# Multiome scRNA-seq and scATAC-seq Downstream Analysis with Seurat and Signac

*Clustering Determination*

**Session 2**

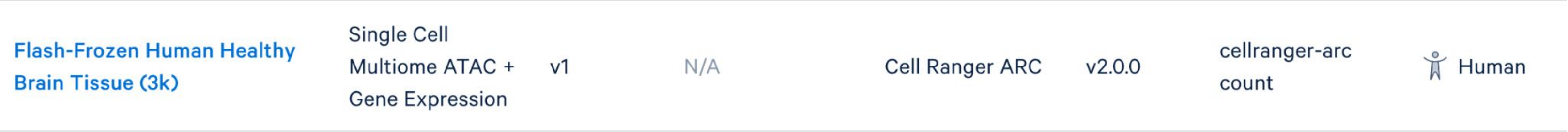
Cynthia Cardinault

March 22, 2024

# Topics

- Clustering with Seurat and Signac
    - Available approaches
    - Steps
    - Clustering for RNA-seq
    - Clustering for ATAC-seq
    - Combination of RNA and ATAC-seq modalities
- Clustering overview in Loupe Browser

# Single Cell Multiome ATAC + Gene Expression Dataset by Cell Ranger ARC 2.0.0



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Flash-Frozen Human Healthy Brain Tissue (3k) | Single Cell Multiome ATAC + Gene Expression | v1 | N/A | Cell Ranger ARC | v2.0.0 | cellranger-arc count | Human |

❑ Link to data:
[Flash-Frozen Human Healthy Brain Tissue (3k)](https://www.10xgenomics.com/datasets/frozen-human-healthy-brain-tissue-3-k-1-standard-2-0-0)
https://www.10xgenomics.com/datasets/frozen-human-healthy-brain-tissue-3-k-1-standard-2-0-0

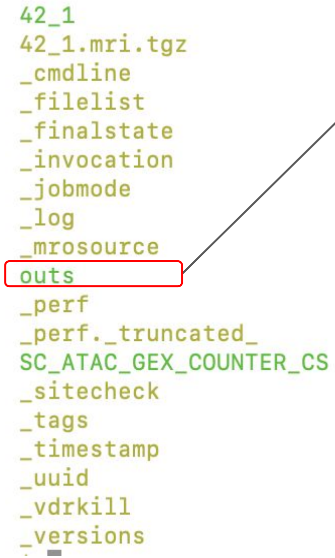
❑ Cell Ranger-ARC (cellranger-atac v2.0.0)

❑ Seurat v5 (2022-11-18)

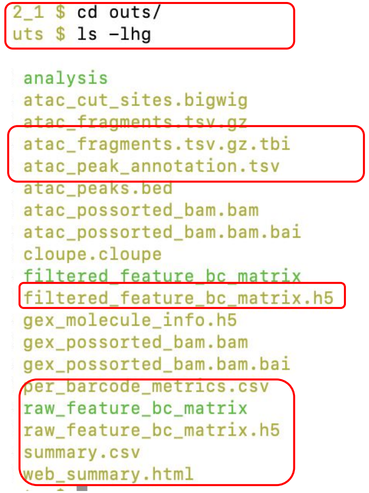
❑ Signac v1.9.0 (2022-12-08)

# Single Cell Multiome ATAC + Gene Expression Dataset

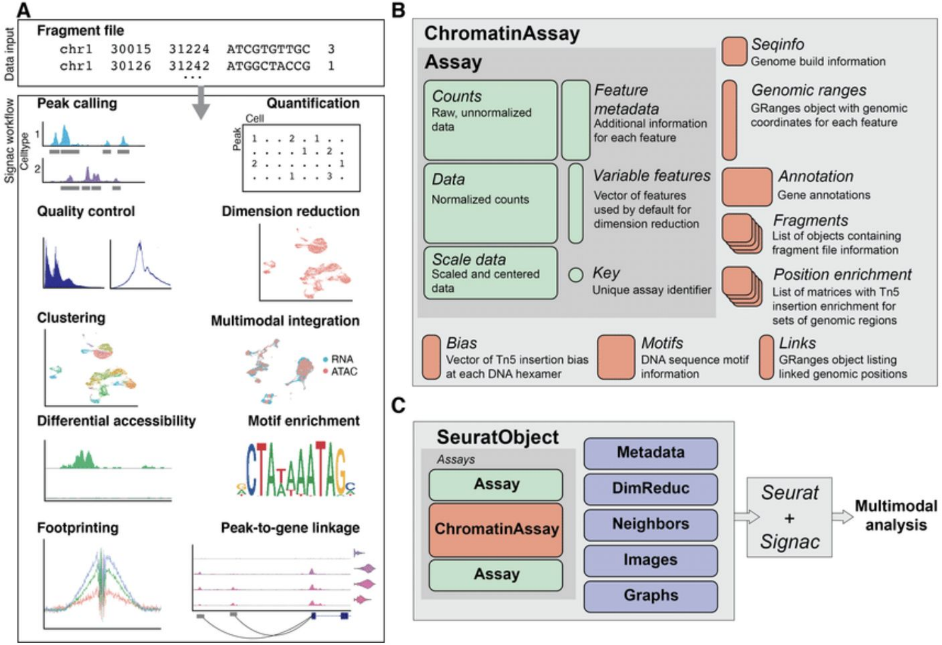
# by Cell Ranger ARC 2.0.0



```
42_1
42_1.mri.tgz
_cmdline
_filelist
_finalstate
_invocation
_jobmode
_log
_mrosource
outs
_perf
_perf._truncated_
SC_ATAC_GEX_COUNTER_CS
_sitecheck
_tags
_timestamp
_uuid
_vdrkill
_versions
```



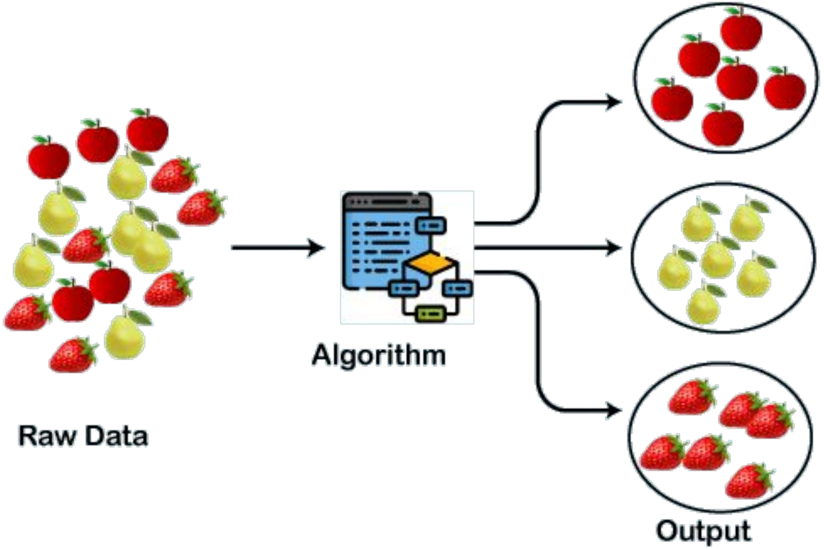
```
2_1 $ cd outs/
uts $ ls -lhg

 analysis
 atac_cut_sites.bigwig
 atac_fragments.tsv.gz
 atac_fragments.tsv.gz.tbi
 atac_peak_annotation.tsv
 atac_peaks.bed
 atac_possorted_bam.bam
 atac_possorted_bam.bam.bai
 cloupe.cloupe
 filtered_feature_bc_matrix
 filtered_feature_bc_matrix.h5
 gex_molecule_info.h5
 gex_possorted_bam.bam
 gex_possorted_bam.bam.bai
 per_barcode_metrics.csv
 raw_feature_bc_matrix
 raw_feature_bc_matrix.h5
 summary.csv
 web_summary.html
```



# What is clustering

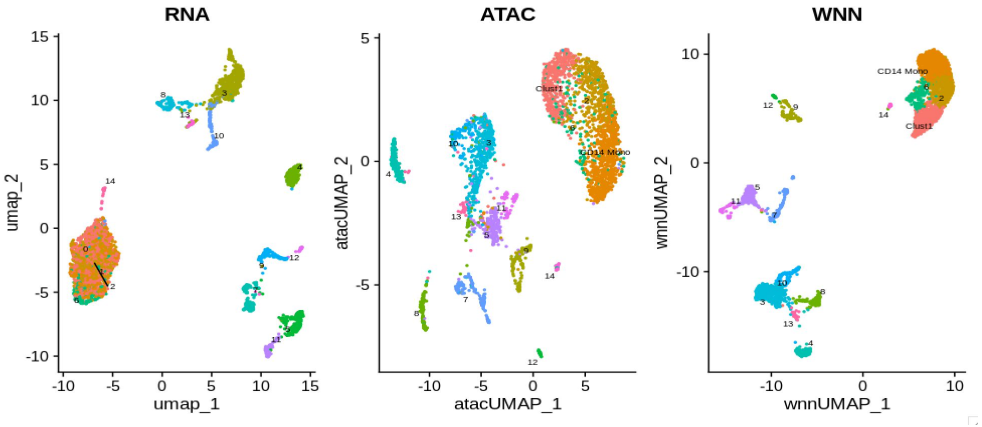
Clustering is a technique in machine learning used to group similar data points together based on certain features or characteristics

1. Partitioning Clustering
2. Density-Based Clustering
3. Distribution Model-Based Clustering
4. Hierarchical Clustering
5. Fuzzy Clustering



# Choice of algorithms

Each algorithm has its very own strengths and weaknesses, and the selection relies upon at the particular problem and the characteristics of the data.



PCA for GEX (k-means)

LSI for ATAC (spherical clustering)

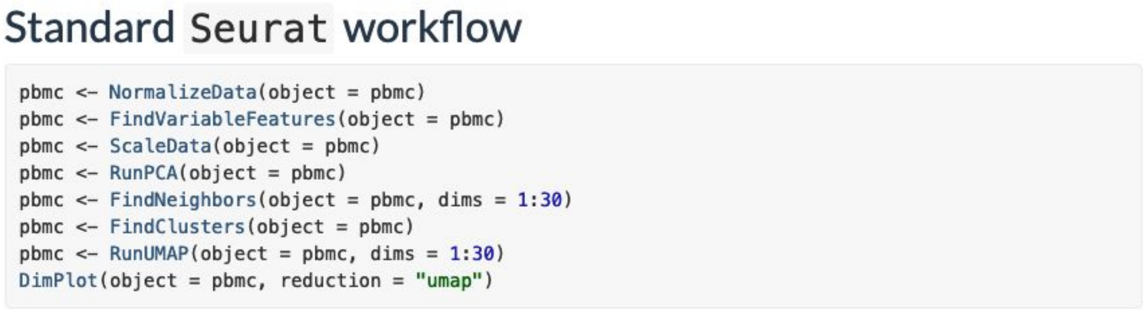
WNN for the integrated

# Clustering determination in multiome datasets

1. Perform pre-processing and dimensional reduction on both assays **independently**
2. For RNA-seq we have 2 approaches for clustering:
   a. Standard Seurat Workflow
   b. SCtransform

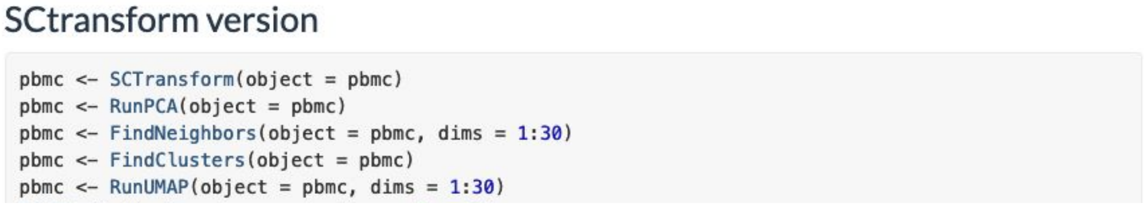
3. For clustering for RNA and ATAC-seq data we applied:
   a. PCA for GEX (k-means)
   b. LSI for ATAC (spherical clustering)
   c. WNN for the integrated

# Approaches for clustering determination



## Standard Seurat workflow

```
pbmc <- NormalizeData(object = pbmc)
pbmc <- FindVariableFeatures(object = pbmc)
pbmc <- ScaleData(object = pbmc)
pbmc <- RunPCA(object = pbmc)
pbmc <- FindNeighbors(object = pbmc, dims = 1:30)
pbmc <- FindClusters(object = pbmc)
pbmc <- RunUMAP(object = pbmc, dims = 1:30)
DimPlot(object = pbmc, reduction = "umap")
```



## SCtransform version

```
pbmc <- SCTransform(object = pbmc)
pbmc <- RunPCA(object = pbmc)
pbmc <- FindNeighbors(object = pbmc, dims = 1:30)
pbmc <- FindClusters(object = pbmc)
pbmc <- RunUMAP(object = pbmc, dims = 1:30)
```

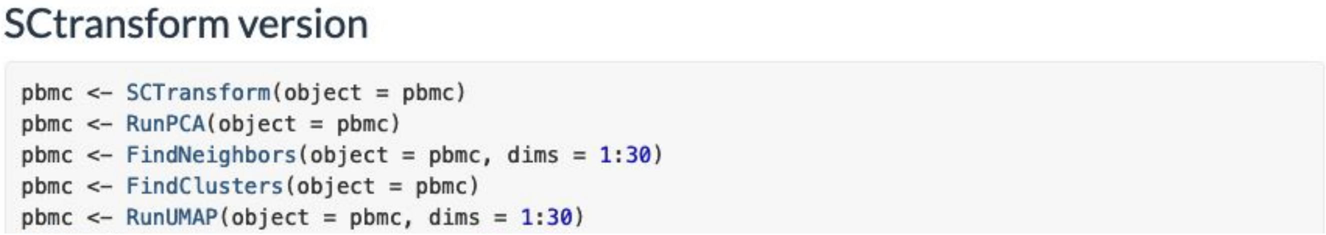
Seurat 5.0.0

**(1) SCTransform()** is defined as a framework for the normalization and variance stabilization of molecular count data.

**(2) SCTransform** omits the need for heuristic steps.

**(3) Global-scaling** relies on an assumption that each cell originally contains the same number of RNA molecules. Seurat developers proposed the **SCTransform** alternative workflow.

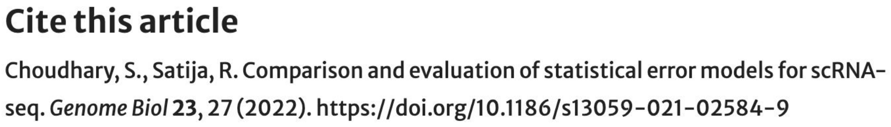
# Standard Seurat workflow: normalization



## SCtransform version

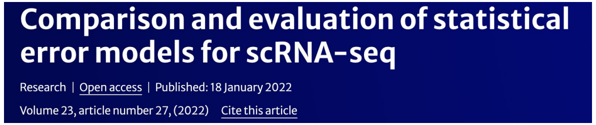
```
pbmc <- SCTransform(object = pbmc)
pbmc <- RunPCA(object = pbmc)
pbmc <- FindNeighbors(object = pbmc, dims = 1:30)
pbmc <- FindClusters(object = pbmc)
pbmc <- RunUMAP(object = pbmc, dims = 1:30)
```

**(4)** The function can also **remove confounding sources of variation**, for example, mitochondrial percentage.
**(5)** In **Seurat v5, SCT v2 is applied by default.** It uses a "Regularized negative binomial regression", which removes unwanted effects from UMIs and return *Pearson* residuals.



### Cite this article

Choudhary, S., Satija, R. Comparison and evaluation of statistical error models for scRNA-seq. *Genome Biol* **23**, 27 (2022). https://doi.org/10.1186/s13059-021-02584-9

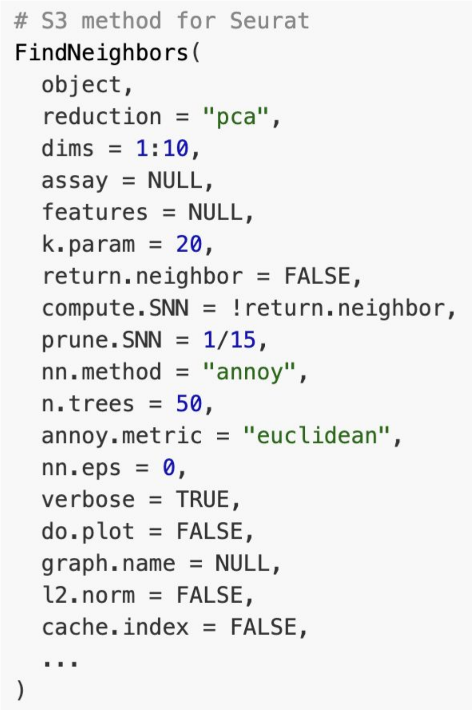


**Comparison and evaluation of statistical error models for scRNA-seq**

Research | Open access | Published: 18 January 2022

Volume 23, article number 27, (2022) Cite this article

# Cluster determination in RNA assay



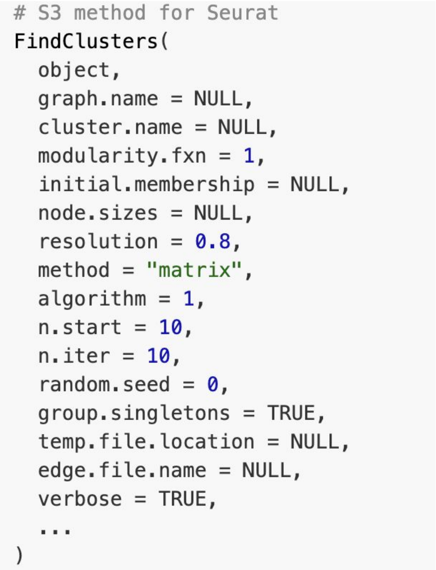
```
# S3 method for Seurat
FindNeighbors(
  object,
  reduction = "pca",
  dims = 1:10,
  assay = NULL,
  features = NULL,
  k.param = 20,
  return.neighbor = FALSE,
  compute.SNN = !return.neighbor,
  prune.SNN = 1/15,
  nn.method = "annoy",
  n.trees = 50,
  annoy.metric = "euclidean",
  nn.eps = 0,
  verbose = TRUE,
  do.plot = FALSE,
  graph.name = NULL,
  l2.norm = FALSE,
  cache.index = FALSE,
  ...
)
```

1. Construct a KNN graph based on the euclidean distance in the PCA space
2. Use that KNN graph to construct the SNN graph by calculating the neighborhood overlap (Jaccard index) between every cell and its k.param nearest neighbors. This step takes as input the first 10 PCs.

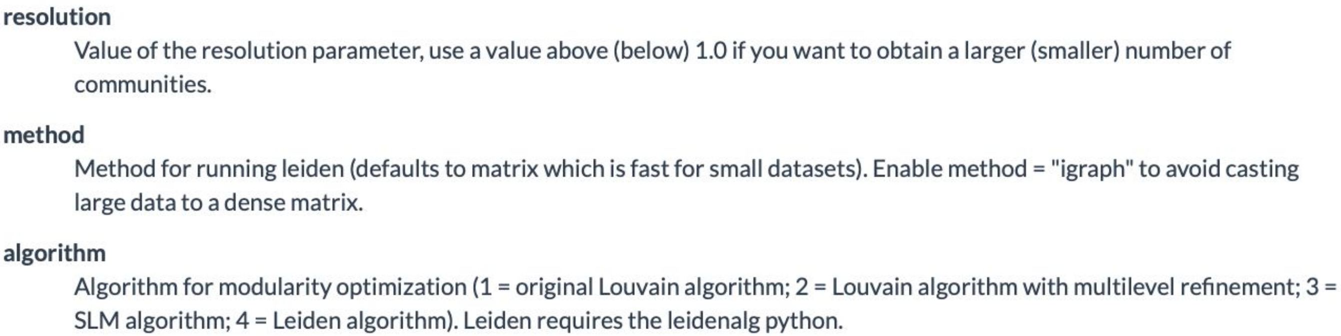
# Cluster determination in RNA assay

Identify clusters of cells by a SNN modularity optimization based clustering algorithm.

1. First calculate k-nearest neighbors and construct the SNN graph.
2. Then optimize the modularity function to determine clusters.



```
# S3 method for Seurat
FindClusters(
  object,
  graph.name = NULL,
  cluster.name = NULL,
  modularity.fxn = 1,
  initial.membership = NULL,
  node.sizes = NULL,
  resolution = 0.8,
  method = "matrix",
  algorithm = 1,
  n.start = 10,
  n.iter = 10,
  random.seed = 0,
  group.singletons = TRUE,
  temp.file.location = NULL,
  edge.file.name = NULL,
  verbose = TRUE,
  ...
)
```



**resolution**

Value of the resolution parameter, use a value above (below) 1.0 if you want to obtain a larger (smaller) number of communities.

**method**

Method for running leiden (defaults to matrix which is fast for small datasets). Enable method = "igraph" to avoid casting large data to a dense matrix.

**algorithm**

Algorithm for modularity optimization (1 = original Louvain algorithm; 2 = Louvain algorithm with multilevel refinement; 3 = SLM algorithm; 4 = Leiden algorithm). Leiden requires the leidenalg python.

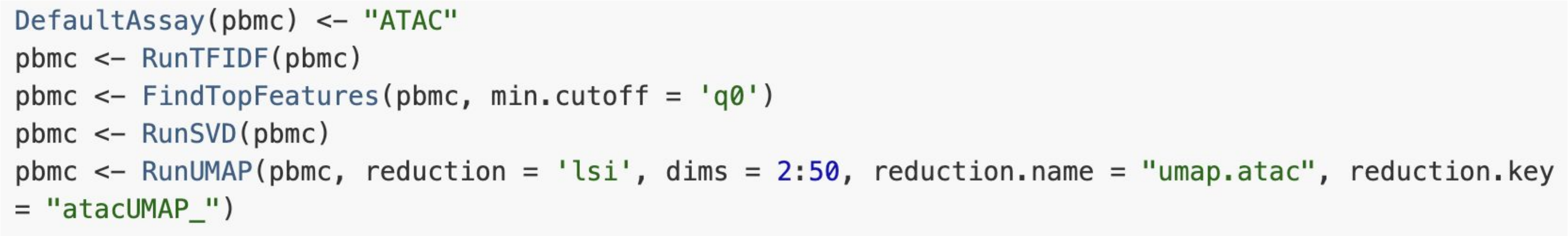
- Louvain performs a KNN graph, with edges drawn between cells with similar gene expression patterns.
- Louvain keeps visiting all nodes in a network until there are no more node movements that increase the quality function, while Leiden uses a fast local move procedure in this phase.

# Cluster determination in ATAC assay

After constructing the initial feature count matrix, several data transformation methods can be applied to compensate for the inherent sparsity before downstream analysis.

- Binarization is one of the most frequently used transformation methods to alleviate potential problems arising from sequencing depth or PCR amplification artifacts.
- There are a growing number of tools adopted **latent semantic indexing (LSI)**. For example: **Signac** and **ArchR**.
- **LSI** is a NLP originally designed to assess document similarity based on word counts.  In the case of **scATAC-seq** data, cells are regarded as documents, whereas peak regions are regarded as words.

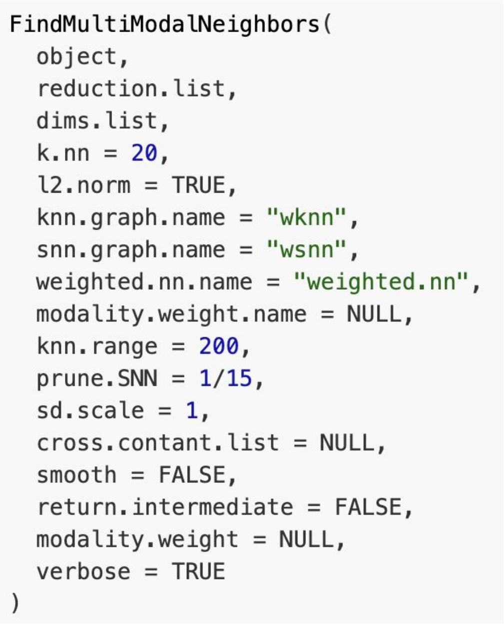
# Cluster determination in ATAC assay



```
DefaultAssay(pbmc) <- "ATAC"
pbmc <- RunTFIDF(pbmc)
pbmc <- FindTopFeatures(pbmc, min.cutoff = 'q0')
pbmc <- RunSVD(pbmc)
pbmc <- RunUMAP(pbmc, reduction = 'lsi', dims = 2:50, reduction.name = "umap.atac", reduction.key
= "atacUMAP_")
```

1. For Signac, the combined steps of TF-IDF followed by SVD are known as latent semantic indexing (LSI).
2. TF-IDF (Term frequency inverse document frequency) normalization.
3. RunSVD ( singular value decomposition) uses irlba (implicitly restarted Lanczos bidiagonalization algorithm), finds a few approximate largest SV and its  corresponding singular vectors of a sparse matrix.

# Combination of RNA-seq and ATAC-seq modalities

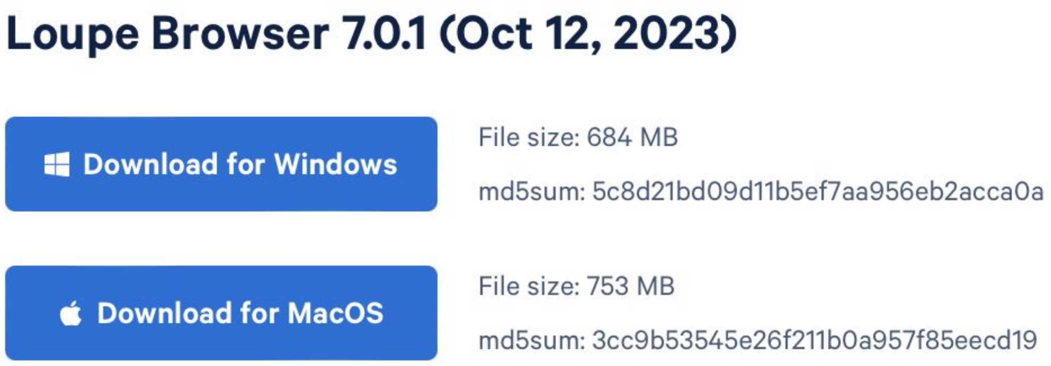


```
FindMultiModalNeighbors(
  object,
  reduction.list,
  dims.list,
  k.nn = 20,
  l2.norm = TRUE,
  knn.graph.name = "wknn",
  snn.graph.name = "wsnn",
  weighted.nn.name = "weighted.nn",
  modality.weight.name = NULL,
  knn.range = 200,
  prune.SNN = 1/15,
  sd.scale = 1,
  cross.contant.list = NULL,
  smooth = FALSE,
  return.intermediate = FALSE,
  modality.weight = NULL,
  verbose = TRUE
)
```

1. Construct a weighted nearest neighbor (WNN) graph.
   a. Identify the NN based on a weighted combination of two modalities.
2. Takes as input two dimensional reductions, one computed for each modality.

# Explore Cell Ranger-ARC clusters with Loupe Browser

Loupe is a visualization R package designed to offer intuitive analysis features for exploring 10x Genomics data. It also supports the analysis of Seurat processed data by converting Seurat objects into Loupe Browser files using the LoupeR package.



## Loupe Browser 7.0.1 (Oct 12, 2023)

File size: 684 MB

md5sum: 5c8d21bd09d11b5ef7aa956eb2acca0a

File size: 753 MB

md5sum: 3cc9b53545e26f211b0a957f85eecd19

https://www.10xgenomics.com/support/software/loupe-browser/downloads

# Thanks