Cancer Genomics Cloud & R: Find, Access, and Analyze Petabyte-Scale Cancer Genomic Data on the Cloud

Nan Xiao
Seven Bridges
TCGA is a tremendous gift to the cancer research community …

More than 11,000 cases representing 33 cancer types
UNDERSTANDING TCGA DATASET

- Multiple Samples per Case
- Multiple Analyses per Sample
- Rich metadata: barcodes, UUIDs, XMLs...

---

Nature Genetics 45, 1113-1120 (2013)
As the amount and diversity of data increases, it becomes more difficult to learn from them.
The CGC aims to provide a collaborative environment where researchers can take advantage of co-localized public data (like TCGA) and public tools; but also recombine these with their private data and tools.
sevenbridges is an R/Bioconductor package offering easier programmatic access to CGC.
The **sevenbridges** package offers:

- Complete R API client for CGC/Seven Bridges API
- Common Workflow Language Tool Interface
- Task monitoring / Batch tasks support
- and many helper functions/utilities in R.
Welcome to your new project!

Projects are the core building blocks of the CGC Platform. Each project corresponds to a distinct scientific investigation, serving as a container for its data, analysis pipelines, and results. Projects are shared only by designated project members.

Within your project, you can:
- Start exploring TCGA dataset straight away
- Install your tools on the CGC and create workflows
- Upload your own private data and analyze it along with TCGA data
- Collaborate securely with other researchers

Please record the details of your project here, such as its aims, experimental context, and any other ideas that you’d like to share with your project members. Remember that details of each pipeline execution you run on the CGC are logged on the task page. This notepad is just for your own notes.

You can also use markdown here to add formatting to your notes.

Good luck with your research! If you get stuck, take a look at the Knowledge Center. The Seven Bridges CGC Team

Don't work alone. The best research happens in teams.

_invite members

Share your tools, data, and ideas with collaborators

Tasks

0/1

DESeq2 run - 01-13-17 15:17:35
Submitted by nanx · Jan. 13, 2017 10:18
PROJECT MANAGEMENT VIA API

```r
library("sevenbridges")

# Create Auth object
a = Auth(platform = "cgc", token = "your_token")

# List project and details
a$project(owner = "user")
a$project(detail = TRUE)

# Get billing group ID
bid = a$billing()$id

# Create new project
a$project_new("brca_test", bid,
  description = "BRCA Test")

# Add member to project
m = a$project(id = "demo/brca_test")$
  member_add(username = "new_user")
```
FIND DATA ON CGC
MORE THAN ONE PETABYTE OF TCGA DATA AT YOUR FINGERTIPS

Open Data
Information NOT unique to an individual.
• de-identified clinical data
• gene expression data
• copy number alterations
• epigenetic data

Controlled Data
Information that IS unique to an individual.
• primary sequencing data
• raw & processed SNP6 array data
• raw exon array data
• mutation calls for an individual
CGC DATA BROWSER

- Query metadata
- Add files to project
library("SPARQL")
endpoint = "https://opensparql.sbgenomics.com/blazegraph/namespace/tcga_metadata_kb/sparql"
query = "
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix tcga: <https://www.sbgenomics.com/ontologies/2014/11/tcga#>

where
{
?case a tcga:Case .
?case tcga:hasDiseaseType ?disease_type .
?disease_type rdfs:label 'Lung Adenocarcinoma' .
...
}
"'}
qd = SPARQL(endpoint, query)
df = qd$results
head(df)
DATASETS API

term = list(
    "entity" = "cases",
    "hasSample" = list(
        "hasSampleType" = "Primary Tumor",
        "hasPortion" = list(
            "hasPortionNumber" = 11
        )
    ),
    "hasNewTumorEvent" = list(
        "hasNewTumorAnatomicSite" = c("Liver", "Pancreas"),
        "hasNewTumorEventType" = list(
            "filter" = list(
                "contains" = "Recurrence"
            )
        )
    )
)

a$api(path = "query", body = term, method = "POST")
ADD & ANNOTATE DATA
UPLOAD YOUR DATA

- CGC Uploader (GUI)
- Command Line Uploader
- FTP / HTTP / S3 Volume
- API
UPLOAD A FILE VIA API

myfile = "file_path.fastq"
p = a$project(id = "demo/tcga-demo")

# Load `.meta` for the file by default
p$upload(myfile, overwrite = TRUE)

# Pass metadata manually
p$upload(myfile, overwrite = TRUE,
    metadata = list(library_id = "test_id",
                    platform = "Illumina x11"))

# Rename file
p$upload(myfile, overwrite = TRUE,
    metadata = list(library_id = "new_id")
    name = "sample_new_name.fastq")
# Upload a folder

dirpath = "path_to_dir"
list.files(dirpath)
p$upload(dirpath, overwrite = TRUE)

# Upload a list of files

dirpath = "path_to_dir"
myfiles = list.files(dirpath, recursive = TRUE, full.names = TRUE)
p$upload(myfiles, overwrite = TRUE)
# Locate a sample BAM file in project
p = a$project(id = "demo/tcga-demo")
fl = p$file("sample.bam", exact = TRUE)

# Show tags for single file
fl$tag()

# Add new tags
fl$add_tag("new year new tag")

# Set tags to overwrite existing
x = list("this", "is", 2017)
fl$set_tag(x)

# Set metadata
fl$meta()
fl$set_meta()}
# attach all metadata except "bad_field" and "sample_id"

```r
p$upload(manifest_file = "~/manifest.csv",
          overwrite = TRUE,
          subset = score < 0.5,
          select = -c(bad_field, sample_id))
```
ANALYZE THE DATA
Run an analysis immediately, with ~230 tools and workflows on the CGC today.
RUNNING ANALYSIS TASKS

DESeq2 run - 01-13-17 15:17:35

Inputs
- Raw count data
  - TCGA: US82800149_251976011805_S01_GE2_105...
  - TCGA: US82800149_251976011806_S01_GE2_105...
  - TCGA: US82800149_251976011807_S01_GE2_105...
  - TCGA: US82800149_251976011808_S01_GE2_105...
  - TCGA: US82800149_251976011809_S01_GE2_105...

- Additional metadata
  - No files selected

App Settings
- Fit type: parametric
- FDR cutoff: 0.1
- Reference (control) level: No value

Outputs
- DESeq2 analysis summary: No value
- MA plots: No value
- Collected outputs: No value
- DESeq2 analysis results: No value
- A plot of dispersion estimates: No value
# List public apps
a$app(visibility = "public")

# Copy an app to project
aid = a$public_app()[[1]]$id
a$copy_app(aid, project = pid, name = "new_app_name")

# Add & run new tasks
tsk = p$task_add(name = "new_task",
                description = "new task",
                app = "demo/tcga-demo/rna-seq-alignment-star/0",
                inputs = list(...))

 tsk$run()
 tsk$abort()
RUN TASKS IN BATCH MODE

# Batch by items
```r
tsk = p$task_add(name = "RNA DE Report Batch 1",
                 description = "RNA DE Analysis Report",
                 app = rna_app$id,
                 batch = batch(input = "bamfiles"),
                 inputs = list(...))
```

# Batch by metadata
```r
tsk = p$task_add(name = "RNA DE Report Batch 2",
                 description = "RNA DE Analysis Report",
                 app = rna_app$id,
                 batch = batch(input = "fastq",
                                c("metadata.sample_id",
                                   "metadata.library_id")),
                 inputs = list(...))
```
CONTAINERS + CWL MAKES IT EASY TO PUT NEW TOOLS ON THE CGC… AND OTHER PLACES.
Specification for describing scalable, portable and reproducible computational workflows.
FOR DEVELOPERS, THIS MEANS...

• Easier and faster to deploy your tools.

• Write once, runs everywhere, regardless of operating system/infrastructure.
CWL TOOL: INPUTS, OUTPUTS, AND PARAMETERS
WRITE CWL WITH WEB IDE

cowsay say_what > output.txt
# Create CWL Tool
runif = Tool(id = "runif",
             label = "runif",
             hints = requirements(docker(pull = "rocker/r-base")),
             baseCommand = "Rscript runif.R",
             stdout = "output.txt",
             outputs = output(id = "random", glob = "*.txt"))

# Convert to JSON or YAML
runif$toJSON()
runif$toYAML()
GETTING STARTED

• Try sevenbridges package from Bioconductor
• docker pull sevenbridges/sevenbridges-r
• Pull requests: github.com/sbg/sevenbridges-r
Vignette reference version 1.5.5

All vignettes

- Complete Guide for Seven Bridges API R Client
- Describe and Execute CWL Tools/Workflows in R
- Master Tutorial: Use R for Cancer Genomics Cloud
- Find Data on CGC via Data Explorer, SPARQL, and Datasets API
- Creating Your Docker Container and Command Line Interface (with doopt)
- IDE Container: RStudio Server, Shiny Server, and More

https://sbg.github.io/sevenbridges-r/
NEED MORE RESOURCES?

• Apply to NIHCommons Credits Pilot for new opportunities in cloud computing & storage before Jan 16:
  
  • https://datascience.nih.gov/commons

  • https://www.commons-credit-portal.org/
MORE ON CANCER GENOMICS CLOUD AND SEVEN BRIDGES PLATFORMS

• www.cancergenomicscloud.org

• www.sevenbridges.com/tcga

• www.sevenbridges.com
Thank you!

This project has been funded in whole or in part with Federal funds from the National Cancer Institute, National Institutes of Health, Department of Health and Human Services, under Contract No. HHSN261201400008C.