scisorATAC: standard workflow

Wen Hu and Careen Foord

2023-04-30

Background

The *scisorATAC* package allows you to down-sample reads and exons from long-read RNA and cells and peaks from ATAC data. RNA data should be processed through the *scisorseqr* pipeline and its generated *AllInfo.gz* files are used for input. ATAC data should be processed through *CellRanger* and *Signac*. This is a linux-based package which operates in R.

RNA Uses

1.Compares exon inclusion and exclusion between 2 groups

2.Down-sample read counts per exon to ensure equal power given across exons

3.Down-sample number of exons selected to calculate % exons significant

4.Repeat the percent significant calculation a user-given number of times and plot the distribution

ATAC Uses

1.Down-sample number of cells from starting data set and call peaks, and then down-sample peaks.

2.Repeat the percent significant calculation a user-given number of times and plot the distribution

Setup

Software required for RNA functions

- python2.7
- R >= 4.2 with the following installed:
  - rstatix
  - ggplot2
  - magrittr or tidyverse
  - dplyr

Software required for ATAC functions

- R >= 4.2 with the following installed:
  - Signac
  - Seurat
  - GenomeInfoDb
  - harmony
  - dplyr
  - tidyr
  - GenomicRanges

Documentation Cheat Sheet

Each function's proper usage and inputs can be viewed interactively by putting a "?" in front of the function name like so:

```
?casesVcontrols
```

Notes

- As many parts of this package rely on random sampling, make sure to remove any previously set seeds before starting your analysis. If going through the example, set example = TRUE in functions to reproduce the example results

```
rm(.Random.seed, envir=globalenv())
```

- when inputting function variables, place them in the order specified below.

Example Dataset

Example datasets are available to use by running the command below after installation

```
DownloadRefs()
```

RNA Analysis

Step 1: Exon Comparison Analysis

The first step is to run the exon comparison analysis between 2 user given AllInfo files. The output will create an OutputDir with the results of the comparison.

For this the user needs to specify:

- caseList: complete path to the first AllInfo.gz file. ** This input only accepts the complete path **
- controlList: complete path to the second AllInfo.gz file. ** This input only accepts the complete path **
- chrom_file: a user-made file with the desired chromosomes tested. See an example in the "Refs" folder
- CellTypeFile: a user-made file with the list of all celltypes to be considered. See an example in the "Refs" folder
- annotation_path: path to the species specific gencode annotation

other flexible inputs:

- numThreads: number of threads to be used; default = 10
- ci_low min percent spliced inclusion considered; default = 0.05
- ci_upper: max percent spliced inclusion considered; default = 0.95
- min_reads: minimum number of reads for sum of 2 allInfos for a given exon. default = 10
- OL_fraction: the fraction of the reads for a given position must be either inclusion or exclusion; default = 0.8
- zipping_function: command for unzipping files; default Linux as "zcat".
- OutputDir: name of Output Directory. Default = "OutputDir".

```
casesVcontrols(caseList = "complete_path_to_caseList", controlList = "complete_path_to_controlList",
               chrom_file = "path_to_chromFile",numThreads = 10,
               annotation_path = "path_to_anno",
               ci_low = 0.05, ci_upper = 0.95, min_reads = 10,
               zipping_function = "zcat", OL_fraction = 0.8,
               CellTypeFile = "path_to_CelltypeFile", OutputDir="OutputDir")
```

Step 2: Downsampling Reads Per Exon

As some exons have more reads, and thus more power, than others, this function down-samples exons which have a number greater than or equal to the number of reads down-sampled by, and removes those which have less.

This function will down-sample reads from all sub-folders.

Required Input:

- Num_Downsampled_Reads: Number of reads you wish to down-sample to.
- example: to replicate example data. Automatically set value if not specified = FALSE.

To see an example output for Macaque PFC chr22 V. Macaque VIS chr22 run:

```
downsampleReads(Num_Downsampled_Reads = 10, example = TRUE)
```

Output:

- Sampling_DPSI_Table.tab with the following structure:

|Exon_Gene |cases_reads_included |cases_reads_excluded |controls_reads_included |controls_reads_excluded |cases_reads_included_DS |cases_reads_excluded_DS |controls_reads_included_DS |controls_reads_excluded_DS |OG DPSI |DS DPSI|

- CorrelationPlot.pdf: plot of correlation of orginial DPSI and downsampled DPSI

Step 3: Downsampling Exons to calculate % Exons Significant and Iterating

Down-samples number exons and then repeats to generate a distribution of significant events.

This function will down-sample exons from all sub-folders.

Required Input:

- Num_Exons_Selected: Number of exons you wish to down-sample and calculate % exons significant.
- Num_Repeats: Number of iterations. Recommended at least 50 but will vary by data set.

To see an example output for Macaque PFC chr22 V. Macaque VIS chr22 run:

```
downsampleExonsAndIterate(Num_Exons_Selected = 10, Num_Repeats = 100, example = TRUE)
```

Step 4: Plotting Results

Plot the distributions of significant exons from down-sampling with this function.

This function plots all sub-folders distributions and uses wilcox test to calculate differences.

```
ViolinPlot()
```

Output: Downsampling_ViolinPlot.pdf

** Following the example dataset, all % Exons significant are 0.

ATAC Analysis

ATAC functionallity gives you the option to compare downsampled data between 2 cell types, or of the same cell type in different conditions.

Generating an Example ATAC Object

This function generates a small subset of the original dataset. It creates a Seurat object named "combined" with the chromatin assay 'ATAC', can be applied as input object for random subsampling.

Inputs:

- example.data.path: path to example data folder "Refs"
- outDir: the path to a directory will be created for storing the Seurat object generated for subsampling. A Seurat object named as 'combined' includes chromatin assay 'ATAC' will be save as "combined.7K.ATAC.Robj" under this path. The cell type and condition information of each cell can be found as combined$celltype and combined$condition. The example data will cover 7,000 cells and 215683 peaks.
- harmony: Runharmony will be performed if set to be true, default = FALSE

```
Create_Example_ATACobj(example.data.path = "path_to_refs", outDir="OutputDir", harmony = FALSE)
```

Condition Specific ATAC Comparison

Calling differential accessible peaks by comparing two conditions of one specific cell type.

Required input:

- ATACobj: Object has the chromatin assay created with the fragment files of cellranger-arc or cellranger-atac output
- annotatinon.gr: A set of GRanges containing annotations for the genome used, default setting is NULL.
- AssayName: The assay name of the chromatin assay, default setting is "ATAC", which is a required input.
- celltype.query: The query cell type name. The cell type name should have been assigned to cells of the chromatin assay, and the assignment should be listed as column "celltype" in ATACob@meta.data. This is a required input.
- conditionA: Condition A for comparison, which is a required input.
- conditionB: Condition B for comparison, which is a required input.
- cellnum: Number of cells to be randomly subsampled from the whole chromatin assay, the default setting is 500.
- peaknum: Number of peaks to be randomly subsampled from the peaks called from the subsampled cells, the default setting is 5000.
- MinCellRatio:Only test peaks that are detected in a minimum fraction of MinCellRatio cells in either of the two conditions, the default setting is 0.02. To test for differential accessible peaks, the test method is set to be 'LR' and no cutoff for |log2FC|.
- random.repeats: Random subsampling times
- outputDir: The path to the output files.
- savePeakRobj: The peaks called by MACS2 for each subsampling will be stored as assay 'peaks'. It will be saved as Robj for downstream analysis. The default setting is FALSE

```
DAPeaks_ByCondition(ATACobj = combined, annotatinon.gr = NULL,
AssayName = "ATAC", celltype.query = c("ExN_CUX2_RORB"), conditionA = c("VIS"),
conditionB = "PFC", cellnum = 500, peaknum = 5000, MinCellRatio = 0.02,
random.repeats = 10, outputDir = PathToOutputFiles, savePeakRobj = FALSE)
```

Cell Type Specific ATAC Comparison

Calling differential accessible peaks by comparing two conditions of one specific cell type.

Required input:

- ATACobj: Object has the chromatin assay created with the fragment files of cellranger-arc or cellranger-atac output

- annotatinon.gr: A set of GRanges containing annotations for the genome used, default setting is NULL.
- AssayName: The assay name of the chromatin assay, default setting is "ATAC", which is a required input.
- condition.query: The query condition name. The condition name should have been assigned to cells of the chromatin assay, and the assignment should be listed as column "condition" in ATACob@meta.data. This is a required input.
- celltypeA: celltype A for comparison, which is a required input.
- celltypeB: celltype B for comparison, which is a required input.
- cellnum: Number of cells to be randomly subsampled from the whole chromatin assay, the default setting is 500.
- peaknum: Number of peaks to be randomly subsampled from the peaks called from the subsampled cells, the default setting is 5000.
- MinCellRatio:Only test peaks that are detected in a minimum fraction of MinCellRatio cells in either of the two conditions, the default setting is 0.02. To test for differential accessible peaks, the test method is set to be 'LR' and no cutoff for |log2FC|.
- random.repeats: Random subsampling times
- outputDir: The path to the output files.
- savePeakRobj: The peaks called by MACS2 for each subsampling will be stored as assay 'peaks'. It will be saved as Robj for downstream analysis. The default setting is FALSE

```
DAPeaks_ByCelltype(ATACobj = combined, annotatinon.gr = NULL, AssayName = "ATAC",
condition.query = c("VIS"),  celltypeA = c("ExN_CUX2_RORB"),  celltypeB = c("ExN_RORB"),
cellnum = 500, peaknum = 5000, MinCellRatio = 0.02, random.repeats = 10,
outputDir = PathToOutputFiles , savePeakRobj = FALSE)
```

ATAC Outputs

For each subsampling, Output files include:

- Rand.Vx_condition.query_celltypeA.VS. celltypeB_Signac.Robj: Saved Robj with 'peaks' assay for each subsampling (savePeakRobj = TRUE)
- Rand.Vx_condition.query_celltypeA.VS. celltypeB _subsampled.peaks.gr.csv: Subsampled peaks called from subsampled cells
- Rand.Vx_condition.query_celltypeA.VS. celltypeB _all.peaks.granges.Robj: Granges object of all peaks called from subsampled cells
- Rand.Vx_condition.query_celltypeA.VS. celltypeB _DA.peaks.csv: List of all tested peaks
- Stats table of significant peaks of each subsampling: Sig.Peak.Stats_repeats.random.subsampling_condition.query_celltyp