SCEPTRE interoperability with MuData

Gene Katsevich

December 2, 2023

In this document, we present the `sceptreIGVF` package ([GitHub](#)), which provides an interface between the `sceptre` package and `MuData` objects. At present, the `sceptreIGVF` package provides the following two functions:

- `convert_mudata_to_sceptre_object()` converts a `MuData` object to a `sceptre_object`
- `assign_grnas_sceptre()` inputs a `MuData` object, uses `sceptre` to assign gRNAs, and returns a new `MuData` object with another assay containing the gRNA assignments

1 Converting `MuData` to `sceptre_object`

```r
# load libraries
library(sceptreIGVF)
library(MuData)
```

Let us take the Gasperini pilot data as an example:

```r
# read in the MuData object
gasperini_data_dir <- "~/code/research/Pipeline_Gasperini_2019/"
gasperini_data_fp <- paste0(
  gasperini_data_dir,
  "/mudata/Gasperini_2019_sample_pilot.h5mu"
)
gasperini_mudata <- readH5MU(gasperini_data_fp)
gasperini_mudata
```

```
## A MultiAssayExperiment object of 2 listed
##  experiments with user-defined names and respective classes.
##  Containing an ExperimentList class object of length 2:
##  [1] guides: SingleCellExperiment with 98 rows and 7314 columns
##  [2] scRNA: SingleCellExperiment with 2127 rows and 7314 columns
## Functionality:
##  experiments() - obtain the ExperimentList instance
##  colData() - the primary/phenotype DataFrame
##  sampleMap() - the sample coordination DataFrame
##  `$`, `[`, `[[` - extract colData columns, subset, or experiment
##  *Format() - convert into a long or wide DataFrame
##  assays() - convert ExperimentList to a SimpleList of matrices
##  exportClass() - save data to flat files
```

We can convert the `MuData` object to a `sceptre_object` using `convert_mudata_to_sceptre_object()`:

```r
gasperini_sceptre_object <- convert_mudata_to_sceptre_object(gasperini_mudata)
gasperini_sceptre_object
```

```
## An object of class sceptre_object.
##
## Attributes of the data:
##   • 7314 cells
##   • 2127 responses
##   • High multiplicity-of-infection
##   • 98 targeting gRNAs (distributed across 49 targets)
##   • 0 non-targeting gRNAs
##   • 12 covariates (batch_number, doublet_info, doublet_scores, grna_n_nonzero, grna_n_umis, n_counts,
```

We could then apply the `sceptre` pipeline to this `sceptre_object`. However, we might be interested in integrating individual `sceptre` modules into our Nextflow pipeline. For this reason, we would like to have functions wrapping around `sceptre` that operate directly on `MuData` objects.

2   gRNA assignment based on `MuData` objects

Going back to the initial `MuData` object, let us use `assign_grnas_sceptre()` to assign gRNAs based on the thresholding method:

```
gasperini_mudata_updated <- assign_grnas_sceptre(
  mudata = gasperini_mudata,
  method = "thresholding",
  threshold = 5
)
```

The result is another `MuData` object with an additional assay containing the gRNA assignments:

```
gasperini_mudata_updated
```

```
## A MultiAssayExperiment object of 3 listed
##  experiments with user-defined names and respective classes.
##  Containing an ExperimentList class object of length 3:
##  [1] guides: SingleCellExperiment with 98 rows and 7314 columns
##  [2] scRNA: SingleCellExperiment with 2127 rows and 7314 columns
##  [3] grna_assignment: SingleCellExperiment with 98 rows and 7314 columns
## Functionality:
##  experiments() - obtain the ExperimentList instance
##  colData() - the primary/phenotype DataFrame
##  sampleMap() - the sample coordination DataFrame
##  `$`, `[`, `[[` - extract colData columns, subset, or experiment
##  *Format() - convert into a long or wide DataFrame
##  assays() - convert ExperimentList to a SimpleList of matrices
##  exportClass() - save data to flat files
```

In the context of a Nextflow pipeline, the resulting `MuData` object could be written to disk using `MuData::writeH5MU()` and then read in by the next process in the pipeline.

The function `sceptreIGVF::assign_grnas_sceptre()` has the same gRNA assignment options as `sceptre::assign_grnas()`: thresholding, maximum, and mixture. For documentation of these options, see the corresponding portion of the `sceptre` vignette here.