

restfulSE – experiments with HDF5 server content wrapped in SummarizedExperiment

Vincent J. Carey, stvjc at channing.harvard.edu, Shweta Gopaulakrishnan, reshg at channing.harvard.edu, includes contributions from S. Pollack

1 November 2022

Contents

1	restfulSE	2
1.1	HSDS-backed SummarizedExperiment	2
2	Some details	4
2.1	Motivation	4
2.2	Executive summary	4
2.3	10xGenomics examples	5
2.4	Background	7

1 restfulSE

This R package includes proof-of-concept code illustrating several approaches to Summarized-Experiment design with assays stored out-of-memory.

1.1 HSDS-backed SummarizedExperiment

Several data structures are introduced

- to model the HDF Scalable Data Service (HSDS) server data architecture and
- to perform targeted extraction of numerical data from HDF5 arrays stored on the server.

We work with HDF Object store (<https://www.hdfgroup.org/solutions/hdf-cloud/>).

1.1.1 Illustration with 10x genomics 1.3 million neurons

We used Martin Morgan's `TENxGenomics` package to transform the sparse-formatted HDF5 supplied by 10x into a dense HDF5 matrix to support natural slicing. Thanks to native compression in HDF5, the data volume expansion is modest.

A helper function in the `restfulSE` package creates a `RESTfulSummarizedExperiment` instance that points to the full numerical dataset.

```
library(restfulSE)
my10x = sel.3M()
## snapshotDate(): 2022-10-24
## see ?restfulSEData and browseVignettes('restfulSEData') for documentation
## downloading 1 resources
## retrieving 1 resource
## loading from cache
my10x
## class: SummarizedExperiment
## dim: 27998 1306127
## metadata(0):
## assays(1): counts
## rownames(27998): ENSMUSG000000051951 ENSMUSG000000089699 ...
## ENSMUSG000000096730 ENSMUSG000000095742
## rowData names(12): ensid seqnames ... symbol entrezid
## colnames(1306127): AACCTGAGATAGGAG-1 AACCTGAGCGGCTTC-1 ...
## TTTGTCAAGTTAAAGTG-133 TTTGTCATCTGAAAGA-133
## colData names(4): Barcode Sequence Library Mouse
```

As an exercise, we acquire the ENSEMBL identifiers for mouse genes annotated to hippocampus development, which has GO ID [GO:0021766](#), and check counts for 10 genes on 6 samples:

```
library(org.Mm.eg.db)
##
hippdev = select(org.Mm.eg.db,
  keys="GO:0021766", keytype="GO", column="ENSEMBL")$ENSEMBL
## 'select()' returned 1:many mapping between keys and columns
hippdev = intersect(hippdev, rownames(my10x))
#unnname(assay(my10x[ hippdev[1:10], 10001:10006]))
```

restfulSE – experiments with HDF5 server content wrapped in SummarizedExperiment

The result:

```
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    0    0    0    0    0    0
[2,]    0    0    0    0    0    0
[3,]    0    0    0    1    0    0
[4,]    0    1    2    6    5    0
[5,]    0    0    0    0    0    0
[6,]    1    2    4    8    7    3
[7,]    0    0    0    0    0    0
[8,]    0    0    0    0    0    2
[9,]    0    0    0    0    0    0
[10,]   3    0    3    0    1    9
```

1.1.2 Illustration with GTEx tissue expression

We exported the content of the [recount2 GTEx gene-level quantifications](#) to our HDF5 server. A convenience function is available:

```
tiss = gtexTiss()
## snapshotDate(): 2022-10-24
## see ?restfulSEData and browseVignettes('restfulSEData') for documentation
## downloading 1 resources
## retrieving 1 resource
## loading from cache
tiss
## class: RangedSummarizedExperiment
## dim: 58037 9662
## metadata(0):
## assays(1): recount
## rownames(58037): ENSG00000000003.14 ENSG00000000005.5 ...
##   ENSG00000283698.1 ENSG00000283699.1
## rowData names(3): gene_id bp_length symbol
## colnames(9662): SRR660824 SRR2166176 ... SRR612239 SRR615898
## colData names(82): project sample ... title characteristics
```

We'll use this remote data as a tool for investigating transcriptional patterns in brain anatomy. We can identify the samples from brain using the 'smtsd' colData element:

```
binds = grep("Brain", tiss$smtsd)
table(tiss$smtsd[binds][1:100]) # check diversity in 100 samples
##
##           Brain - Amygdala
##                   4
## Brain - Anterior cingulate cortex (BA24)
##                   5
##           Brain - Caudate (basal ganglia)
##                   10
##           Brain - Cerebellar Hemisphere
##                   9
##           Brain - Cerebellum
##                   13
```

```
##           Brain - Cortex
##                               13
##           Brain - Frontal Cortex (BA9)
##                               10
##           Brain - Hippocampus
##                               8
##           Brain - Hypothalamus
##                               5
## Brain - Nucleus accumbens (basal ganglia)
##                               7
##           Brain - Putamen (basal ganglia)
##                               4
##           Brain - Spinal cord (cervical c-1)
##                               3
##           Brain - Substantia nigra
##                               9
```

We'll identify genes annotated to neurotrophic functions using another convenience function in this package:

```
ntgenes = goPatt(termPattern="neurotroph")
## 'select()' returned 1:1 mapping between keys and columns
## 'select()' returned 1:many mapping between keys and columns
head(ntgenes)
##           GO EVIDENCE ONTOLOGY           ENSEMBL SYMBOL
## 1 GO:0004897      IBA           MF ENSG00000122756  CNTFR
## 2 GO:0004897      IDA           MF ENSG00000122756  CNTFR
## 3 GO:0004897      IMP           MF ENSG00000160712  IL6R
## 4 GO:0004897      IDA           MF ENSG00000134352  IL6ST
## 5 GO:0004897      IDA           MF ENSG00000113594  LIFR
## 6 GO:0005030      IEA           MF ENSG00000151892  GFRA1
```

2 Some details

2.1 Motivation

Extensive human and computational effort is expended on downloading and managing large genomic data at site of analysis. Interoperable formats that are accessible via generic operations like those in RESTful APIs may help to improve cost-effectiveness of genome-scale analyses.

In this report we examine the use of HDF5 server as a back end for assay data, mediated through the RangedSummarizedExperiment API for interactive use.

A modest server configured to deliver HDF5 content via a RESTful API has been prepared and is used in this vignette.

2.2 Executive summary

We want to provide rapid access to array-like data. We'll work with the Banovich 450k data as there is a simple check against an in-memory representation.

restfulSE – experiments with HDF5 server content wrapped in SummarizedExperiment

We build a SummarizedExperiment by combining an assay-free RangedSummarizedExperiment with this reference.

```
ehub = ExperimentHub::ExperimentHub()
## snapshotDate(): 2022-10-24
tag = names(AnnotationHub::query(ehub, "banoSEMta"))
banoSE = ehub[[tag[1]]]
## see ?restfulSEData and browseVignettes('restfulSEData') for documentation
## loading from cache
ds = HSDSArray(endpoint=URL_hsd(),svrtype="hsds",
               domain="/shared/bioconductor/bano_meQTLex.h5",dsetname="/assay001")
assays(banoSE, withDimnames=FALSE) = SimpleList(betas=ds)
banoSE
## class: RangedSummarizedExperiment
## dim: 329469 64
## metadata(0):
## assays(1): betas
## rownames(329469): cg00000029 cg00000165 ... ch.9.98989607R ch.9.991104F
## rowData names(10): addressA addressB ... probeEnd probeTarget
## colnames(64): NA18498 NA18499 ... NA18489 NA18909
## colData names(35): title geo_accession ... data_row_count naid
```

We can update the SummarizedExperiment metadata through subsetting operations, and then extract the relevant assay data. The data are retrieved from the remote server with the `assay` method.

```
rbanoSub = banoSE[5:8, c(3:9, 40:50)]
assay(rbanoSub)
## <4 x 18> matrix of class DelayedMatrix and type "double":
##           NA18501      NA18502      NA18516 ...      NA19138      NA19140
## cg00000363  0.325433263  1.377820005  0.596699897 .  0.966695669  1.203765271
## cg00000622  0.003436888 -0.668499289 -1.210634762 .  0.076062477  0.958031578
## cg00000714 -1.184443665 -1.654047967 -0.174729357 .  0.325742947 -0.008202908
## cg00000734  0.153831565 -1.299289359  1.903976827 .  1.185320424  0.319937329
```

2.3 10xGenomics examples

2.3.1 t-SNE for a set of genes annotated to hippocampus

We have used Martin Morgan's TENxGenomics package to create a dense HDF5 representation of the assay data, and placed it in HSDS.

We will subset genes annotated to hippocampus development. Here are some related categories:

```
12092 G0:0021766      hippocampus development
12096 G0:0021770      parahippocampal gyrus development
34609 G0:0097410      hippocampal interneuron differentiation
34631 G0:0097432      hippocampal pyramidal neuron differentiation
34656 G0:0097457      hippocampal mossy fiber
35169 G0:0098686      hippocampal mossy fiber to CA3 synapse
42398 G0:1990026      hippocampal mossy fiber expansion
```

Oct 2022: Following code needs revision.

restfulSE – experiments with HDF5 server content wrapped in SummarizedExperiment

```
library(org.Mm.eg.db)
atab = select(org.Mm.eg.db, keys="G0:0021766", keytype="G0", columns="ENSEMBL")
hg = atab[, "ENSEMBL"]
ehub = ExperimentHub::ExperimentHub()
lkst100k = AnnotationHub::query(ehub, "st100k")
tenx100k = ehub[[names(lkst100k)]]
length(hgok <- intersect(hg, rownames(tenx100k)))
```

This is a very scattered collection of rows in the matrix. We acquire expression measures for genes annotated to hippocampus on 4000 samples. t-SNE is then used to project the log-transformed measures to the plane.

```
hipn = assay(tenx100k[hgok, 1:4000]) # slow
d = dist(t(log(1+hipn)), method="manhattan")
proj = Rtsne(d)
```

```
plot(proj$Y)
```

2.3.2 A set of genes related to the visual cortex

Tasic et al. (Nature neuro 2016, DOI 10.1038/nn.4216) describe single cell analysis of the adult murine brain, identify clusters of cells with distinct transcriptional profiles and anatomic location, and enumerate lists of genes that discriminate these clusters. The tasicST6 DataFrame provides details.

```
#data("tasicST6", package = "restfulSEData")
ehub = ExperimentHub::ExperimentHub()
## snapshotDate(): 2022-10-24
tag = names(AnnotationHub::query(ehub, "tasicST6"))
tasicST6 = ehub[[tag[1]]]
## see ?restfulSEData and browseVignettes('restfulSEData') for documentation
## downloading 1 resources
## retrieving 1 resource
## loading from cache
tasicST6
## DataFrame with 49 rows and 4 columns
##           clid      txtype1      txtype2      genes
##   <character> <character> <character> <List>
## 1           f01      Vip          Mybpc1  Crispld2, Cxcl14, Tpm2, ...
## 2           f02      Vip          Parm1   Cxcl14, Car4, Tac2
## 3           f03      Vip          Sncg    Reln, Npy2r, Tnfaip8l3, ...
## 4           f04      Vip          Chat    Aebp1, Slc18a3, Pvrl4, ...
## 5           f05      Vip          Gpc3    Bcar3, Mab21l1, Pbx3, ...
## ...
## 45          f45      Oligo 9630013A20Rik  Brca1, Rnf122, Mbp, ...
## 46          f46      Oligo          Opalin  Mbp, Mog, Aspa, ...
## 47          f47      Micro          Ctss    Cx3cr1, Clqb, Cd53, ...
## 48          f48      Endo          Xdh     Tbc1d4, AI467606, Exosc7, ...
## 49          f49      SMC          Myl9    Bgn, Nupr1, Casq2, ...
```

restfulSE – experiments with HDF5 server content wrapped in SummarizedExperiment

Key high-level discrimination concerns cells regarded as GABAergic vs. glutamatergic (inhibitory vs excitatory neurotransmission).

2.4 Background

Banovich et al. published a subset of DNA methylation measures assembled on 64 samples of immortalized B-cells from the YRI HapMap cohort.

```
library(restfulSE)
ehub = ExperimentHub::ExperimentHub()
## snapshotDate(): 2022-10-24
tag = names(AnnotationHub::query(ehub, "banoSEMeta"))
banoSEMeta = ehub[[tag[1]]]
## see ?restfulSEData and browseVignettes('restfulSEData') for documentation
## loading from cache
banoSEMeta
## class: RangedSummarizedExperiment
## dim: 329469 64
## metadata(0):
## assays(1): betas
## rownames(329469): cg00000029 cg00000165 ... ch.9.98989607R ch.9.991104F
## rowData names(10): addressA addressB ... probeEnd probeTarget
## colnames(64): NA18498 NA18499 ... NA18489 NA18909
## colData names(35): title geo_accession ... data_row_count naid
```

The numerical data have been exported using H. Pages' `saveHDF5SummarizedExperiment` applied to the `banovichSE` `SummarizedExperiment` in the `yriMulti` package. The HDF5 component is imported to HSDS with the `hload` utility distributed with the python package `h5pyd`.