HIHISIV - The database of the HIV and SIV Host Immune Response

Contents

Data organization

All material was organized as described in this structure below:

```
/hihisiv_gitlab/
|--- a_code/
|    |--- microarray_analysis/
|    |--- rna-seq_analysis/
|         |--- get_sra/
|    |--- tables_organizing/
|--- b_database/
|    |--- tables_hihisiv/ # files '.csv'
          |--- gene_gene/
          |--- gene_species/
          |--- platform_transcript_id/
          |--- saved_google_sheet/
          |--- traits/
|    |--- Dockerfile
|    |--- init.sql
|--- c_web_app/
```

```
|    |--- hihisiv_webapp/
|--- README.md
|--- docker-compose.yml
|--- metadata.html
```

A code

Microarray_analysis

- parameters.R (information about the experiment)
- module_processing.R
- dependencies.R
- raw_activity.R (CEL files, normalized matrix - affy, impute)
- e-set_activity.R (normalized matrix, eSet object)
- limma_activity.R (eSet object and phenodata matrix, differentially expressed genes matrix, limma, ggplot2, RColorBrewer)

Rna-seq_analysis

- get_sra.sh (archive with ids, SRA data, aria2c from aria2)
- sra_to_fastq.sbatch (SRA data, fastq files, fastq-dump from sratoolkit (v. 2.11.3))
- quality_control.sh (fastq, .html reports, fastqc multiqc)
- alignment_rsem.sh (fastq paired, bowtie2; perl; rsem)
- limma_voom.R
- rsem_matrix.sbatch (*genes.results, geneMat.txt, bowtie2; rsem)

Tables to database

- organize_tables.R

B database

Tables built from sheet directly. Manually curated and mounted.

- experiments.csv
- design.csv
- tissue.csv
- platform.csv
- experiment_platform.csv
- project.csv
- project_publication.csv
- publication.csv
- experiment_virus.csv
- virus.csv
- experiment_host.csv
- species.csv
- trait.csv

Tables about platforms:

- extracted from GPL file from GEO;
- removed viral probes in rhesus microarray platform.

Database docker

- a) Build database docker container (remove old containers and images if needed):

```
cd B_Database/ docker build -t hihisiv-postgres .
```

- b) Create empty directory on host to store database files, for example:

```
mkdir /home/user/hihisiv_db
```

C web_app

- Build web application docker container (remove old containers and images if needed):

```
cd c_web_app/hihisiv_webapp
docker build -t hihisiv-webapp .
```

All-in-one - docker-compose

Requirements: docker and docker-composed

Installation steps:

- Edit web application container environment variables to enter database initialization file directory (`b_database/tables_hihisiv`) and database files directory:

```
vi .env
```

- Set `VIRTUAL_HOST` on `docker-compose.yml` to host name of the host that will run the web application.
- Start containers with:

```
docker-compose up -d
```

- Open on the browser:

```
localhost
```

- Stop docker:

```
docker-compose down
```