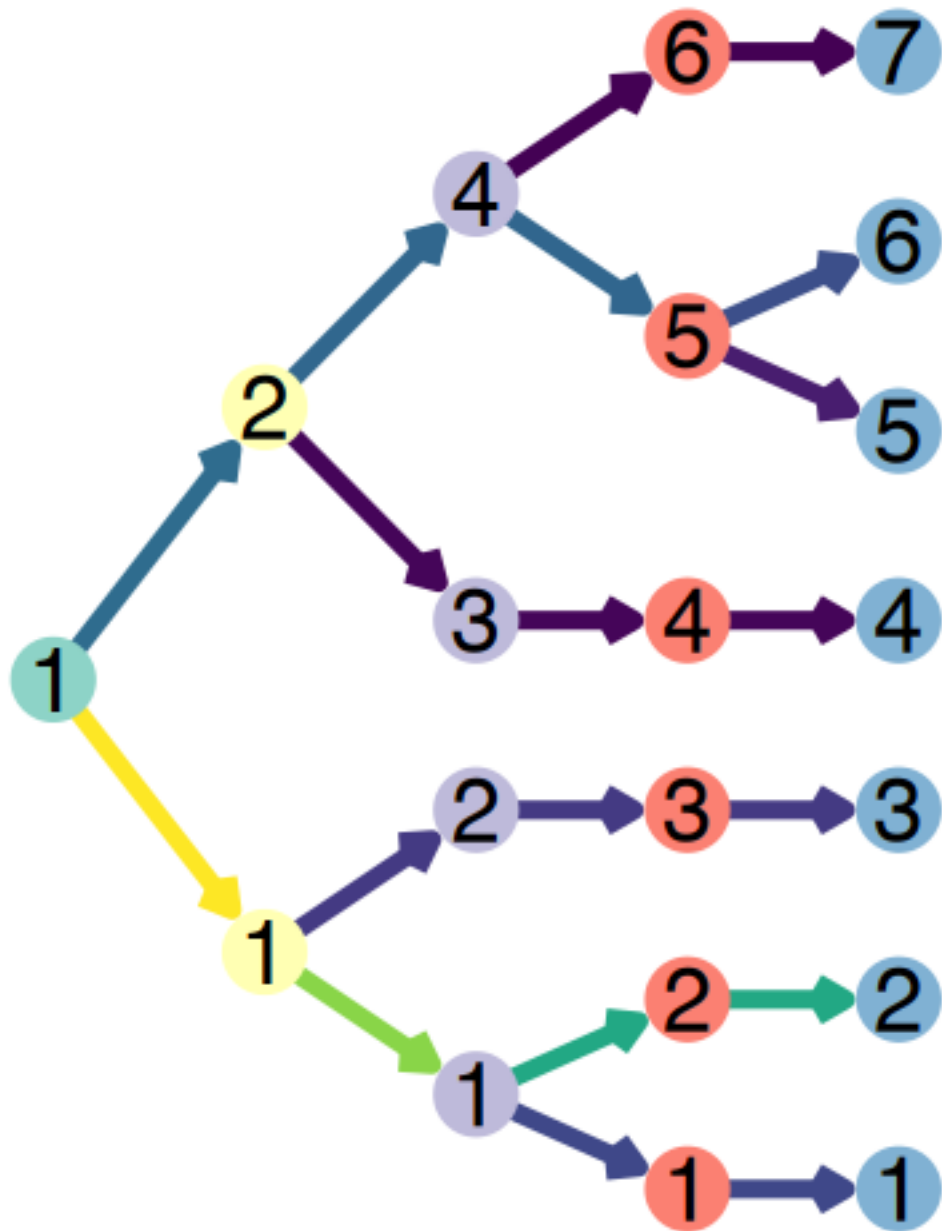
```
[31]: # filter markers
iterbi.marker.chain.uniq <- subset(iterbi.marker.chain, (cluster_pct-bcg_pct)>0.
  ↪ 3 & cluster_pct>0.7 & bcg_pct<0.5)
```

```
[32]: options(repr.plot.width=12, repr.plot.height=7)
DrawMarkerChainDotplot(seuset, iterbi.marker.chain.uniq)
```



4.8 module chain

```
[33]: options(repr.plot.width=3, repr.plot.height=4)  
      DrawIterbiClusterTree(seuset, iterbi.cellMeta)
```

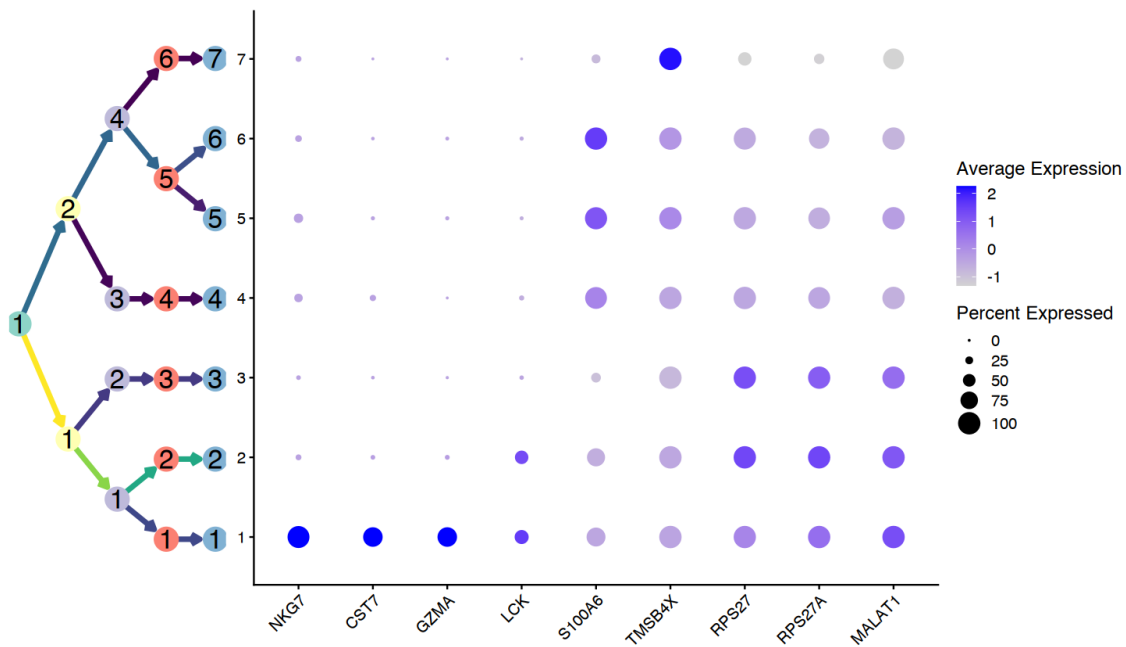



```
[34]: cluster.chain <- GetClusterChain(iterbi.cellMeta, iterbi.bifucation, "L4_1")
      cluster.chain
```

```
L0\_1 'Parent_node' L1\_1 'Parent_node' L2\_1 'Parent_node' L3\_1 'Parent_node'
L4\_1 'End_node'
```

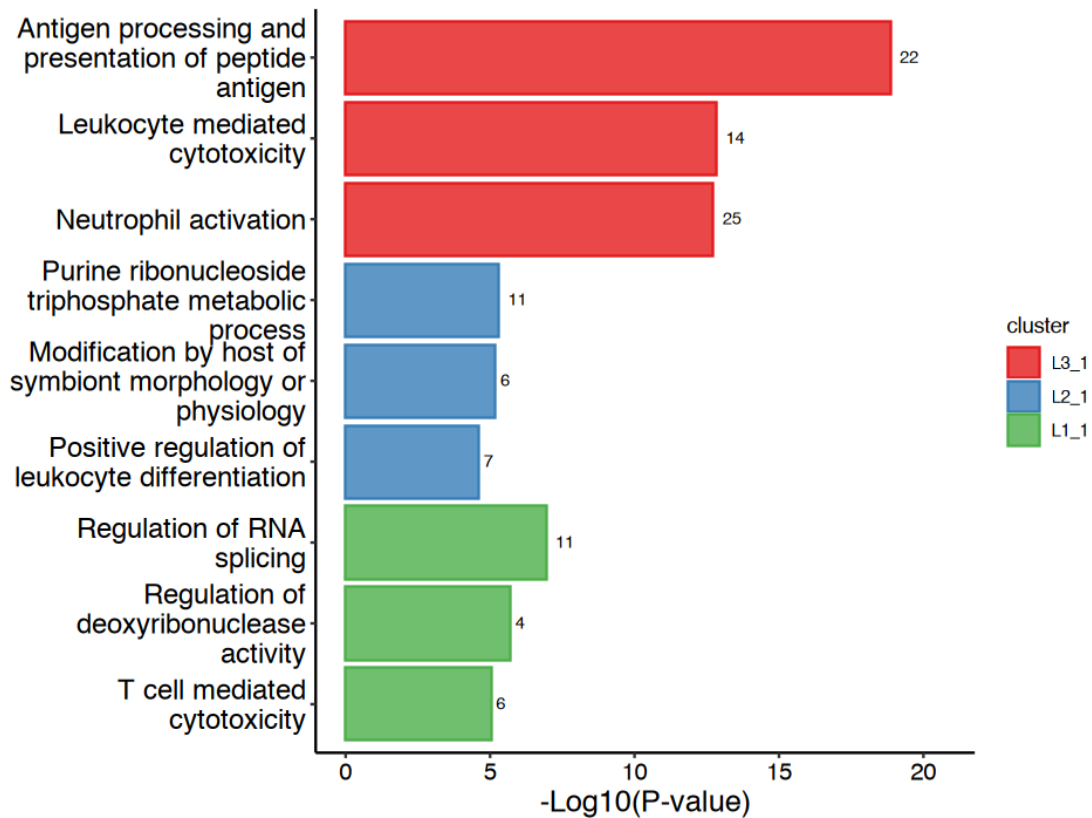
```
[35]: target.marker.chain <- subset(iterbi.marker.chain, cluster %in% names(cluster.
      ↪chain))
```

```
[36]: options(repr.plot.width=12, repr.plot.height=7)
      DrawMarkerChainDotplot(seuset, target.marker.chain)
```



4.9 GO chain

```
[37]: options(repr.plot.width=8, repr.plot.height=6)
      DrawGOchain(seuset@assays$iterbi$GO_chain, cluster.chain)
```



4.10 initial bifurcation event

```
[38]: # different level, different cluster
cluster_1 <- "L3_6"
cluster_2 <- "L4_1"
```

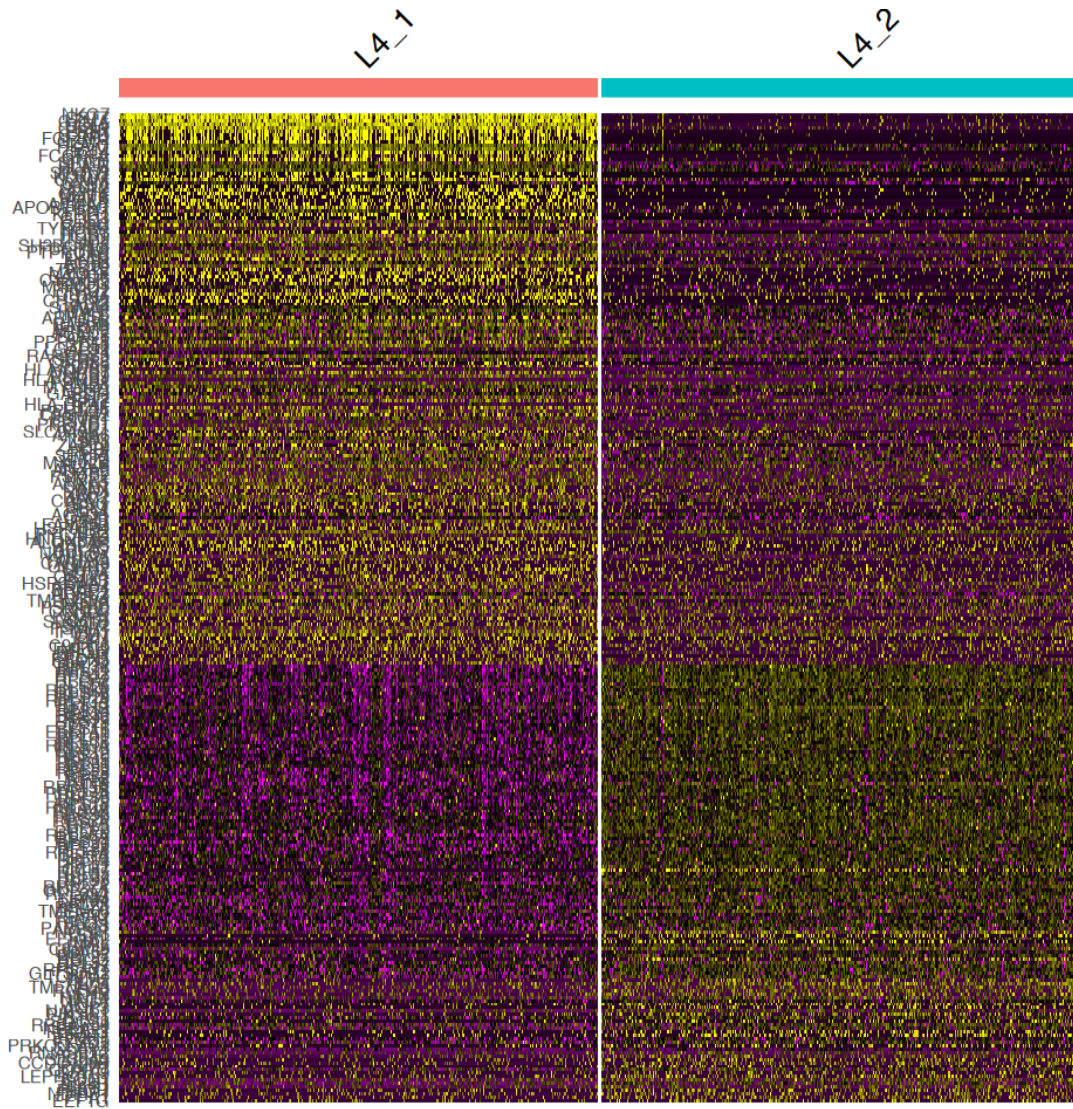
```
[ ]: # different level, same cluster
cluster_1 <- "L3_1"
cluster_2 <- "L4_1"
```

```
[42]: # same level, different cluster
cluster_1 <- "L4_1"
cluster_2 <- "L4_2"
```

```
[43]: iterbi.tracer <- GetInitialBifurcation(seuset, iterbi.cellMeta, cluster_1,
  ↪cluster_2)
```

```
[44]: tmp.seuset <- iterbi.tracer[["subset_seuratObj"]]
diff_marker_1 <- iterbi.tracer[["diff_marker_1"]]
diff_marker_2 <- iterbi.tracer[["diff_marker_2"]]
```

```
[45]: options(repr.plot.width=8, repr.plot.height=8)
DoHeatmap(tmp.seuset, features = rbind(diff_marker_1, diff_marker_2)$gene) +
  NoLegend()
```



5 sessionInfo

```
[46]: sessionInfo()
```

```
R version 3.6.2 (2019-12-12)
Platform: x86_64-apple-darwin15.6.0 (64-bit)
Running under: macOS Mojave 10.14.6
```

Matrix products: default

BLAS: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib

LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib

locale:

[1] C/UTF-8/C/C/C/C

attached base packages:

[1] stats graphics grDevices utils datasets methods base

other attached packages:

[1] dplyr_1.0.8 ggplot2_3.3.5 Seurat_3.1.4 iterbi_0.4.9

loaded via a namespace (and not attached):

[1] pbdZMQ_0.3-3	tidyr_1.2.0	bit64_4.0.5
[4] knitr_1.38	irlba_2.3.3	multcomp_1.4-12
[7] data.table_1.14.2	rpart_4.1-15	doParallel_1.0.16
[10] generics_0.1.2	metap_1.3	BiocGenerics_0.32.0
[13] org.Mm.eg.db_3.10.0	callr_3.7.0	cowplot_1.0.0
[16] TH.data_1.0-10	usethis_2.1.5	RSQLite_2.2.0
[19] RANN_2.6.1	europepmc_0.3	future_1.24.0
[22] bit_4.0.4	enrichplot_1.6.1	HiClimR_2.2.1
[25] mutoss_0.1-12	xml2_1.3.2	assertthat_0.2.1
[28] viridis_0.6.2	xfun_0.29	hms_1.1.1
[31] evaluate_0.15	fansi_0.4.2	progress_1.2.2
[34] caTools_1.18.0	igraph_1.2.11	DBI_1.1.2
[37] htmlwidgets_1.5.1	stats4_3.6.2	purrr_0.3.4
[40] ellipsis_0.3.2	backports_1.1.9	gbRd_0.4-11
[43] RcppParallel_5.0.2	vctrs_0.3.8	Biobase_2.46.0
[46] Cairo_1.5-12.2	remotes_2.2.0	here_1.0.1
[49] ROCR_1.0-7	cachem_1.0.6	withr_2.5.0
[52] ggforce_0.3.3	triebeard_0.3.0	checkmate_2.0.0
[55] sctransform_0.2.1	prettyunits_1.1.1	mnormt_1.5-6
[58] cluster_2.1.0	DOSE_3.12.0	ape_5.4
[61] IRdisplay_0.7.0	lazyeval_0.2.2	crayon_1.4.1
[64] labeling_0.4.2	pkgconfig_2.0.3	tweenr_1.0.2
[67] nlme_3.1-142	pkgload_1.1.0	nnet_7.3-12
[70] devtools_2.3.1	rlang_1.0.1	globals_0.14.0
[73] lifecycle_1.0.1	sandwich_2.5-1	clustree_0.4.3
[76] rsvd_1.0.3	rprojroot_2.0.3	polyclip_1.10-0
[79] matrixStats_0.61.0	lmtest_0.9-39	Matrix_1.3-2
[82] urltools_1.7.3	IRkernel_1.1	zoo_1.8-7
[85] base64enc_0.1-3	ggribges_0.5.2	GlobalOptions_0.1.2
[88] processx_3.5.2	png_0.1-7	viridisLite_0.4.0
[91] rjson_0.2.20	bitops_1.0-6	KernSmooth_2.23-16
[94] blob_1.2.1	shape_1.4.6	stringr_1.4.0
[97] qvalue_2.18.0	parallelly_1.31.0	gridGraphics_0.5-1
[100] jpeg_0.1-8.1	S4Vectors_0.24.4	scales_1.1.1

[103] memoise_2.0.1	magrittr_2.0.2	plyr_1.8.6
[106] ica_1.0-2	gplots_3.0.3	bibtex_0.4.2.2
[109] gdata_2.18.0	compiler_3.6.2	lsei_1.2-0
[112] RColorBrewer_1.1-2	plotrix_3.7-8	clue_0.3-60
[115] fitdistrplus_1.0-14	cli_3.1.1	listenv_0.8.0
[118] patchwork_1.0.1.9000	pbapply_1.4-3	ps_1.6.0
[121] htmlTable_2.0.1	Formula_1.2-3	MASS_7.3-55
[124] tidyselect_1.1.1	stringi_1.5.3	GOSemSim_2.12.0
[127] latticeExtra_0.6-29	ggrepel_0.9.1	grid_3.6.2
[130] fastmatch_1.1-0	tools_3.6.2	future.apply_1.6.0
[133] parallel_3.6.2	parallelDist_0.2.6	circlize_0.4.14
[136] rstudioapi_0.13	uuid_0.1-4	foreach_1.5.1
[139] foreign_0.8-72	gridExtra_2.3	farver_2.1.0
[142] Rtsne_0.15	ggraph_2.0.5	BiocManager_1.30.12
[145] rvcheck_0.1.8	digest_0.6.28	Rcpp_1.0.8.3
[148] RcppAnnoy_0.0.15	ncdf4_1.19	org.Hs.eg.db_3.10.0
[151] httr_1.4.2	AnnotationDbi_1.48.0	ComplexHeatmap_2.9.4
[154] npsurv_0.4-0	Rdpack_0.11-1	colorspace_2.0-2
[157] brio_1.1.3	fs_1.5.2	reticulate_1.24
[160] IRanges_2.20.2	splines_3.6.2	uwot_0.1.5
[163] sn_1.5-5	graphlayouts_0.7.1	multtest_2.42.0
[166] ggplotify_0.0.5	plotly_4.9.2	sessioninfo_1.1.1
[169] jsonlite_1.7.2	tidygraph_1.2.0	testthat_3.1.2
[172] R6_2.5.0	TFisher_0.2.0	Hmisc_4.4-1
[175] pillar_1.7.0	htmltools_0.5.2	glue_1.6.1
[178] fastmap_1.1.0	clusterProfiler_3.14.3	BiocParallel_1.20.1
[181] codetools_0.2-16	fgsea_1.12.0	pkgbuild_1.1.0
[184] tsne_0.1-3	mvtnorm_1.1-1	utf8_1.2.1
[187] lattice_0.20-45	tibble_3.1.6	numDeriv_2016.8-1.1
[190] curl_4.3	leiden_0.3.3	gtools_3.8.2
[193] magick_2.7.1	GO.db_3.10.0	survival_3.2-13
[196] repr_1.1.0	desc_1.4.1	munsell_0.5.0
[199] DO.db_2.9	GetoptLong_1.0.5	iterators_1.0.13
[202] reshape2_1.4.4	gtable_0.3.0	

[]: