

Mouse_CNCC_E13.5

April 13, 2022

1 install iterbi

```
[47]: 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/Rtmpm4oF0Z/remotesef72555380db/leezx-
iterbi-50f06ac/DESCRIPTION' (822ms)
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

```
[1]: 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Bioconductor version '3.10' is out-of-date; the current release version '3.14' is available with R version '4.1'; see 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 Seurat standard analysis

```
[2]: print(load("/Users/zxli/Dropbox/Projects/EllyLab/mouse/singleCell/case/Vc1_CNCC/
↳E13.5_CNCC_merged_updated.Rdata"))
```

```
[1] "seuset"    "markers"    "all_tsne"
```

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

Centering and scaling data matrix

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

```
[5]: # # 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
```

```
[6]: # seuset <- FindNeighbors(seuset, dims = 1:10)
# seuset <- FindClusters(seuset, resolution = 0.5, verbose = F)
```

```
[7]: 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"

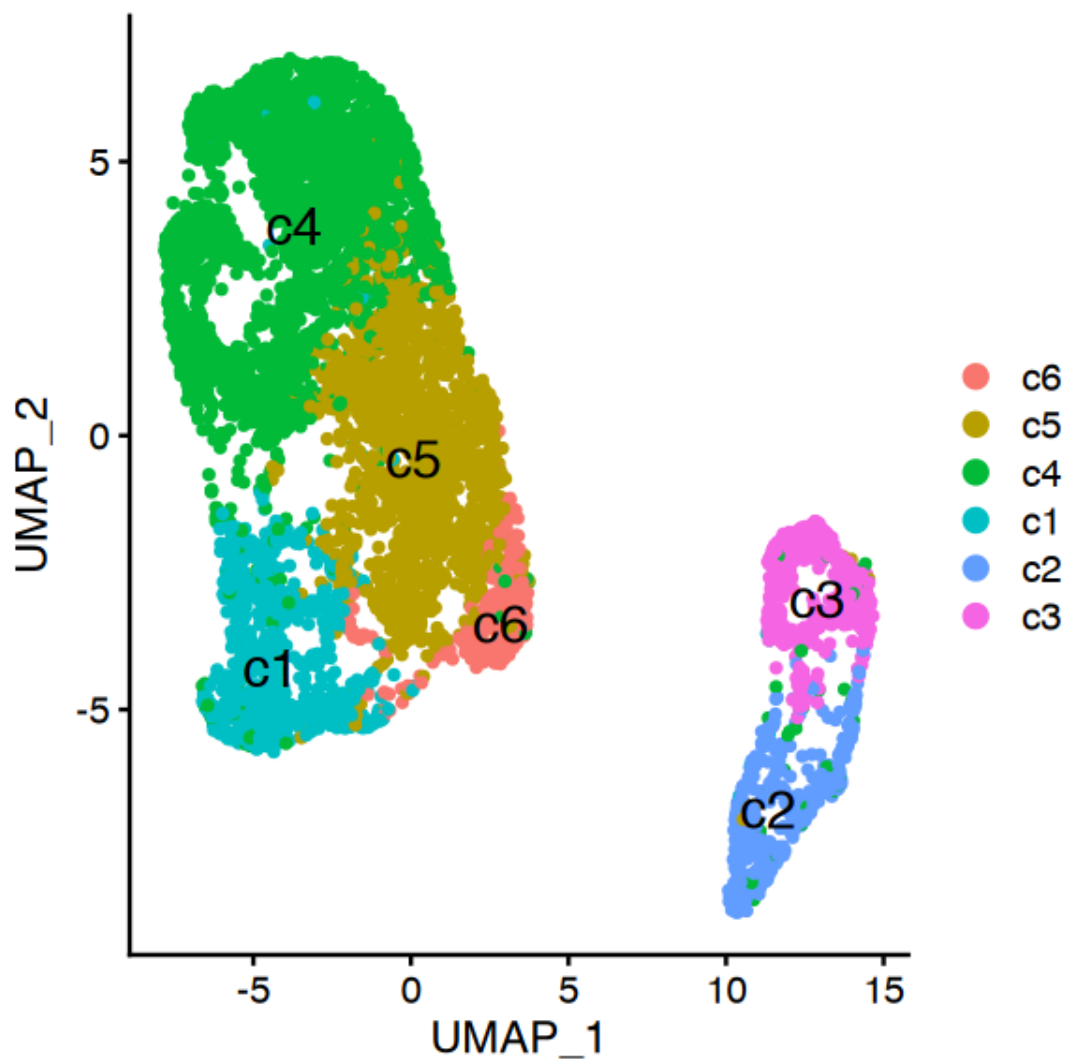
```
[8]: options(repr.plot.width=5, repr.plot.height=5)
DimPlot(seuset, reduction = "umap", label = TRUE, pt.size = 1, label.size = 6)
```

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."



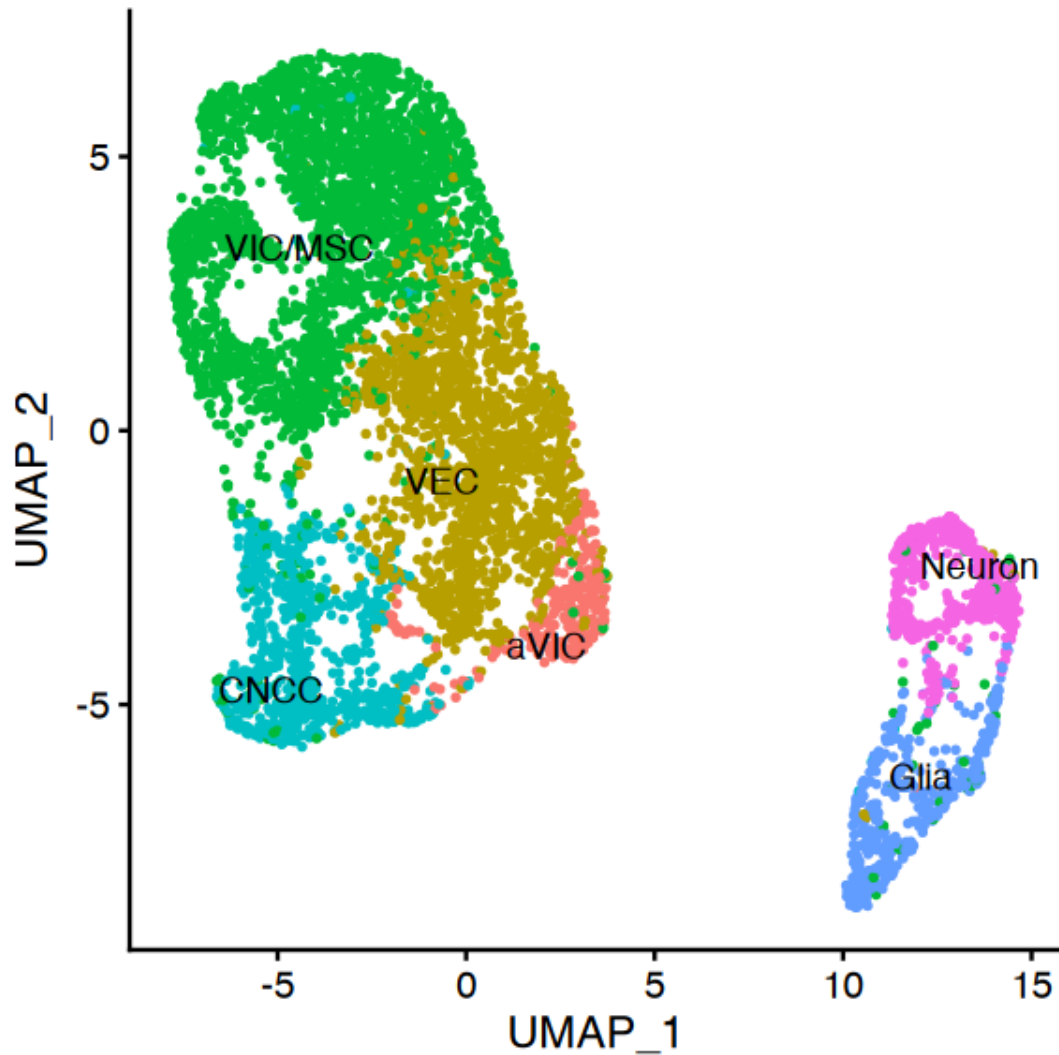
```
[10]: new.cluster.ids <- c("aVIC", "VEC", "VIC/MSC", "CNCC", "Glia", "Neuron")
```

```
[11]: levels(seuset)
```

1. 'c6' 2. 'c5' 3. 'c4' 4. 'c1' 5. 'c2' 6. 'c3'

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

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



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

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

```
aVIC    VEC VIC/MS  CNCC    Glia  Neuron
```

285 1634 2698 827 455 494

3 iterbi

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

```
[2]: print(load("CNCC.seurat.Rdata"))
```

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

3.1 iteratively bifurcated clustering

```
[3]: 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_4 ...

Successfully split L2_4 to L3_7 and L3_8

Bifurcating L2_3 ...

Successfully split L2_3 to L3_5 and L3_6

We are now at L3

Bifurcating L3_1 ...

Successfully split L3_1 to L4_1 and L4_2

Bifurcating L3_7 ...

Successfully split L3_7 to L4_13 and L4_14

Bifurcating L3_6 ...

Bifurcating L3_2 ...

Warning message in irlba(A = t(x = object), nv = npcs, ...):

"You're computing too large a percentage of total singular values, use a standard svd instead."

Bifurcating L3_5 ...

Successfully split L3_5 to L4_9 and L4_10

Bifurcating L3_8 ...

We are now at L4

Bifurcating L4_1 ...

Successfully split L4_1 to L5_1 and L5_2

Bifurcating L4_13 ...

Bifurcating L4_2 ...

Successfully split L4_2 to L5_3 and L5_4

Bifurcating L4_14 ...

Successfully split L4_14 to L5_27 and L5_28

Bifurcating L4_9 ...

Bifurcating L4_10 ...

We are now at L5

Bifurcating L5_2 ...

Bifurcating L5_1 ...

Successfully split L5_1 to L6_1 and L6_2

Bifurcating L5_4 ...

Successfully split L5_4 to L6_7 and L6_8

Bifurcating L5_28 ...

only have 45 cells, set it as an end node

Bifurcating L5_27 ...

Warning message in irlba(A = t(x = object), nv = npcs, ...):

"You're computing too large a percentage of total singular values, use a standard svd instead."

Bifurcating L5_3 ...

Successfully split L5_3 to L6_5 and L6_6

We are now at L6

Bifurcating L6_1 ...

Bifurcating L6_7 ...

Successfully split L6_7 to L7_13 and L7_14

Bifurcating L6_8 ...

Bifurcating L6_5 ...

Successfully split L6_5 to L7_9 and L7_10

Bifurcating L6_6 ...

Bifurcating L6_2 ...

We are now at L7

Bifurcating L7_13 ...

Bifurcating L7_9 ...

Bifurcating L7_14 ...

Bifurcating L7_10 ...

```
Warning message in irlba(A = t(x = object), nv = npcs, ...):  
"You're computing too large a percentage of total singular values, use a  
standard svd instead."  
We are now at L8
```

Bifurcating stopped! No more clusters can be split

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

3.2 order clusters

```
[3]: print(load("CNCC.raw.iterbi.result.Rdata"))
```

```
[1] "iterbi.result"
```

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

switch 3 4 at level L2

switch 1 2 at level L3

switch 7 8 at level L3

switch 5 6 at level L3

switch 3 4 at level L5

switch 7 8 at level L6

3.3 rename clusters

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

The following `from` values were not present in `x`: L0_1, L2_2, L3_1, L4_9, L5_2, L5_27, L6_7, L7_13, L7_9

The following `from` values were not present in `x`: L2_2, L3_2, L3_8, L3_6, L4_13, L4_10, L4_9, L5_2, L5_27, L5_28, L6_1, L6_2, L6_8, L6_6, L7_13, L7_14, L7_10, L7_9

The following `from` values were not present in `x`: L2_2, L3_2, L3_8, L3_6, L4_13, L4_10, L4_9, L5_2, L5_27, L5_28, L6_1, L6_2, L6_8, L6_6, L7_13, L7_14, L7_10, L7_9

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

L3_2, L3_8, L3_6, L4_2, L4_14, L4_10, L5_2, L5_4, L5_28, L6_2, L6_8, L6_6,
L7_14, L7_10

The following `from` values were not present in `x`: L0_1, L1_1, L2_1, L2_3,
L3_1, L3_7, L3_5, L4_1, L4_13, L4_9, L5_1, L5_3, L5_27, L6_1, L6_7, L6_5, L7_13,
L7_9

3.4 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

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

3.5 GO annotation (optional)

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

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

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"1.48% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"1.65% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"3.53% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"2.41% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"4.49% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"1.95% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"2.37% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"2.8% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"3.36% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"2.23% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"3.4% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"5.56% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"2.62% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"2.84% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"7.05% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"4.07% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"4.96% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"5.93% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"5.97% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"0.65% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"6.99% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"2.42% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :

"1.94% of input gene IDs are fail to map..."

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

Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",

```
"ENTREZID"), :
"1.14% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns
```

```
Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"5.39% of input gene IDs are fail to map..."
'select()' returned 1:many mapping between keys and columns
```

```
Warning message in bitr(genes, fromType = "SYMBOL", toType = c("ENSEMBL",
"ENTREZID"), :
"3.48% of input gene IDs are fail to map..."
```

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

L1_1	L1_2	L2_1	L2_3	L2_4	L3_1	L3_4	L3_5	L3_6	L3_7	L4_2	L4_3	L4_6
29	32	18	20	30	15	19	33	27	24	18	19	22
L4_7	L4_9	L5_10	L5_2	L5_4	L5_5	L6_2	L6_3	L6_5	L6_7	L6_8	L7_7	L7_8
14	8	13	15	10	13	17	17	15	13	17	8	13

3.6 write iterbi result into seurat

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

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

```
[13]: save(seuset, file = "CNCC.seurat.Rdata")
```

3.7 visualization

```
[4]: print(load("CNCC.seurat.Rdata"))
```

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

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

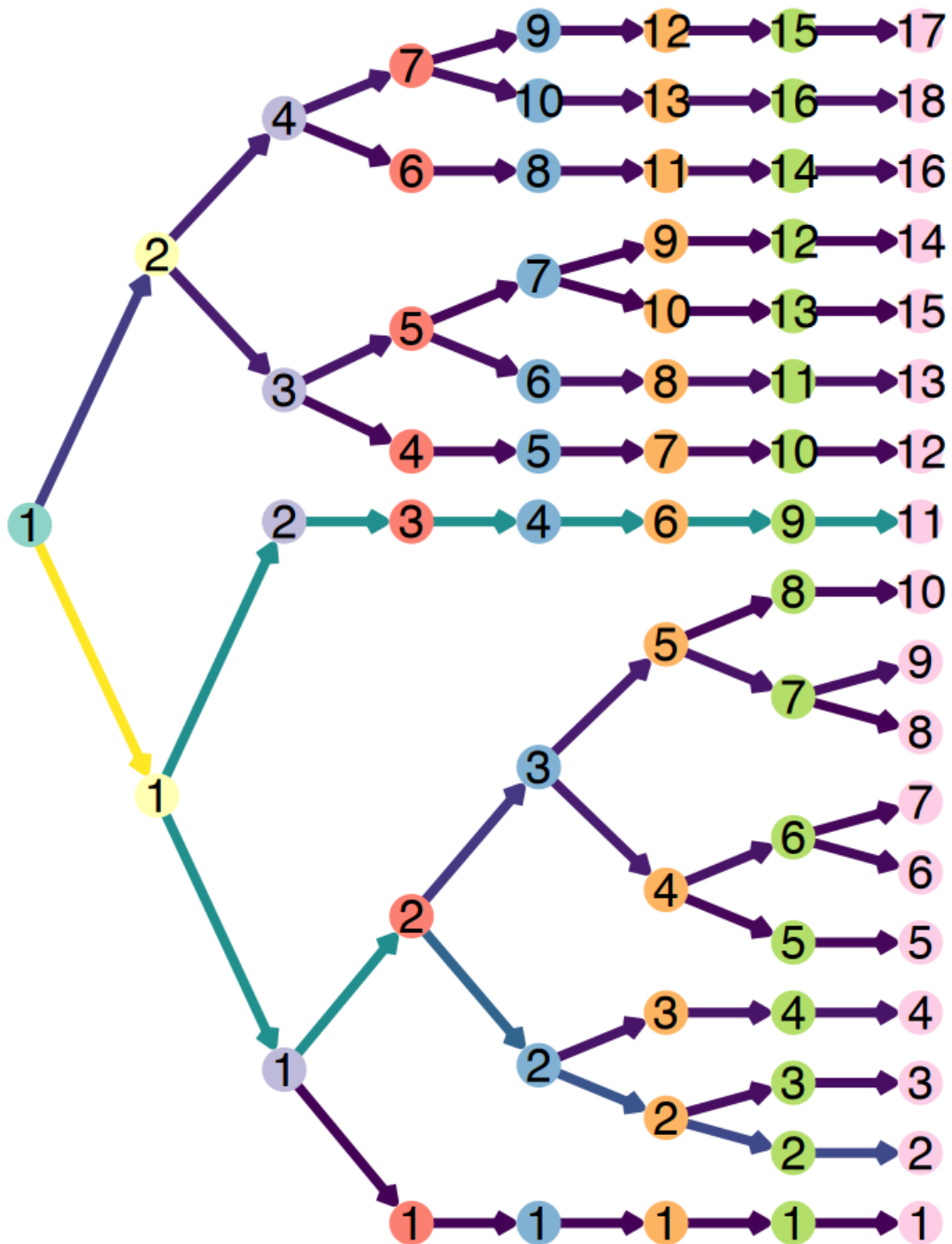
```
[15]: go.marker.chain <- seuset@assays$iterbi[["GO_chain"]]
```

3.7.1 binary tree

```
[16]: options(repr.plot.width=6, repr.plot.height=8)
DrawIterbiClusterTree(seuset, iterbi.cellMeta)
```

```
Warning message:
"The `add` argument of `group_by()` is deprecated as of dplyr 1.0.0.
```

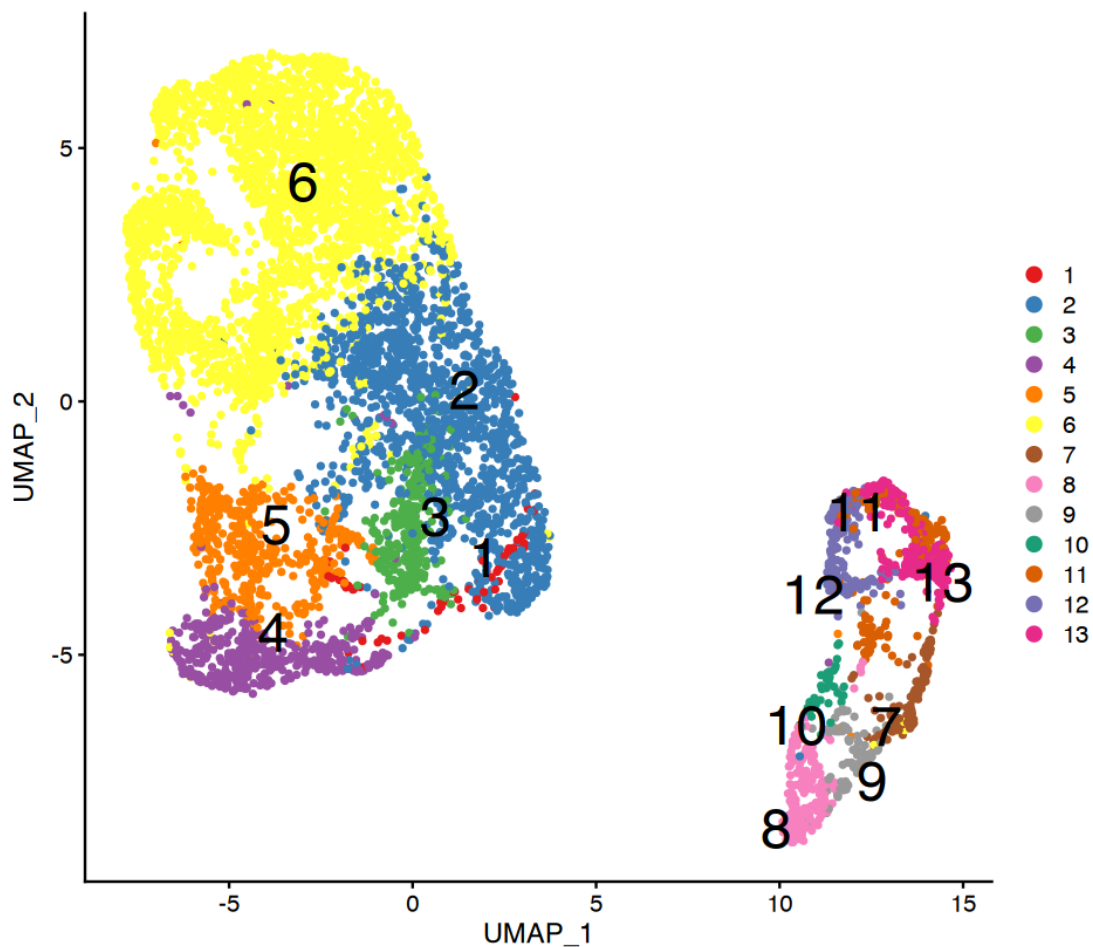
Please use the ``add`` argument instead.
This warning is displayed once every 8 hours.
Call ``lifecycle::last_lifecycle_warnings()`` to see where this warning was generated."



3.7.2 UMAP or tSNE

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

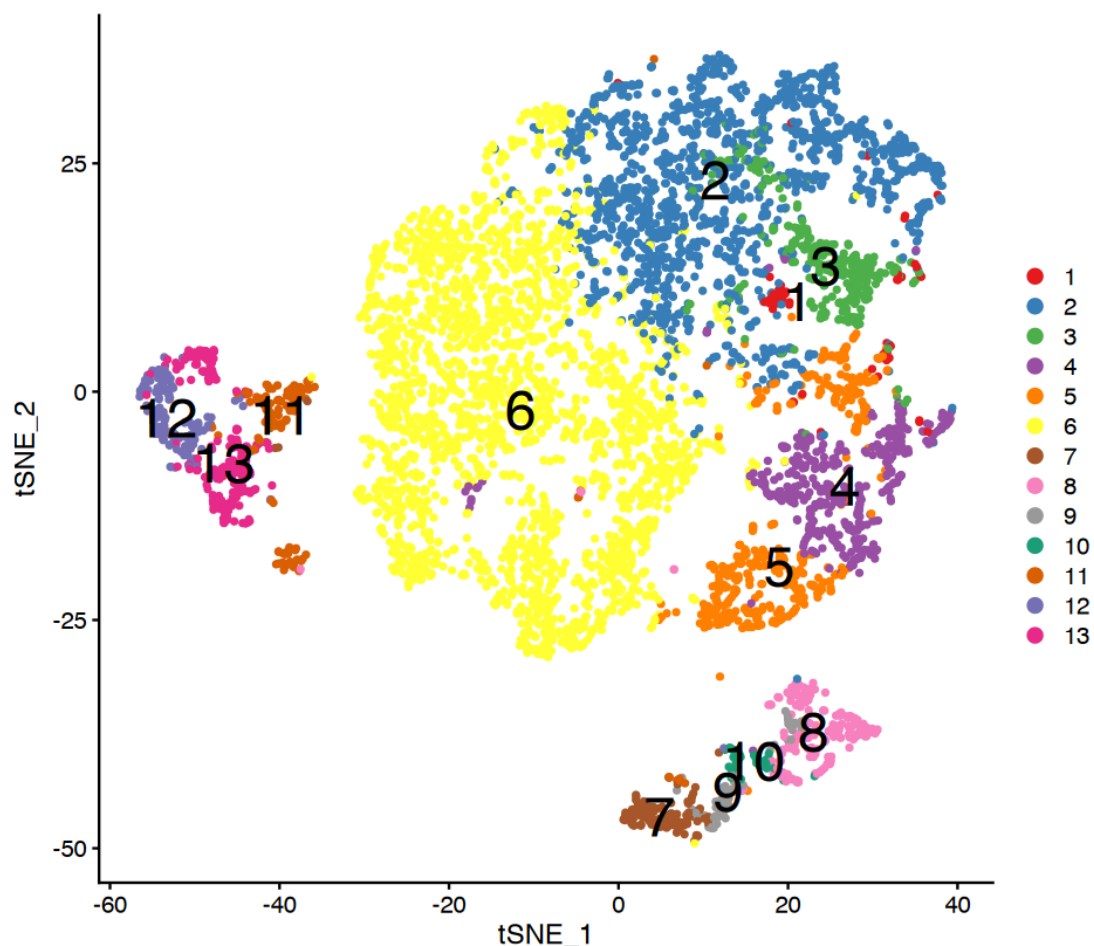
```
[19]: options(repr.plot.width=8, repr.plot.height=7)
DimPlot(seuset, reduction = "umap", group.by = "L5", pt.size = 1, label = T,
  ↪label.size = 10, repel = T) +
  scale_color_manual(values=IterbiColors())
```



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

```
[21]: options(repr.plot.width=8, repr.plot.height=7)
DimPlot(seuset, reduction = "tsne", group.by = "L5", pt.size = 1, label = T,
  ↪label.size = 10) +
```

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



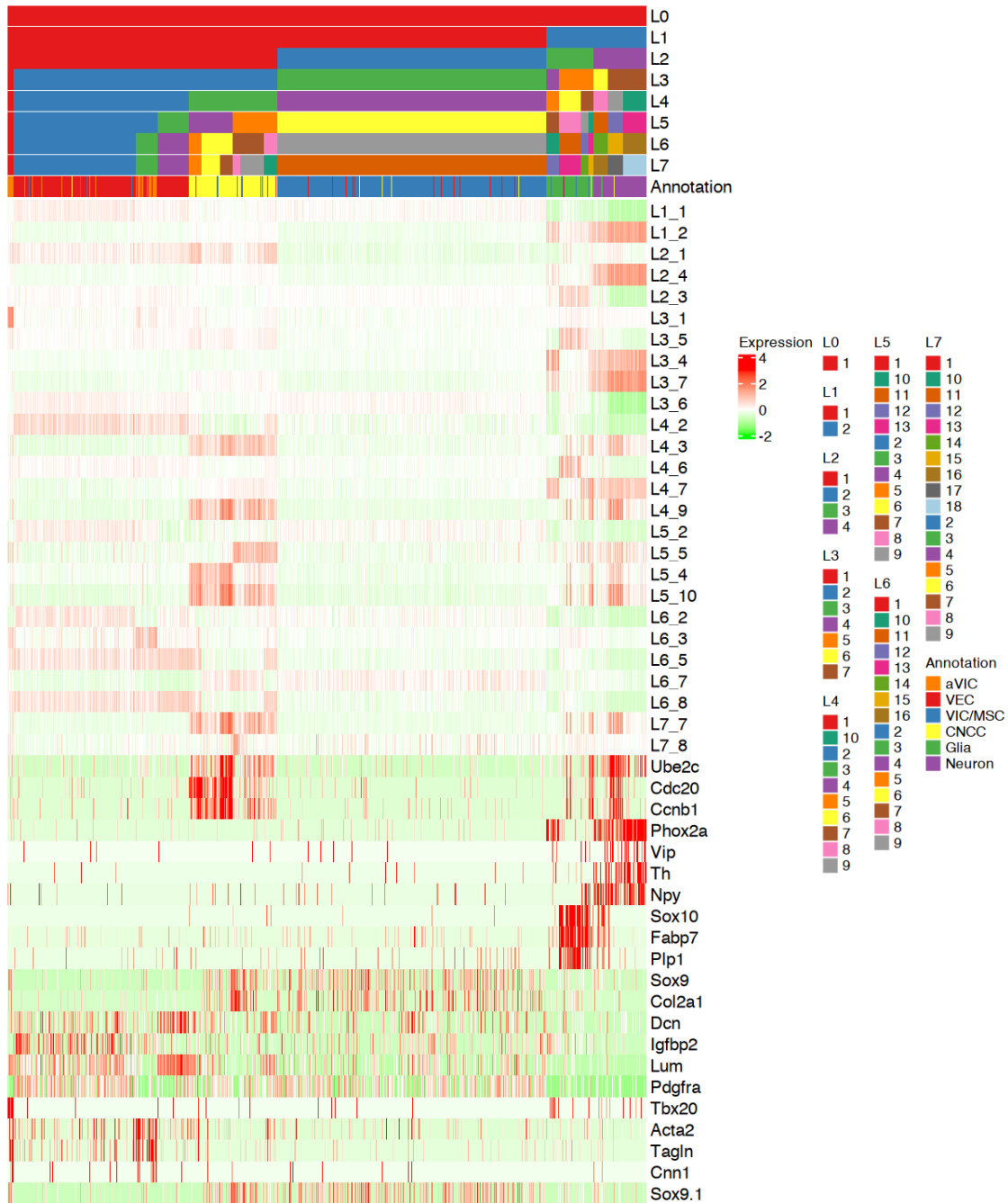
3.7.3 heatmap

```
[22]: known_markers <- c("Ube2c", "Cdc20", "Ccnb1", "Phox2a", "Vip", "Th", "Npy",  
  ↪ "Sox10", "Fabp7", "Plp1",  
    "Twist", "Sox9", "Col2a1", "Dcn", "Igfbp2", "Lum", "Pdgfra",  
  ↪ "Vcam",  
    "Tbx20", "Acta2", "Tagln", "Cnn1", "Sox9")  
known_markers <- known_markers[known_markers%in% rownames(seuset)]
```

```
[23]: options(repr.plot.width=10, repr.plot.height=12)  
DrawMarkerChainHeatmap(seuset, iterbi.cellMeta, iterbi.marker.chain,  
  ↪ compare_anno="annotation",  
    known_markers=known_markers)
```

Warning message:

"The input is a data frame, convert it to the matrix."



3.7.4 dotplot

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

Attaching package: 'dplyr'

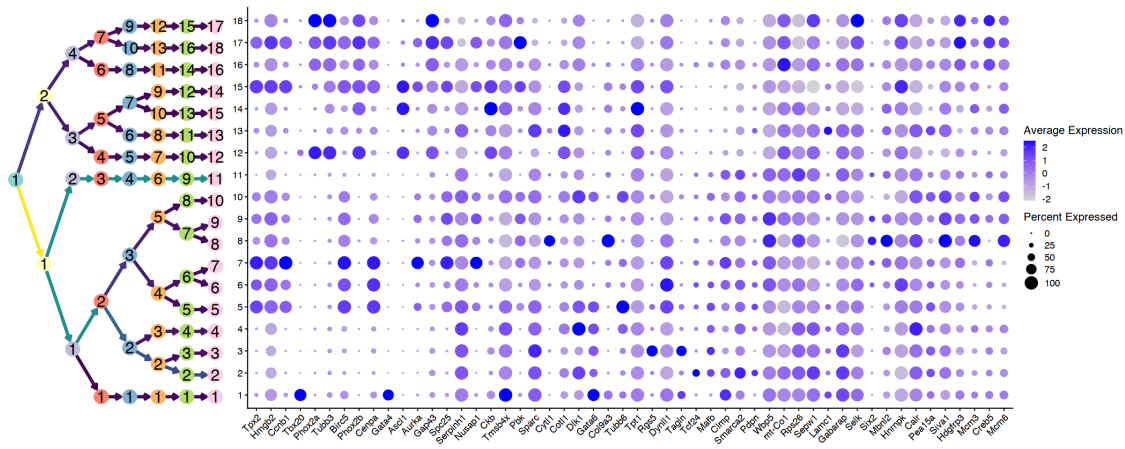
The following objects are masked from 'package:stats':

filter, lag

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

intersect, setdiff, setequal, union

```
[25]: options(repr.plot.width=20, repr.plot.height=8)
DrawMarkerChainDotplot(seuset, iterbi.marker.chain, top_n = 2, rel_heights = c(
  0.1,10,0.8))
```



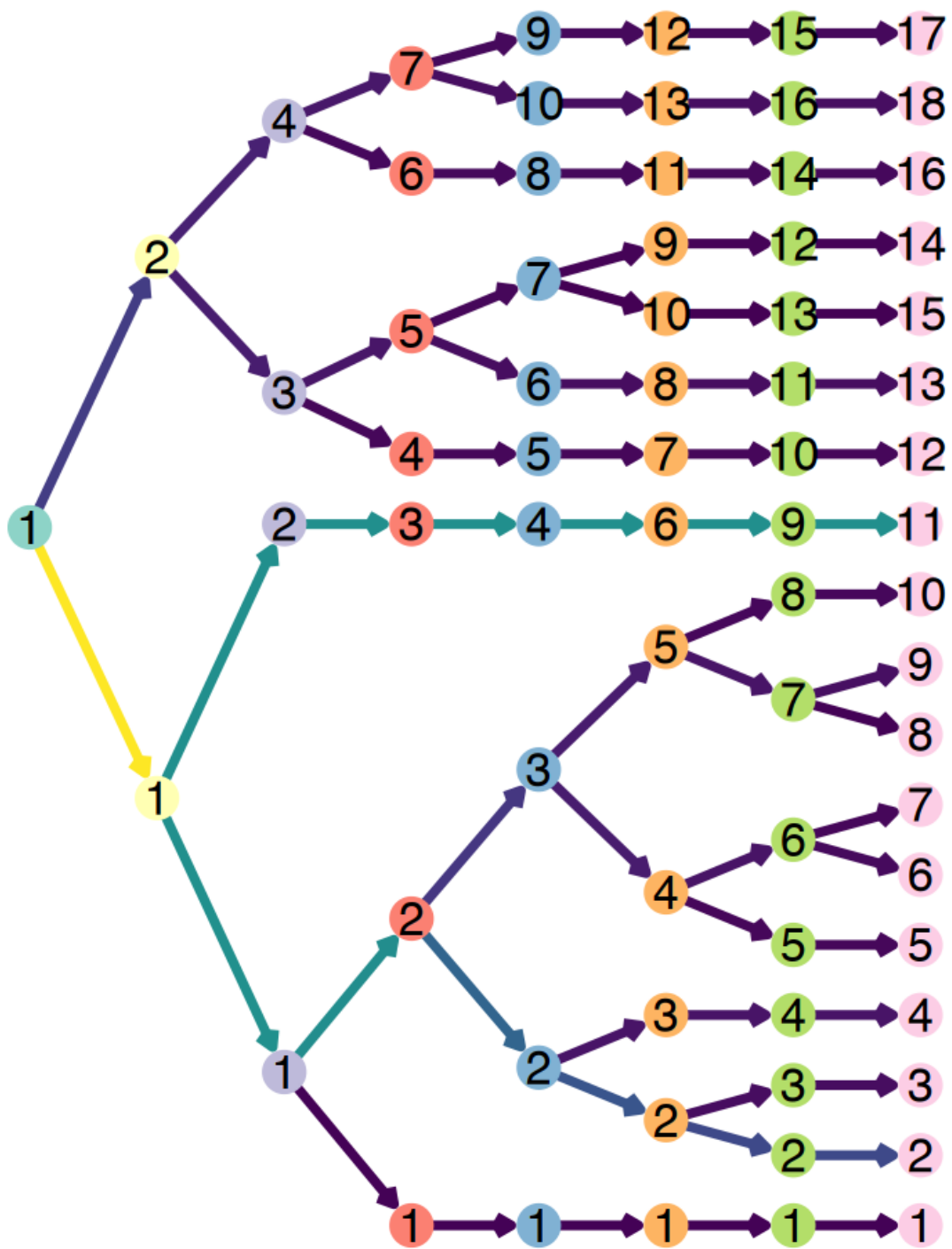
```
[27]: # filter markers
iterbi.marker.chain.uniq <- subset(iterbi.marker.chain, (cluster_pct-bcg_pct)>0.
  3 & cluster_pct>0.7 & bcg_pct<0.5)
```

```
[28]: options(repr.plot.width=20, repr.plot.height=8)
DrawMarkerChainDotplot(seuset, iterbi.marker.chain.uniq, top_n = 2, rel_heights = c(
  0.1,10,0.8))
```



3.8 module chain

```
[29]: options(repr.plot.width=6, repr.plot.height=8)
      DrawIterbiClusterTree(seuset, iterbi.cellMeta)
```

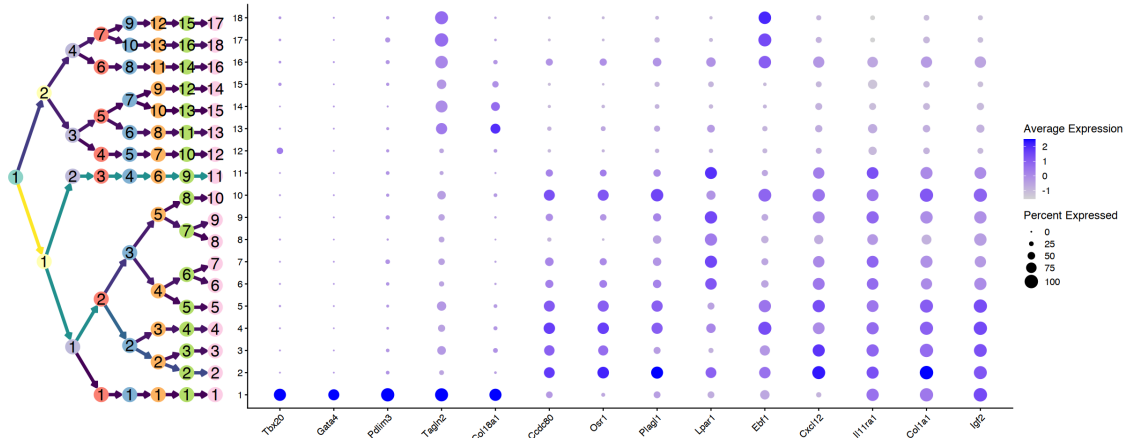


```
[40]: cluster.chain <- GetClusterChain(iterbi.cellMeta, iterbi.bifucation, "L7_1")
cluster.chain
```

```
L0\_1 'Parent_node' L1\_1 'Parent_node' L2\_1 'Parent_node' L3\_1 'Parent_node'
L4\_1 'End_node' L5\_1 'End_node' L6\_1 'End_node' L7\_1 'End_node'
```

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

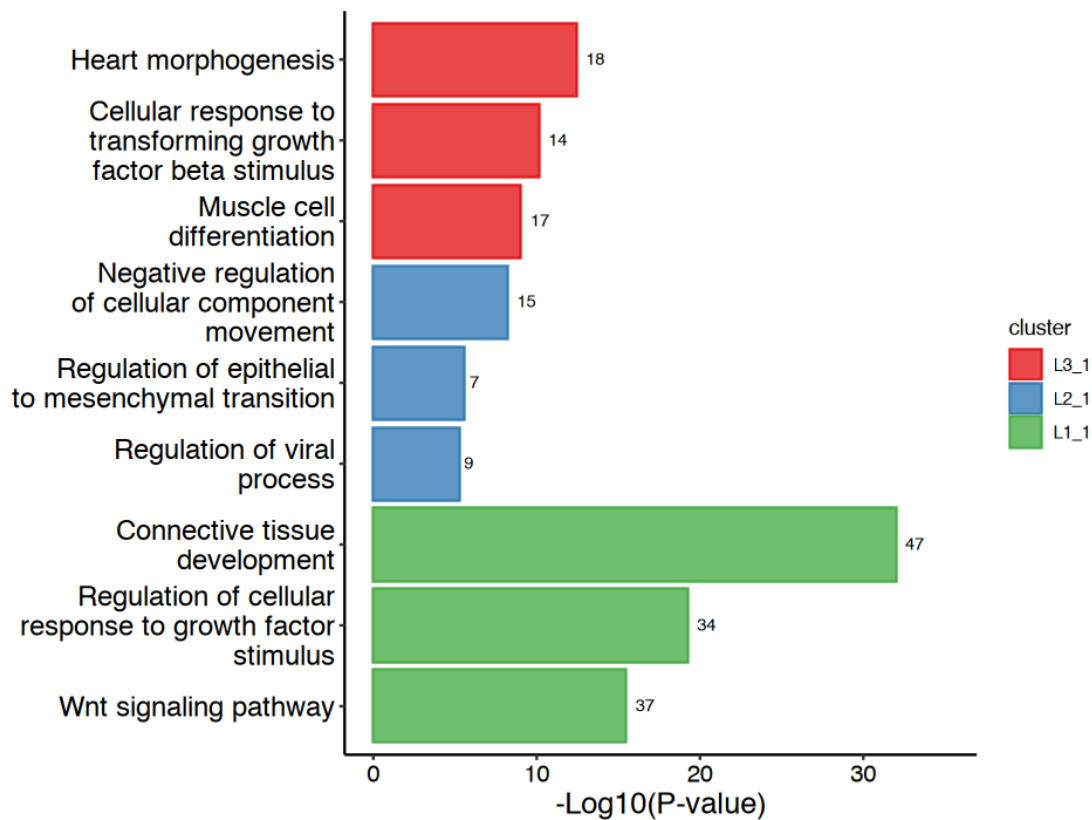
```
[42]: options(repr.plot.width=20, repr.plot.height=8)
      DrawMarkerChainDotplot(seuset, target.marker.chain, top_n = 5, rel_heights =
      ↪ c(0.1,10,0.8))
```



3.9 GO chain

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

```
Warning message in brewer.pal(12, "Set1"):
"n too large, allowed maximum for palette Set1 is 9
Returning the palette you asked for with that many colors
"
```



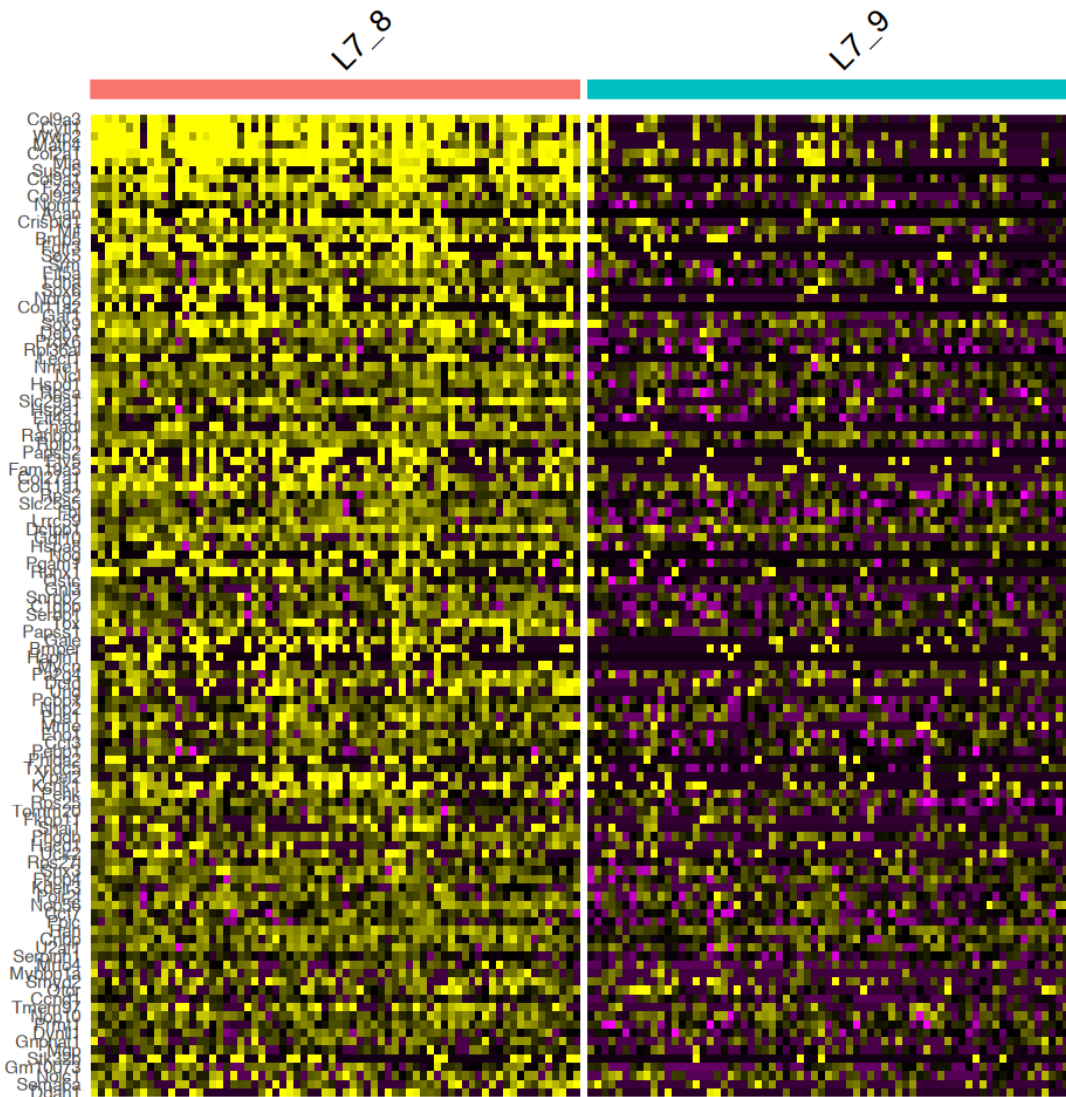
3.10 initial bifurcation event

```
[44]: # different level, different cluster
      cluster_1 <- "L7_8"
      cluster_2 <- "L7_9"

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

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

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



4 sessionInfo

```
[48]: 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] utf8_1.2.1	reticulate_1.24	tidyselect_1.1.1
[4] RSQLite_2.2.0	AnnotationDbi_1.48.0	htmlwidgets_1.5.1
[7] BiocParallel_1.20.1	grid_3.6.2	Rtsne_0.15
[10] munsell_0.5.0	codetools_0.2-16	mutoss_0.1-12
[13] ica_1.0-2	pbdZMQ_0.3-3	future_1.24.0
[16] withr_2.5.0	GOSemSim_2.12.0	colorspace_2.0-2
[19] Biobase_2.46.0	knitr_1.38	uuid_0.1-4
[22] rstudioapi_0.13	stats4_3.6.2	ROCR_1.0-7
[25] DOSE_3.12.0	gbRd_0.4-11	listenv_0.8.0
[28] labeling_0.4.2	Rdpack_0.11-1	urltools_1.7.3
[31] repr_1.1.0	mnormt_1.5-6	polyclip_1.10-0
[34] bit64_4.0.5	farver_2.1.0	rprojroot_2.0.3
[37] parallelly_1.31.0	vctrs_0.3.8	generics_0.1.2
[40] TH.data_1.0-10	xfun_0.29	R6_2.5.0
[43] doParallel_1.0.16	clue_0.3-60	graphlayouts_0.7.1
[46] rsvd_1.0.3	gridGraphics_0.5-1	fgsea_1.12.0
[49] bitops_1.0-6	cachem_1.0.6	assertthat_0.2.1
[52] scales_1.1.1	enrichplot_1.6.1	multcomp_1.4-12
[55] ggraph_2.0.5	nnet_7.3-12	gtable_0.3.0
[58] npsurv_0.4-0	org.Mm.eg.db_3.10.0	Cairo_1.5-12.2
[61] globals_0.14.0	tidygraph_1.2.0	sandwich_2.5-1
[64] rlang_1.0.1	GlobalOptions_0.1.2	splines_3.6.2
[67] lazyeval_0.2.2	europemc_0.3	checkmate_2.0.0
[70] BiocManager_1.30.12	reshape2_1.4.4	backports_1.1.9
[73] qvalue_2.18.0	Hmisc_4.4-1	clusterProfiler_3.14.3
[76] tools_3.6.2	ggplotify_0.0.5	ellipsis_0.3.2
[79] gplots_3.0.3	RColorBrewer_1.1-2	BiocGenerics_0.32.0
[82] clustree_0.4.3	ggribes_0.5.2	TFisher_0.2.0
[85] Rcpp_1.0.8.3	plyr_1.8.6	progress_1.2.2
[88] base64enc_0.1-3	purrr_0.3.4	prettyunits_1.1.1
[91] rpart_4.1-15	pbapply_1.4-3	GetoptLong_1.0.5
[94] viridis_0.6.2	cowplot_1.0.0	S4Vectors_0.24.4
[97] zoo_1.8-7	ggrepel_0.9.1	cluster_2.1.0
[100] here_1.0.1	magrittr_2.0.2	magick_2.7.1
[103] data.table_1.14.2	DO.db_2.9	circlize_0.4.14
[106] triebeard_0.3.0	lmtest_0.9-39	RANN_2.6.1
[109] mvtnorm_1.1-1	parallelDist_0.2.6	fitdistrplus_1.0-14
[112] matrixStats_0.61.0	hms_1.1.1	patchwork_1.0.1.9000

[115] lsei_1.2-0	evaluate_0.15	jpeg_0.1-8.1
[118] IRanges_2.20.2	gridExtra_2.3	shape_1.4.6
[121] compiler_3.6.2	tibble_3.1.6	KernSmooth_2.23-16
[124] ncd4_1.19	crayon_1.4.1	htmltools_0.5.2
[127] Formula_1.2-3	tidyr_1.2.0	RcppParallel_5.0.2
[130] DBI_1.1.2	tweenr_1.0.2	ComplexHeatmap_2.9.4
[133] MASS_7.3-55	Matrix_1.3-2	cli_3.1.1
[136] gdata_2.18.0	parallel_3.6.2	metap_1.3
[139] igraph_1.2.11	pkgconfig_2.0.3	sn_1.5-5
[142] rvcheck_0.1.8	numDeriv_2016.8-1.1	foreign_0.8-72
[145] IRdisplay_0.7.0	plotly_4.9.2	xml2_1.3.2
[148] foreach_1.5.1	multtest_2.42.0	bibtex_0.4.2.2
[151] stringr_1.4.0	digest_0.6.28	sctransform_0.2.1
[154] RcppAnnoy_0.0.15	tsne_0.1-3	fastmatch_1.1-0
[157] leiden_0.3.3	htmlTable_2.0.1	uwot_0.1.5
[160] gtools_3.8.2	rjson_0.2.20	lifecycle_1.0.1
[163] nlme_3.1-142	jsonlite_1.7.2	HiClimR_2.2.1
[166] viridisLite_0.4.0	fansi_0.4.2	pillar_1.7.0
[169] lattice_0.20-45	G0.db_3.10.0	fastmap_1.1.0
[172] httr_1.4.2	plotrix_3.7-8	survival_3.2-13
[175] glue_1.6.1	png_0.1-7	iterators_1.0.13
[178] bit_4.0.4	ggforce_0.3.3	stringi_1.5.3
[181] blob_1.2.1	org.Hs.eg.db_3.10.0	latticeExtra_0.6-29
[184] caTools_1.18.0	memoise_2.0.1	IRkernel_1.1
[187] irlba_2.3.3	future.apply_1.6.0	ape_5.4