

ATAC-Seq

Chris Doan

2022-11-8

```
install.packages('hdf5r', repos = "http://cran.us.r-project.org") #need to read h5 files

##  
## The downloaded binary packages are in  
## /var/folders/07/b7qwj15d1zs41kh2whcqhh0c0000gp/T//RtmpWlLkmk/downloaded_packages

setRepositories(ind=1:3) # needed to automatically install Bioconductor dependencies
install.packages("Signac", repos = "http://cran.us.r-project.org") #seurat addon for analyzing chromati

##  
## The downloaded binary packages are in  
## /var/folders/07/b7qwj15d1zs41kh2whcqhh0c0000gp/T//RtmpWlLkmk/downloaded_packages

install.packages('Seurat',  
                 repos = "http://cran.us.r-project.org")

## Warning in download.file(url, destfile, method, mode = "wb", ...): URL 'http://  
## cran.us.r-project.org/bin/macosx/contrib/4.2/Seurat_4.2.1.tgz': status was  
## 'Couldn't resolve host name'

## Error in download.file(url, destfile, method, mode = "wb", ...):  
##   cannot open URL 'http://cran.us.r-project.org/bin/macosx/contrib/4.2/Seurat_4.2.1.tgz'

## Warning in download.packages(pkgs, destdir = tmpd, available = available, :  
## download of package 'Seurat' failed

library(Signac)
library(Seurat)

## Attaching SeuratObject

counts <- Read10X_h5(filename = "ATAC/GSM5723631_Young_HSC_filtered_peak_bc_matrix.h5")

meta <- read.csv(  
  file = 'ATAC/GSM5723631_Young_HSC_singlecell.csv.gz',  
  header = TRUE,  
  row.names = 1)
```

```

chrom_assay <- CreateChromatinAssay(
  counts = counts,
  sep = c(":", "-"),
  genome = 'mm10',
  fragments = './ATAC/GSM5723631_Young_HSC_fragments.tsv.gz',
  min.cells = 10,
  min.features = 200
)

## Computing hash

## Loading required package: BiocGenerics

##
## Attaching package: 'BiocGenerics'

## 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.max, which.min

## Loading required package: S4Vectors

## Loading required package: stats4

##
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:base':
## 
##     expand.grid, I, unname

data <- CreateSeuratObject(
  counts = chrom_assay,
  assay = "peaks",
  meta.data = meta
)

## Warning in CreateSeuratObject.Assay(counts = chrom_assay, assay = "peaks", :
## Some cells in meta.data not present in provided counts matrix.

## Warning: Keys should be one or more alphanumeric characters followed by an
## underscore, setting key from peaks to peaks_

```

```
#data[]]
```

EnsDb.Hsapiens.v86 for human

```
if (!require("BiocManager", quietly = TRUE))
  install.packages("BiocManager")

## Bioconductor version '3.15' is out-of-date; the current release version '3.16'
##   is available with R version '4.2'; see https://bioconductor.org/install

BiocManager::install("EnsDb.Mmusculus.v79")

## 'getOption("repos")' replaces Bioconductor standard repositories, see
## '?repositories' for details
##
## replacement repositories:
##   BioCsoft: https://bioconductor.org/packages/3.15/bioc
##   BioCann: https://bioconductor.org/packages/3.15/data/annotation

## Bioconductor version 3.15 (BiocManager 1.30.19), R 4.2.1 (2022-06-23)

## Warning: package(s) not installed when version(s) same as or greater than current; use
##   'force = TRUE' to re-install: 'EnsDb.Mmusculus.v79'

## Old packages: 'bit', 'data.table', 'evaluate', 'ggfun', 'ggpubr', 'harmony',
##   'markdown', 'Matrix', 'modelr', 'pbapply', 'pkgload', 'plotly', 'plyr',
##   'rbibutils', 'RcppArmadillo', 'RcppEigen', 'rmarkdown', 'roxygen2',
##   'rstatix', 'scCATCH', 'sp', 'spatstat.explore', 'vctrs', 'xfun'

BiocManager::install("GenomeInfoDb") #translation between chromosome names

## 'getOption("repos")' replaces Bioconductor standard repositories, see
## '?repositories' for details
##
## replacement repositories:
##   BioCsoft: https://bioconductor.org/packages/3.15/bioc
##   BioCann: https://bioconductor.org/packages/3.15/data/annotation
##
## Bioconductor version 3.15 (BiocManager 1.30.19), R 4.2.1 (2022-06-23)

## Warning: package(s) not installed when version(s) same as or greater than current; use
##   'force = TRUE' to re-install: 'GenomeInfoDb'

## Old packages: 'bit', 'data.table', 'evaluate', 'ggfun', 'ggpubr', 'harmony',
##   'markdown', 'Matrix', 'modelr', 'pbapply', 'pkgload', 'plotly', 'plyr',
##   'rbibutils', 'RcppArmadillo', 'RcppEigen', 'rmarkdown', 'roxygen2',
##   'rstatix', 'scCATCH', 'sp', 'spatstat.explore', 'vctrs', 'xfun'
```

```

BiocManager::install("biovizBase")

## 'getOption("repos")' replaces Bioconductor standard repositories, see
## '?repositories' for details
##
## replacement repositories:
##   BioCsoft: https://bioconductor.org/packages/3.15/bioc
##   BioCann: https://bioconductor.org/packages/3.15/data/annotation
##
## Bioconductor version 3.15 (BiocManager 1.30.19), R 4.2.1 (2022-06-23)

## Warning: package(s) not installed when version(s) same as or greater than current; use
##   'force = TRUE' to re-install: 'biovizBase'

## Old packages: 'bit', 'data.table', 'evaluate', 'ggfun', 'ggpubr', 'harmony',
##   'markdown', 'Matrix', 'modelr', 'pbapply', 'pkgload', 'plotly', 'plyr',
##   'rbibutils', 'RcppArmadillo', 'RcppEigen', 'rmarkdown', 'roxygen2',
##   'rstatix', 'scCATCH', 'sp', 'spatstat.explore', 'vctrs', 'xfun'

library(GenomeInfoDb)
library(EnsDb.Mmusculus.v79)

## Loading required package: ensemblldb

## Loading required package: GenomicRanges

## Loading required package: GenomicFeatures

## Loading required package: AnnotationDbi

## Loading required package: Biobase

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

## Loading required package: AnnotationFilter

##
## Attaching package: 'ensemblldb'

## The following object is masked from 'package:stats':
##   filter

```

```
annotations <- GetGRangesFromEnsDb(ensdb = EnsDb.Mmusculus.v79)
```

```

## Warning in .Seqinfo.mergexy(x, y): The 2 combined objects have no sequence levels in common. (Use
##   suppressWarnings() to suppress this warning.)

## Warning in .Seqinfo.mergexy(x, y): The 2 combined objects have no sequence levels in common. (Use
##   suppressWarnings() to suppress this warning.)

## Warning in .Seqinfo.mergexy(x, y): The 2 combined objects have no sequence levels in common. (Use
##   suppressWarnings() to suppress this warning.)

## Warning in .Seqinfo.mergexy(x, y): The 2 combined objects have no sequence levels in common. (Use
##   suppressWarnings() to suppress this warning.)

seqlevelsStyle(annotations) <- 'UCSC'

Annotation(data) <- annotations

data <- NucleosomeSignal(object = data) #fragment ratio 147-294: <147

data <- TSSEnrichment(object = data, fast = FALSE)

## Extracting TSS positions

## Finding + strand cut sites

## Finding - strand cut sites

## Computing mean insertion frequency in flanking regions

## Normalizing TSS score

data$blacklist_ratio <- data$blacklist_region_fragments / data$peak_region_fragments

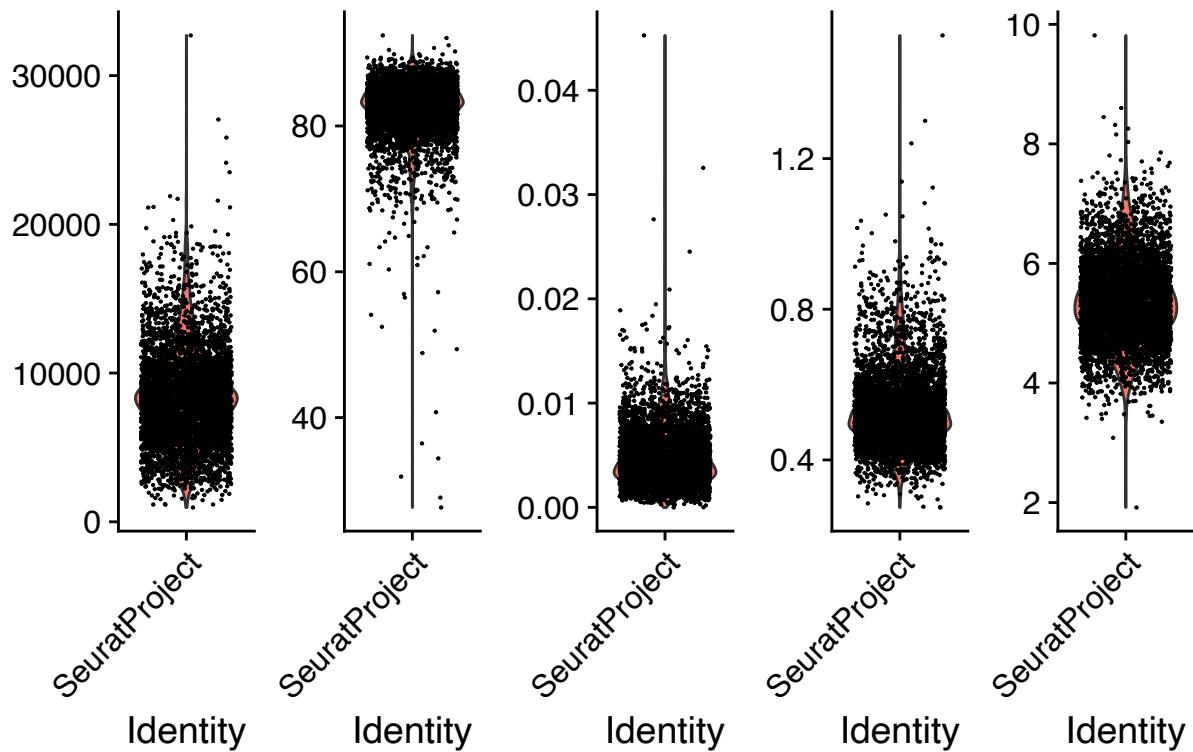
#data[]]

data$pct_reads_in_peaks <- data$peak_region_fragments / data$passed_filters * 100

VlnPlot(
  object = data,
  features = c('peak_region_fragments', 'pct_reads_in_peaks',
              'blacklist_ratio', 'nucleosome_signal', 'TSS.enrichment'),
  pt.size = 0.1,
  ncol = 5
)

```

peak_region_fragments_in_blacklist_ratio_nucleosome_TSS_enrichment



could do this....

```

data <- subset(
  x = data,
  subset = peak_region_fragments > 3000 &
    peak_region_fragments < 20000 &
    pct_reads_in_peaks > 15 &
    blacklist_ratio < 0.05 &
    nucleosome_signal < 4 &
    TSS.enrichment > 2
)

low_prf <- quantile(data[["peak_region_fragments"]]$peak_region_fragments, probs = 0.02)
high_prf <- quantile(data[["peak_region_fragments"]]$peak_region_fragments, probs = 0.98)
low_prp <- quantile(data[["pct_reads_in_peaks"]]$pct_reads_in_peaks, probs = 0.02)

high_blr <- quantile(data[["blacklist_ratio"]]$blacklist_ratio, probs = 0.98)

high_ns <- quantile(data[["nucleosome_signal"]]$nucleosome_signal, probs = 0.98)

low_ts <- quantile(data[["TSS.enrichment"]]$TSS.enrichment, probs = 0.02)

print(low_prf)

##      2%
## 3531.7

```

```

print(hig_prf)

##      98%
## 15713

print(low_prp)

##      2%
## 72.68298

print(high_blr)

##      98%
## 0.01136435

print(hig_ns)

##      98%
## 0.8160401

print(low_ts)

##      2%
## 4.098172

data <- subset(
  x = data,
  subset = peak_region_fragments > low_prf &
  peak_region_fragments < hig_prf &
  pct_reads_in_peaks > low_prp &
  blacklist_ratio < high_blr &
  nucleosome_signal < hig_ns &
  TSS.enrichment > low_ts
)

```

#data

Normalization, dimension reduction

```

data <- RunTFIDF(data)

## Performing TF-IDF normalization

data <- FindTopFeatures(data, min.cutoff = 'q0')
#data

```

```
data <- RunSVD(data)

## Running SVD

## Scaling cell embeddings

DepthCor(data)

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9

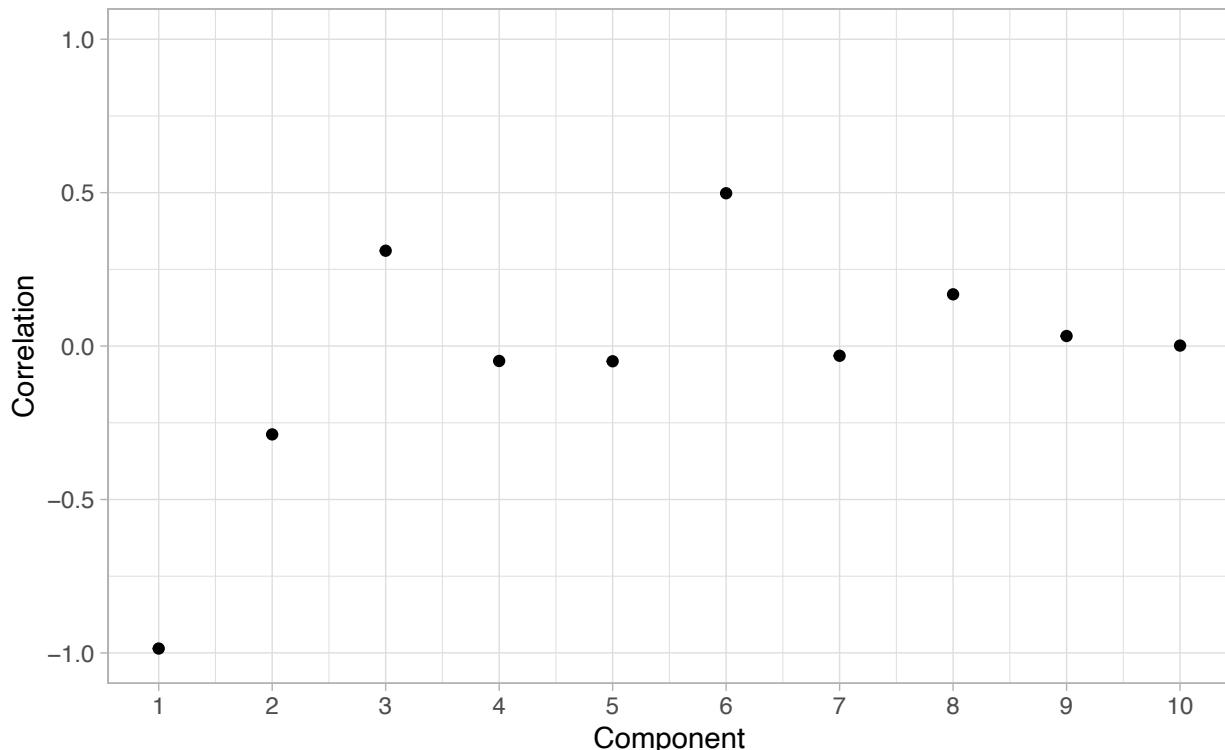
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## width unknown for character 0x9
```

Correlation between depth and reduced dimension components

Assay: peaks Reduction: Isi



```
data <- RunUMAP(object = data, reduction = 'lsi', dims = 2:30)

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

## 20:32:46 UMAP embedding parameters a = 0.9922 b = 1.112

## 20:32:46 Read 4363 rows and found 29 numeric columns

## 20:32:46 Using Annoy for neighbor search, n_neighbors = 30

## 20:32:46 Building Annoy index with metric = cosine, n_trees = 50

## 0%   10    20    30    40    50    60    70    80    90   100%
## [----|----|----|----|----|----|----|----|----|----|
## ****|*****|*****|*****|*****|*****|*****|*****|*****|*****|
```

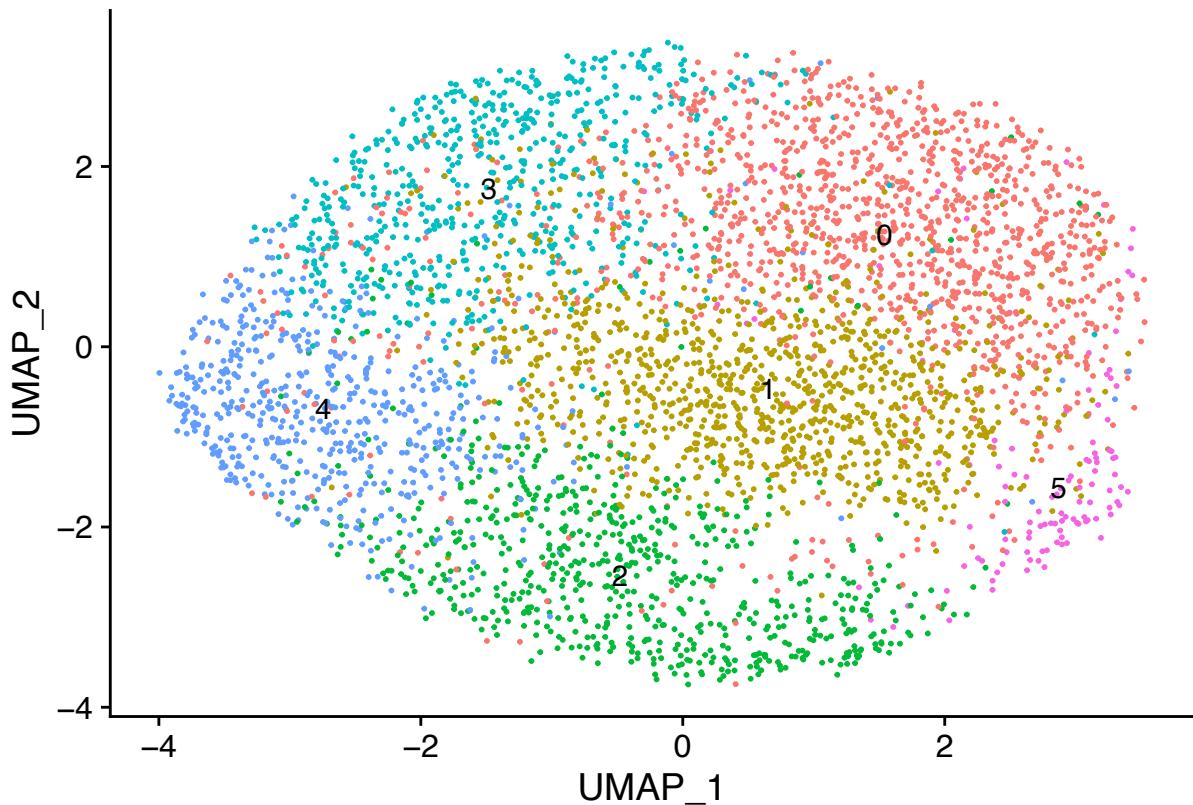
```

data <- FindNeighbors(object = data, reduction = 'lsi', dims = 2:30)

## Computing nearest neighbor graph
## Computing SNN

data <- FindClusters(object = data, verbose = FALSE, algorithm = 3)
DimPlot(object = data, label = TRUE) + NoLegend()

```



Multiple samples....

```

import_atac <- function(count_path, meta_path, fragment_path){
  counts <- Read10X_h5(filename = count_path)

  meta <- read.csv(
    file = meta_path,
    header = TRUE,
    row.names = 1)

  chrom_assay <- CreateChromatinAssay(
    counts = counts,
    sep = c(":", "-"),
    genome = 'mm10',
    fragments = fragment_path,
    min.cells = 10,
    min.features = 200
  )

```

```

)

data <- CreateSeuratObject(
  counts = chrom_assay,
  assay = "peaks",
  meta.data = meta
)

Annotation(data) <- annotations

data <- NucleosomeSignal(object = data) #fragment ratio 147-294: <147 --- mononucleosome:nucleosome

data <- TSSEnrichment(object = data, fast = FALSE)

data$blacklist_ratio <- data$blacklist_region_fragments / data$peak_region_fragments

data$pct_reads_in_peaks <- data$peak_region_fragments / data$passed_filters * 100

low_prf <- quantile(data[["peak_region_fragments"]]$peak_region_fragments, probs = 0.02)
high_prf <- quantile(data[["peak_region_fragments"]]$peak_region_fragments, probs = 0.98)
low_prr <- quantile(data[["pct_reads_in_peaks"]]$pct_reads_in_peaks, probs = 0.02)

high_blr <- quantile(data[["blacklist_ratio"]]$blacklist_ratio, probs = 0.98)

high_ns <- quantile(data[["nucleosome_signal"]]$nucleosome_signal, probs = 0.98)

low_ts <- quantile(data[["TSS.enrichment"]]$TSS.enrichment, probs = 0.02)

data <- subset(
  x = data,
  subset = peak_region_fragments > low_prf &
    peak_region_fragments < high_prf &
    pct_reads_in_peaks > low_prr &
    blacklist_ratio < high_blr &
    nucleosome_signal < high_ns &
    TSS.enrichment > low_ts
)

#data <- RunTFIDF(data)
#data <- FindTopFeatures(data, min.cutoff = 'q0')
#data <- RunSVD(data)

return(data)
}

young <- import_atac("ATAC/GSM5723631_Young_HSC_filtered_peak_bc_matrix.h5",
  'ATAC/GSM5723631_Young_HSC_singlecell.csv.gz',
  './ATAC/GSM5723631_Young_HSC_fragments.tsv.gz')

```

```

## Computing hash

## Warning in CreateSeuratObject.Assay(counts = chrom_assay, assay = "peaks", :
## Some cells in meta.data not present in provided counts matrix.

## Warning: Keys should be one or more alphanumeric characters followed by an
## underscore, setting key from peaks to peaks_

## Extracting TSS positions

## Finding + strand cut sites

## Finding - strand cut sites

## Computing mean insertion frequency in flanking regions

## Normalizing TSS score

old <- import_atac("ATAC/GSM5723632_Aged_HSC_filtered_peak_bc_matrix.h5",
  'ATAC/GSM5723632_Aged_HSC_singlecell.csv.gz',
  './ATAC/GSM5723632_Aged_HSC_fragments.tsv.gz')

## Computing hash

## Warning in CreateSeuratObject.Assay(counts = chrom_assay, assay = "peaks", :
## Some cells in meta.data not present in provided counts matrix.

## Warning in CreateSeuratObject.Assay(counts = chrom_assay, assay = "peaks", :
## Keys should be one or more alphanumeric characters followed by an underscore,
## setting key from peaks to peaks_

## Extracting TSS positions

## Finding + strand cut sites

## Finding - strand cut sites

## Computing mean insertion frequency in flanking regions

## Normalizing TSS score

young$dataset <- "young"
old$dataset <- "old"

data <- merge(young, old)

## Warning in CheckDuplicateCellNames(object.list = objects): Some cell names are
## duplicated across objects provided. Renaming to enforce unique cell names.

```

```
##  
## Binding matrix rows  
  
#data  
  
data <- FindTopFeatures(data, min.cutoff = 'q0')  
data <- RunTFIDF(data)  
  
## Performing TF-IDF normalization  
  
data <- RunSVD(data)  
  
## Running SVD  
  
## Scaling cell embeddings  
  
#data  
  
data <- RunUMAP(object = data, reduction = 'lsi', dims = 2:30)  
  
## 23:12:34 UMAP embedding parameters a = 0.9922 b = 1.112  
  
## 23:12:34 Read 7792 rows and found 29 numeric columns  
  
## 23:12:34 Using Annoy for neighbor search, n_neighbors = 30  
  
## 23:12:34 Building Annoy index with metric = cosine, n_trees = 50  
  
## 0%   10    20    30    40    50    60    70    80    90    100%  
  
## [----|----|----|----|----|----|----|----|----|  
  
## *****|*****|*****|*****|*****|*****|*****|*****|*****|  
## 23:12:35 Writing NN index file to temp file /var/folders/07/b7qwj15d1zs41kh2whcqhh0c0000gp/T//RtmpWl1  
## 23:12:35 Searching Annoy index using 1 thread, search_k = 3000  
## 23:12:36 Annoy recall = 100%  
## 23:12:46 Commencing smooth kNN distance calibration using 1 thread with target n_neighbors = 30  
## 23:12:57 Initializing from normalized Laplacian + noise (using irlba)  
## 23:12:57 Commencing optimization for 500 epochs, with 295494 positive edges  
## 23:13:13 Optimization finished
```

```

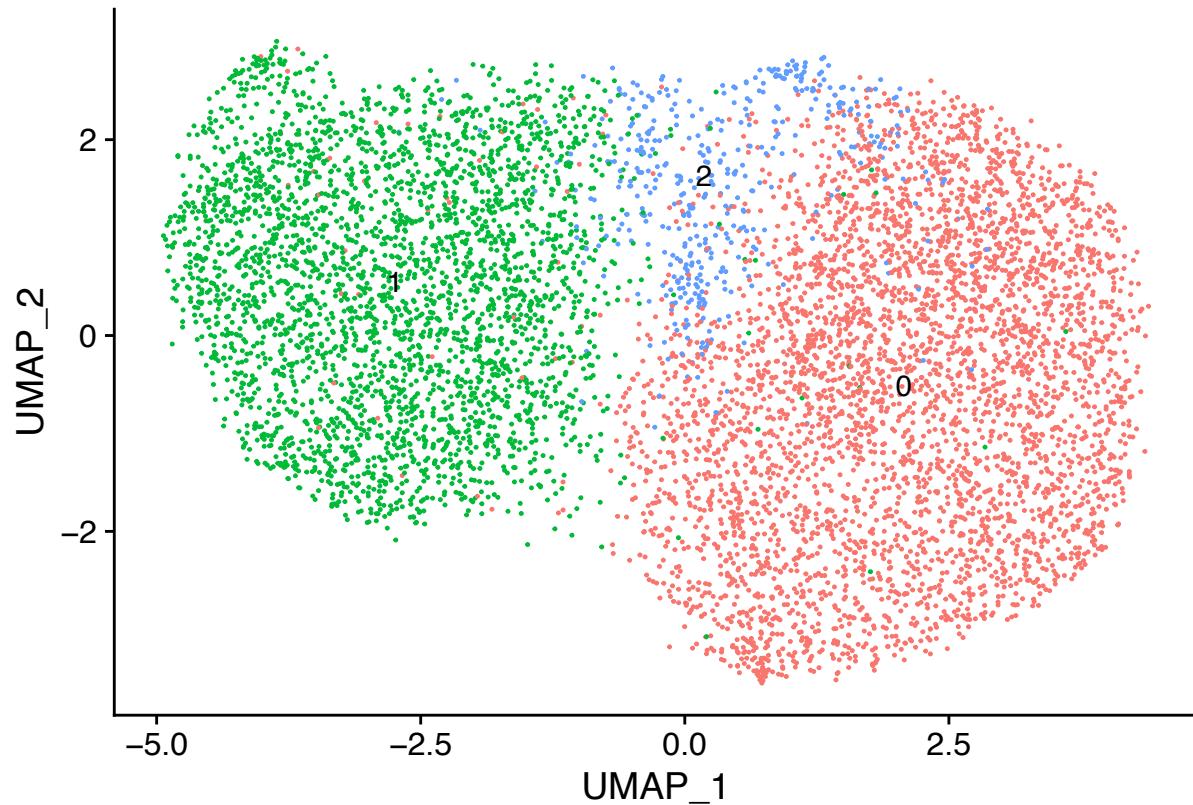
data <- FindNeighbors(object = data, reduction = 'lsi', dims = 2:30)

## Computing nearest neighbor graph
## Computing SNN

data <- FindClusters(object = data, verbose = FALSE, algorithm = 3, resolution = .4)

DimPlot(object = data, label = TRUE) + NoLegend()

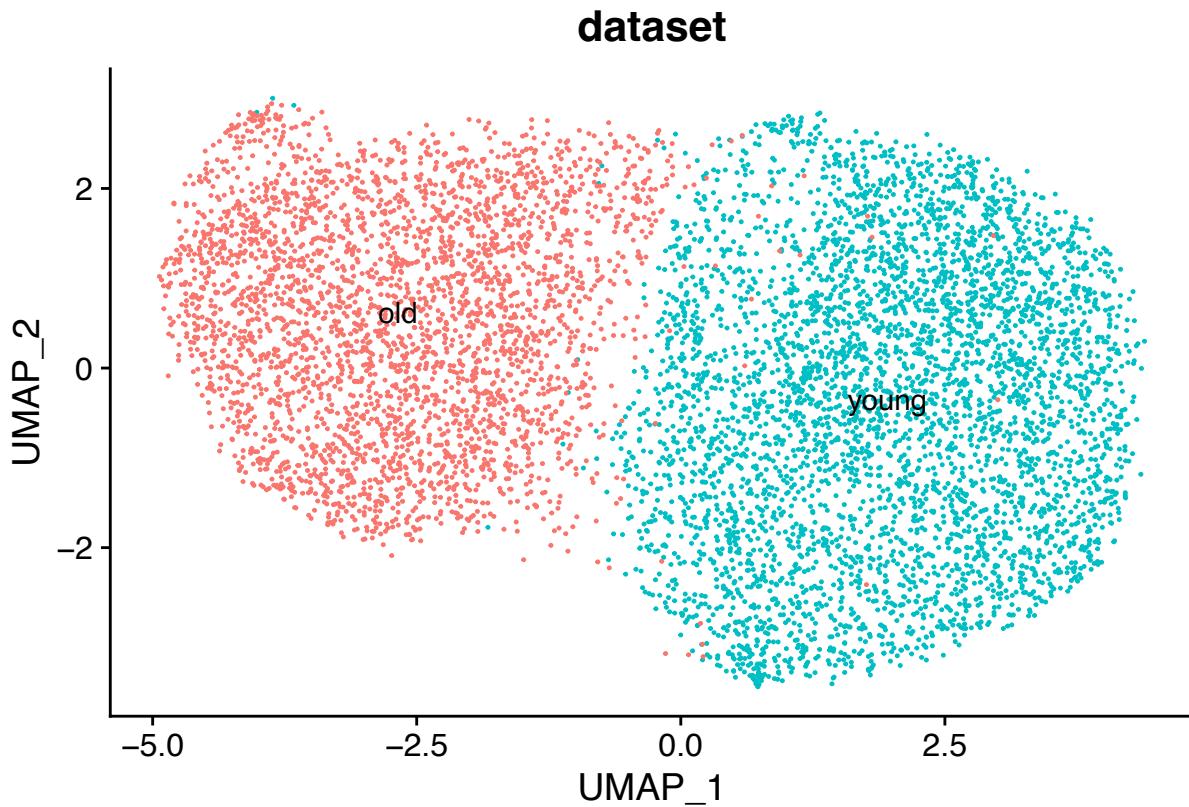
```



```

DimPlot(object = data, label = TRUE, group.by = "dataset") + NoLegend()

```



Data analysis

```

gene.activities <- GeneActivity(data)

## Extracting gene coordinates

## Extracting reads overlapping genomic regions
## Extracting reads overlapping genomic regions

data[['RNA']] <- CreateAssayObject(counts = gene.activities)

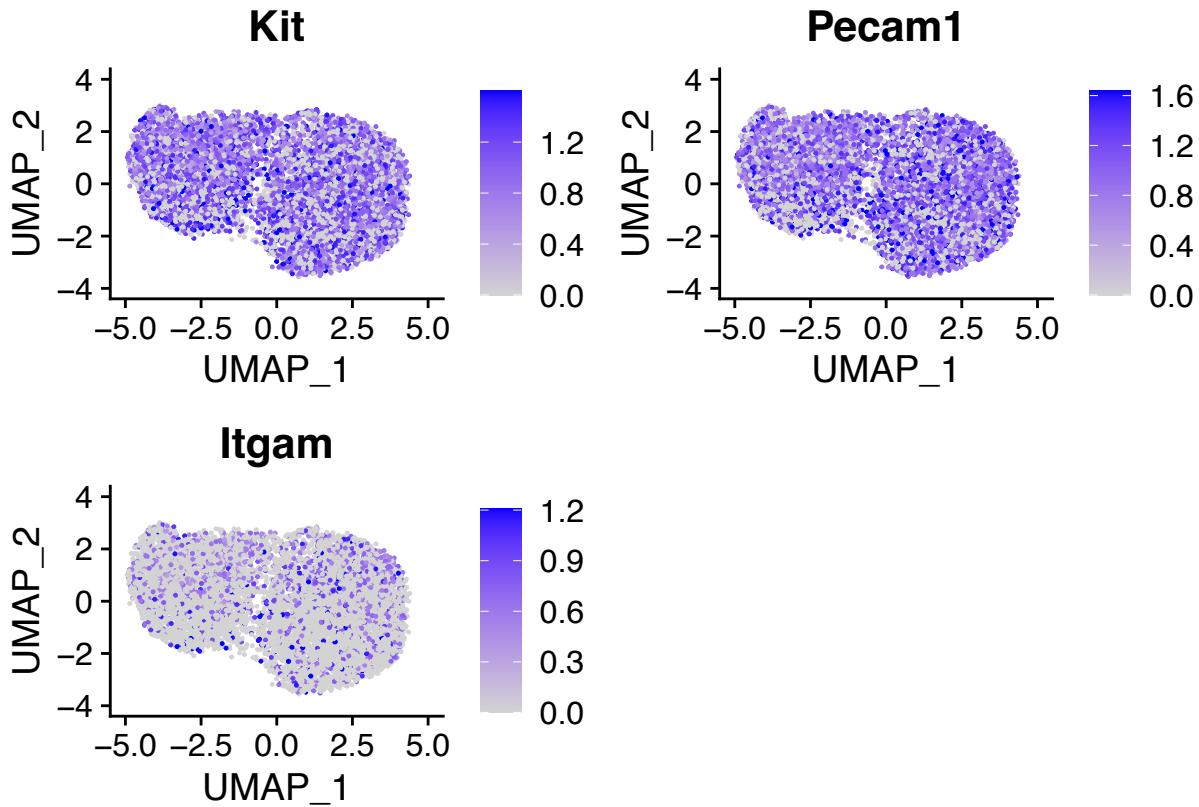
data <- NormalizeData(
  object = data,
  assay = 'RNA',
  normalization.method = 'LogNormalize',
  scale.factor = median(data$nCount_RNA)
)

#data[['RNA']]

DefaultAssay(data) <- 'RNA'

FeaturePlot(
  object = data,
  features = c('Kit', 'Pecam1', 'Itgam'),
  max.cutoff = 'q95'
)

```



```
DefaultAssay(data) <- 'peaks'

da_peaks <- FindMarkers(
  object = data,
  ident.1 = rownames(data)[[data$dataset == "old",]],
  ident.2 = rownames(data)[[data$dataset == "young",]],
  min.pct = 0.05,
  test.use = 'LR',
  latent.vars = 'peak_region.fragments'
)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
da_peaks
```

	p_val	avg_log2FC	pct.1	pct.2	p_val_adj
## chr2-107231278-107232435	1.094643e-134	11.4364859	0.112	0.000	1.941448e-129
## chr19-6295925-6310866	7.278839e-133	-0.9250820	0.392	0.595	1.290968e-127
## chr3-99112422-99115022	5.667198e-128	3.6579187	0.165	0.012	1.005129e-122
## chr17-58643214-58645092	1.818786e-126	3.7934305	0.155	0.010	3.225782e-121
## chr3-57509149-57513454	1.996828e-118	3.5195003	0.154	0.013	3.541555e-113
## chr16-23841670-23845601	3.564753e-109	2.6104226	0.194	0.031	6.322410e-104
## chr11-118602840-118605327	6.046991e-103	2.6511780	0.181	0.028	1.072488e-97
## chr7-70273640-70275132	2.976969e-101	5.2516193	0.103	0.003	5.279923e-96
## chr5-127537049-127538119	1.099834e-97	4.7811699	0.104	0.004	1.950654e-92
## chr19-20599880-20607776	1.239417e-95	1.8682118	0.253	0.069	2.198217e-90
## chr18-38689783-38691471	2.806818e-88	3.3677074	0.121	0.012	4.978145e-83

## chr18-70100869-70102635	3.434534e-01	0.2925942	0.058	0.045	1.000000e+00
## chr4-97673429-97675079	3.444001e-01	0.2525241	0.053	0.043	1.000000e+00
## chr7-125522127-125523994	3.488605e-01	0.2660844	0.051	0.039	1.000000e+00
## chr8-122750781-122753323	3.516711e-01	0.2503590	0.062	0.048	1.000000e+00
## chr8-126698464-126700253	3.528003e-01	0.2506856	0.060	0.048	1.000000e+00
## chr14-96961817-96963629	3.577643e-01	0.2518460	0.053	0.043	1.000000e+00
## chr5-118649190-118653881	3.722990e-01	0.2516280	0.070	0.056	1.000000e+00
## chr3-122634204-122636017	3.726035e-01	0.2575202	0.056	0.046	1.000000e+00
## chr8-12278784-12280440	3.759894e-01	0.2593295	0.057	0.044	1.000000e+00
## chr7-140057313-140059317	3.798408e-01	0.2734907	0.052	0.041	1.000000e+00
## chr16-31908424-31911509	3.830976e-01	0.2751210	0.058	0.046	1.000000e+00
## chr4-3503229-3506134	3.834660e-01	0.2558751	0.052	0.041	1.000000e+00
## chr2-17363627-17366172	3.873807e-01	0.2741811	0.050	0.038	1.000000e+00
## chr11-34395466-34396947	4.262795e-01	0.2583720	0.051	0.039	1.000000e+00
## chr9-99080682-99082185	4.684080e-01	0.2862183	0.056	0.043	1.000000e+00
##		closest_gene	distance		
## chr2-107231278-107232435		Kcna4	58203		
## chr19-6295925-6310866		Ehd1	0		
## chr3-99112422-99115022		Wars2	26045		
## chr17-58643214-58645092		Cntnap5c	232871		
## chr3-57509149-57513454		Wwtr1	0		
## chr16-23841670-23845601		Sst	43979		
## chr11-118602840-118605327		Rbfox3	0		
## chr7-70273640-70275132		Gm29327	23863		
## chr5-127537049-127538119		Tmem132c	0		
## chr19-20599880-20607776		Aldh1a1	0		
## chr18-38689783-38691471		Arhgap26	0		
## chr6-81737474-81739083	1700009C05Rik		161380		
## chr14-18460174-18464129		Ube2e2	109445		
## chr1-15677637-15680698		Kcnb2	0		
## chr2-160224927-160226517		Gm826	84875		
## chr5-26903642-26905903		Dpp6	0		
## chr9-30313064-30315735		Snx19	111593		
## chr13-69592861-69597041		Srd5a1	0		
## chrX-83584530-83604474		Dmd	0		
## chr11-30997947-31002306		Asb3	0		
## chr18-84203116-84205110		Zfp407	0		
## chr4-124684960-124688440		Utp11l	0		
## chr1-51287923-51293408		Sdpr	0		
## chr10-86876096-86880365		Stab2	0		
## chr10-69611392-69612675		Ank3	0		
## chr3-57554145-57558727		Wwtr1	0		
## chr11-66166898-66168976		Dnah9	0		
## chr15-101397948-101398984		Krt7	12058		
## chr3-67601361-67602314		Mfsd1	0		
## chr13-48442459-48445612		Zfp169	42034		
## chr15-41823894-41826428		Oxr1	0		
## chr7-72728445-72729718		Gm28744	60088		
## chr16-6320518-6322339		Rbfox1	486882		
## chr14-73086275-73096470		Rcbtb2	26566		
## chr13-79442519-79443841		Gm29318	144276		
## chr19-6290930-6294154		Ehd1	0		
## chr10-74147864-74148629		Pcdh15	0		
## chr7-63315796-63317114	4930554H23Rik		27575		

## chr15-25523570-25526172	Myo10	96352
## chr12-110973515-110976526	Ankrd9	0
## chr15-42552345-42559570	Angpt1	0
## chr7-66954652-66956694	Adamts17	0
## chr12-12171398-12172937	Gm26171	64530
## chr10-62049514-62050787	Gm5424	20227
## chr10-42944043-42945019	Scml4	0
## chr8-35764874-35766750	Gm22216	17311
## chr1-7146158-7147685	Pcmtd1	0
## chr7-99864800-99866726	Xrra1	0
## chr14-119260108-119261223	Hs6st3	0
## chr19-27123535-27124917	Vldlr	91566
## chr5-139748090-139751704	Ints1	0
## chr2-101600011-101601796	B230118H07Rik	0
## chr6-8734988-8736794	Ica1	0
## chr13-107325593-107326477	Apoo-ps	87387
## chr13-51885648-51888129	Gadd45g	37179
## chr3-27369924-27372749	Ghsr	0
## chr16-90124758-90127407	Sod1	93334
## chr6-100501037-100503972	1700049E22Rik	23427
## chrX-20816547-20817892	Araf	0
## chr7-43880973-43884212	Klk4	0
## chr17-80165486-80166393	Galm	0
## chr4-47521717-47522812	Sec61b	38474
## chr9-37175625-37177697	Pknox2	28302
## chr4-94900553-94902337	Eqtn	4929
## chr9-63644894-63649084	Smad3	0
## chr8-111543112-111548829	Znrf1	0
## chr12-87100889-87105369	Ngb	0
## chr18-15344767-15347776	Aqp4	41617
## chr13-21262754-21265333	Gpx5	21095
## chr14-59340809-59342025	Phf11b	0
## chr4-117002239-117003415	Hectd3	0
## chr8-12735009-12737228	Gm15348	15881
## chr15-27788284-27790742	Trio	0
## chr18-61012292-61015624	Slc6a7	0
## chr12-33182253-33184004	Atxn7l1	0
## chr5-138857009-138857882	Gm5294	35389
## chr3-38257067-38258774	Ankrd50	190486
## chr18-43206732-43208412	Stk32a	0
## chr19-53773190-53775431	Rbm20	0
## chr6-55789387-55790757	Ccdc129	46137
## chrX-102074332-102075703	Nhs12	0
## chr10-21099322-21101582	Ahi1	18893
## chr12-112000637-112001792	Tdrd9	0
## chr10-31329611-31331993	Tpd5211	386
## chr17-37196375-37198383	Olf94	0
## chr9-58466944-58468147	4930461G14Rik	0
## chr5-121947142-121949877	Cux2	0
## chr12-79049947-79051480	Plekhh1	0
## chr8-8333886-8335694	Efnb2	281739
## chr4-154105014-154106246	Trp73	0
## chr15-99259099-99260791	1700120C14Rik	0
## chr6-91280795-91281893	Fbln2	8254

```

## chr8-105464161-105465728          Lrrc36      74
## chr12-39665365-39666083         Arl4a     339363
## chr3-11456184-11457542        RP23-17107.1 365686
## chr12-108326550-108329581       Hhip11      0
## chr17-78454196-78455744       Gm10093    35820
## chrX-144402302-144404107       Trpc5       0
## chr8-22565233-22566957       Slc20a2      0
## chr11-96535764-96537610       Skap1       0
## chr3-102365994-102367056       Ngf        102871
## chr17-26511362-26514738       Dusp1      2842
## chr18-70100869-70102635       Rab27b      0
## chr4-97673429-97675079      E130114P18Rik 0
## chr7-125522127-125523994       Il4ra      28287
## chr8-122750781-122753323       Gm20388     0
## chr8-126698464-126700253      Irf2bp2    104477
## chr14-96961817-96963629       Klhl1      442714
## chr5-118649190-118653881       Med13l      0
## chr3-122634204-122636017      Fnbp11    14536
## chr8-12278784-12280440      A230072I06Rik 0
## chr7-140057313-140059317       Msx3       8223
## chr16-31908424-31911509       Mfi2       9403
## chr4-3503229-3506134        Tmem68    42906
## chr2-17363627-17366172       Neb1       0
## chr11-34395466-34396947       Fam196b     0
## chr9-99080682-99082185       Pik3cb      0

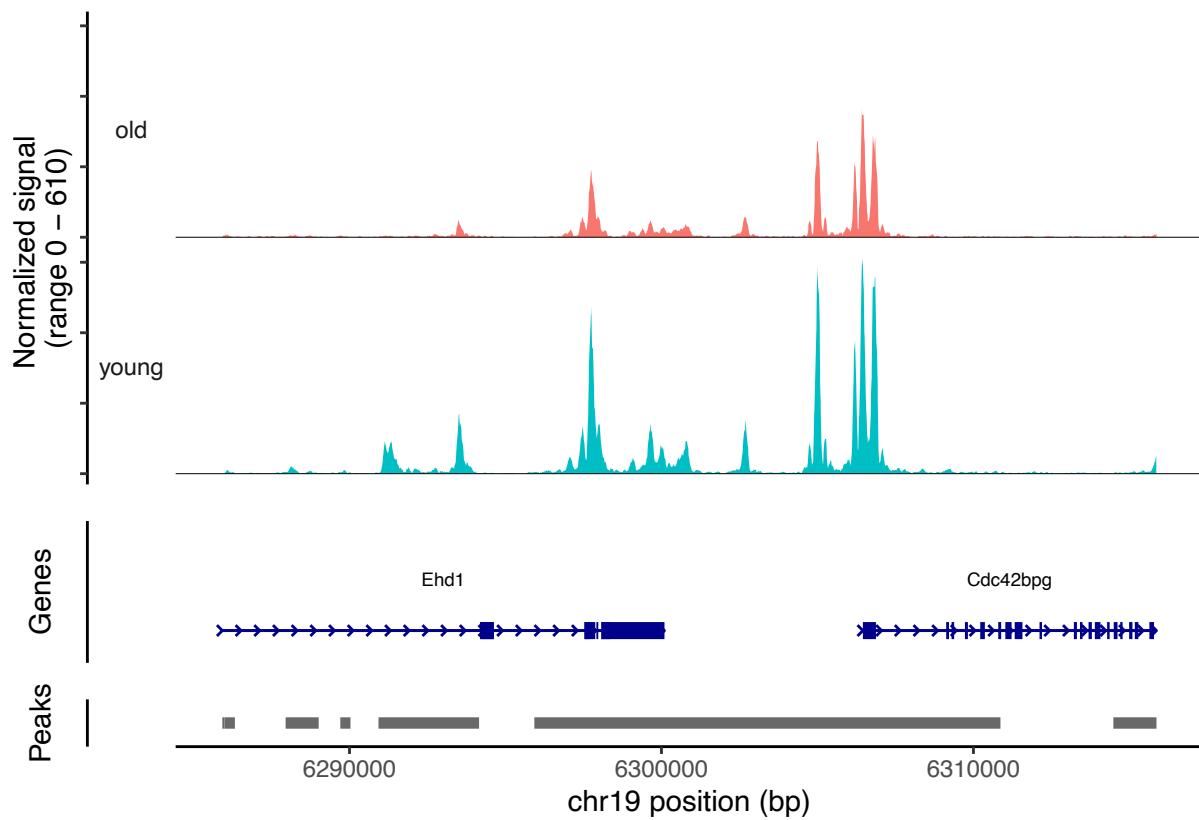
```

```

CoveragePlot(
  object = data,
  region = rownames(da_peaks)[2],
  extend.upstream = 10000,
  extend.downstream = 5000,
  group.by = "dataset"
)

```

```
## Warning: Removed 24 rows containing missing values ('geom_segment()').
```

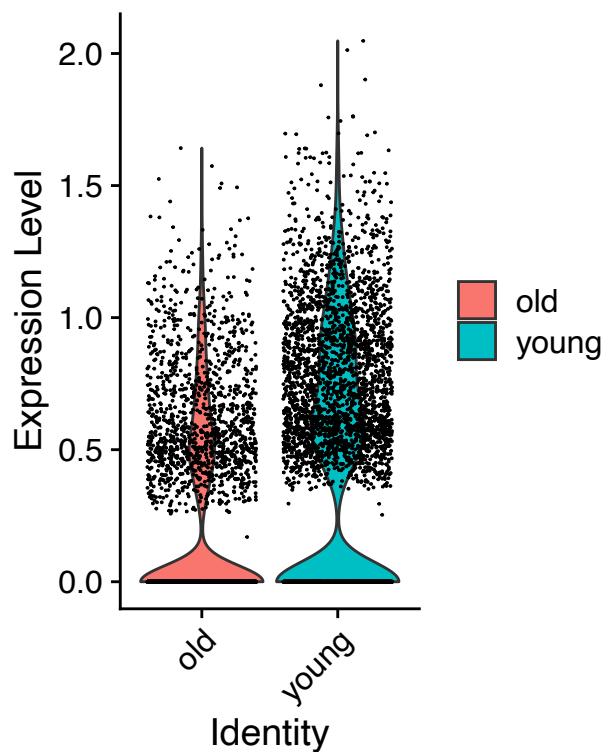


```

plot1 <- VlnPlot(
  object = data,
  features = rownames(da_peaks)[2],
  group.by = "dataset"
)
plot2 <- FeaturePlot(
  object = data,
  features = rownames(da_peaks)[2],
  max.cutoff = 'q95'
)
plot1 | plot2

```

chr19–6295925–6310866



chr19–6295925–6310866

