CRISPRroots version 1.0: User Manual

Giulia I. Corsi, Veerendra P. Gadekar, Jan Gorodkin and Stefan E. Seemann

Center for non-coding RNA in Technology and Health, Department of Veterinary and Animal Sciences,
University of Copenhagen, Thorvaldsensvej 57, 1871 Frederiksberg, Denmark

Contents

1 Config file

The configuration file is textual (*.yaml*) file divided in 3 sections:

1. Project parameters
2. Species files
3. Execution parameters

An example of configuration file is given in the `CRISPRroots` folder and in the test folder.

1.1 Project parameters

- `results_folder`: Path to the folder in which intermediate results are placed. If not existing, the folder will be created.
- `report_folder`: Path to the folder in which reports are placed (assessment of on-target, possible off-targets, genome integrity, tables with summary statistics on pre-processing and processing, differential expression results). If not existing, the folder will be created.
- `samples_folder`: Path to the folder containing all samples in *fastq* format, zipped with *gzip*.
- `biosample`: Name given to the biosample.
- `samples_table`: Path to a tab-separated table specifying the IDs of each sample and their "Condition", either "Original" (the non-edited wild-type) or "Edited". If necessary, additional informative columns to be included in the `DESeq2` formula can be specified (*e.g.* "Time"). An example of this table is shown in **Table 1**. Note that the file suffix/format (*e.g. fastq.gz*) is not present. Paired-end sequenced samples are not duplicated in the table; R1 and R2 are identified by their suffixes (see option `sample_suffix_R1` and `sample_suffix_R2`).
- `sample_suffix, sample_suffix_R1, sample_suffix_R2`: Suffix of *fastq* reads files. For single end reads only one sample suffix is required. Given the example in **Table 1**, a suffix could be `sample_suffix: ".fastq.gz"` for the file Astrocytes_wt_rep1.fq.gz in the samples folder. For paired end reads, suffixes for both sets of reads need to be specified as `sample_suffix_R1: "_R1.fastq.gz"` and `sample_suffix_R2: "_R2.fastq.gz"` for files Astrocytes_wt_rep1_R1.fq.gz and Astrocytes_wt_rep1_R2.fq.gz
- `sequencing`: Type of sequencing, either `"paired"` or `"single"`.
- `single_chr`: List of chromosomes without a homolog (*e.g.* `["chrX",  "chrY"]` in male human)

- `variated_genome`: Set to `"yes"` if the analyses should be performed on a variant-aware version of the genome, in which short variants discovered from the RNA-seq data are introduced in the reference sequence, `"no"` otherwise. If set to `"no"`, a CRISPRoff output file needs to be specified in `crisproff_output`.

Sample_ID	Condition	Time
Astrocytes_wt_rep1	Original	Week5
Astrocytes_wt_rep2	Original	Week5
Astrocytes_wt_rep3	Original	Week5
Astrocytes_wt_rep4	Original	Week10
Astrocytes_wt_rep5	Original	Week10
Astrocytes_wt_rep6	Original	Week10
Astrocytes_APOE_PM_rep1	Edited	Week5
Astrocytes_APOE_PM_rep2	Edited	Week5
Astrocytes_APOE_PM_rep3	Edited	Week5
Astrocytes_APOE_PM_rep4	Edited	Week10
Astrocytes_APOE_PM_rep5	Edited	Week10
Astrocytes_APOE_PM_rep6	Edited	Week10

Tab. 1: Example of samples table for a 6-replicates experiment in which an *APOE* heterozygous mutation was introduced in wild type astrocyte cell lines, harvested after 5 or 10 weeks.

1.2 Species files

- `species`: Scientific name of the subject, lower case, comprising genus and specie separated by an underscore. *e.g.* `"homo_sapiens"`. This information is used by the program eSNPKaryotyping for the analysis of chromosomal aberrations, in which currently only `"homo_sapiens"` and `"mus_musculus"` are supported.

- `picard_reference`: Path to a the *fasta* reference genome. A Picard index generated with Picard's *CreateSequenceDictionary* should be present in the same folder.

- `repeatmasked_regions`: Path to a RepeatMasker *.bed* annotation file.

- `STAR_indexed_transcriptome`: Path to the genome indexed with STAR-2.6.1d.

- `common_variants_vcf`: Path to *.vcf* file of known variants.

- `annotations_gtf`: Path to gene features annotated in *gtf* format.

- `ssu_rrna_silva, lsu_rrna_silva`: Path to file in *fasta* format zipped with *gzip* containing sequences of ribosomal RNA belonging, respectively, to the small and the large ribosomal subunits.

- `dbSNP142`: Path to folder containing common variants in the reference genome, split by chromosome, as required by eSNPKaryotyping.

- `RSeQC_gene_model`: Annotations in *bed* format, required by RSeQC for library type estimation.

1.3 Execution parameters

For each tool, a brief description of the parameters used is given. Please refer to the manuals of these tools for additional information.

- **Cutadapt**: `adapter, adapter_R1, adapter_R2`: Path to a fasta file containing adapter sequences. Two files, `adapter_R1` and `adapter_R2` should be given for paired end reads.
- **Cutadapt**: `pair_filter`: Only for paired end reads, can be set to `"any"` (discard both reads in a pair if any of them satisfies one of the filtering criteria), `"both"` (discard a read pair only if both reads satisfy one of the filtering criteria) or `"first"` (ignore the second read during filtering).
- **Cutadapt**: `phread_score`: Threshold used for quality trimming. Default: `"30"`.
- **Cutadapt**: `min_length`: Reads shorter than this threshold are discarded. We suggest to set it to approx. `"90%"` of the original read length.
- **Cutadapt**: `other`: String with additional Cutadapt parameters. Default: `"--trim-n"`.
- **BBDuck**: `mcf`:Fraction of the read bases to be covered by reference kmers to be considered as match. Default = `"0.5"`.
- **BBDuck**: `K`: Size of the Kmers. Default = `"31"`
- **BBDuck**: `MAX_MEM`: Max amount of memory in GB to be used. Default = `"-Xmx8g"`.
- **STAR**: `threads`: Number of threads to be used. Default = `"12"`.
- **Featurecounts**: `libtype`: Integer indicating the strandness of the reads: `0`=unstranded, `1`=stranded, `2`=reversely stranded; refers to the first reads in paired-end sequencing. If unknown can be inferred with RSeQC (see examples below).
- **DESeq2**: `formula`: Design formula. Note that only the comparison of Edited (nominator) vs Original (denominator) samples will be performed during differential expression. Detault: `~ Condition`
- **BCFtools**: `heterozygous_keep`: Select `"A"` to keep the alternative allele (variant to the reference) or `"R"` for the reference one in the presence of heterozygous mutations.
- **Mutect2**: `base_quality_score_threshold`: Base qualities below this threshold will be changed to a minimum of 6 in `Mutect2`. Default = `30`.

- **Mutect2**: `callable_depth`: Minimum depth for an event to be considered. Default = `10`
- **Mutect2**: `min_base_quality_score`: Minimum base quality to consider a base for calling. Default = `10`.
- **GNU Parallel**: `num_threads`: Number of threads used to launch `Mutect2` in with GNU parallel.
- **R**: `R_install_pkgs`: number of threads to be used when installing R packages. Default = `"10"`.
- **Liftover**: `min_match`: minimum ratio of bases that must remap. Default = `0.95`.
- **Endonuclease**: `cut_position`: Distance from the protospacer adjacent motif 5' end at which the endonuclease cleaves the DNA. Default (SpCas9): `-3`.
- **Endonuclease** `gRNA_sequence`: Sequence of the gRNA, without PAM. *e.g.* `"TGTATTTATACAGAACCACC"`.
- **Endonuclease**: `gRNA_with_PAM_fasta`: path to a *fasta* file containing the gRNA + on-target PAM sequence.
- **Endonuclease**: `binding_site_seq`: List of possible binding sites for the endonuclease. Default (SpCas9): `["GG", "GA", "AG"]`. Note that the ambiguous "N" nucleotide usually reported in the PAM sequence is absent.
- **Endonuclease**: `binding_sites_ratios`: List of weights associated to each of the PAMs defined in `binding_site_seq`. Default (SpCas9): `[1.0, 0.8, 0.9]`.
- **Endonuclease**: `binding_site_distance`: Distance in nucleotides between the binding site (*i.e.* GG in the canonical SpCas9) and the 3' end of the gRNA-DNA duplex on the PAM's strand. In the case of SpCas9 this distance corresponds to the "N" in the "NGG" canonical PAM, thus is equal to 1. Default (SpCas9): `1`.
- **Endonuclease**: `extend_binding`: Extend the size of the region in which the optimal gRNA binding site is searched. Default (SpCas9): `2`.
- **Endonuclease**: `eng_threshold`: Threshold of maximum binding energy in *kcal/mol*. Default : `0.0`.
- **Endonuclease**: `seed_region`: Size of the seed region. Default (SpCas9): `10`.
- **Endonuclease**: `max_mm_seed`: Maximum number of mismatches or bulges tolerated in the seed. Default: `1`.
- **Edits**: `type`: `"KI"` (knockin) or `"KO"` (knockout).

- **Edits**: position: List of genomic coordinates (1-based) at which single base on-target edits are expected, *e.g.* ["chr14:73173676", "chr14:73173674"]. In knockout experiments use the cleavage position.

- **Edits**: mutant: List of mutated nucleotides (*e.g.* ["T", "C"]). The list must be in the same order and of the same length as option position. In knockout experiments, use "N".

- **Edits**: splice_donor, splice_acceptor: Lists of binary values specifying, for each position, if it corresponds to a splice donor/acceptor (1) or not (0). The lists must be in the same order and of the same length as option position. Example: [0, 0].

- **Edits**: intron: List of binary values specifying, for each position, if it is harbored within an intron in the target gene (1) or not (0). The list must be in the same order and of the same length as option position. Example: [0, 0].

- **Edits**: KO: List of Ensembl gene ID of the knockout genes. Example: ["ENSG00000087263.17"]. The gene IDs must be present in the annotation file specified in option annotations_gtf. For knockin editing experiments, please use the Ensemble gene ID of any gene overlapping the edited sites.

- **VariantBasedScreening**: expand_search: Expand the PAM search up to *n* nucleotides from the cut site. Default: 2.

- **ExpressionBasedScreening**: len_promoter: Length of the upstream region of a gene to be considered as promoter. Default: 1000.

2 Software dependencies

The pipeline requires Snakemake>=5.32.00 and conda>=4.8.5. The pipeline was tested in a x86_64 GNU/Linux environment with Ubuntu v.18.04.1 installed. All other software requirements are satisfied by the Conda environments defined in Snakemake, except for the installation of R packages, for which we did not find a suitable combination of package's versions available for installation via Conda and satisfying all of the other dependencies in the "py3.yaml" environment. The installation of R packages is directed by the script 0.0_install_R_pkgs.R, in the CRISPRroots/scripts folder. The packages are installed in the R library related to the conda environment "py3", defined in the "py3.yaml" environment file in the folder CRISPRroots/envs. The R library path related to this conda environment (object name in R script: conda_R_libpath) is searched in the working directory in which the pipeline is executed under the path .snakemake/conda/ and set as the only R library path in the pipeline before installing other packages. In this way we avoid interfering with any other R environments (.libPaths) on the same machine. The conda_R_libpath is then set as environment before loading any package in the R scripts.

3 Pipeline usage examples

After completing the configuration file, the pipeline can be executed from within the same folder containing the config.yaml file with the command:

```
snakemake -s [path_to_CRISPRroots]/run.smk --use-conda
```

We suggest to precede this with a dryrun (--dryrun), which displays what passages will be executed without actually starting them. If you need to first discover the library type of your sequencing data, first run the pipeline with get_lib_type as target rule (see instructions below).

To only execute a part of the pipeline a target rule can be specified as:

```
 snakemake -s [path_to_CRISPRroots]/run.smk --use-conda preproc_and_map
```

The rule preproc_and_map only executes the pre-processing and mapping steps.
Other main target rules are:

- variants_to_genome: executes all of the rules necessary to produce files containing filtered variants (*vcf*) between each sample and the reference genome. Output in:

  ```
  [path_to_results_folder]/6_GATK_variants/[sample name]/variants_filtered.vcf
  ```

- eSNPKaryotyping: Executes the R package eSNP-Karyotyping [1] for the analysis of genome integrity from RNA-seq. The standard workflow is modified to employ reads mapped with STAR [2] instead of TopHat2 [3]. Output in:

  ```
  [path_to_report_folder]/eSNPKaryotyping/
  ```

- on_target_check: executes the on-target editing assessment. Output in:

  ```
  [path_to_report_folder]/on_target_knockin.xlsx
  [path_to_report_folder]/on_target_knockout.xlsx
  ```

- get_variated_genome: produces a variant-aware version of the reference genome, in which variants discovered from the RNA-seq are introduced in the reference sequence. Output in:

  ```
  [path_to_results_folder]/6_GATK_variants/variated_genome.fa
  ```

- get_lib_type: assesses the library type with RSeQC. Output in:

```
    [path_to_results_folder]/2-2_RSeQC_libtype/
```

To avoid removing output files defined as temporary (*e.g.* partially processed reads) while executing only a part of the pipeline use the option `--notemp`. Otherwise, temporary files are removed and will need to be recreated if required by a subsequent execution of the pipeline.

Workload manager systems can be used in combination with Snakemake. An example of a configuration file for execution on a computer cluster managed with Slurm is provided in the test folder.

Please consult the Snakemake manual at `https://snakemake.readthedocs.io` for further instructions on how to run a Snakemake pipeline or type `snakemake -h` for help in the command line.

4 Test folder

A test folder can be downloaded at `https://rth.dk/resources/crispr/crisprroots/downloads/CRISPRroots_test_dataset.tar.gz`. The folder contains:

- A sub-folder, `QPRT_DEL268T_chr16_10M-40M`, with:
  - sub-folder SAMPLES with reads from the GEO bioproject GSE113734 (del268T_rep1-3 and eCtrl_rep1-3 samples) mapping between 10M to 40M bases in chr16 (data published by Haslinger *et al.*, *Mol Autism* (2018)).
  - file `config.yaml` file, that needs to be modified by correcting the paths to that of a local directory.
  - file `cluster_config.yaml` file, as an example of how to set up the pipeline to work with a workload manager system (Slurm in this case).
  - file `gRNA_plus_PAM.fa`, that contains the gRNA used by Haslinger *et al.*
  - table `samples_table.tsv` describing the test samples
- A `resources` folder that includes a subset of the resources in the reference files (see section below) dedicated to chr16.

After adjusting the PATHs in the configuration file, enter the test sub-folder `QPRT_DEL268T_chr16_10M-40M` and executed the pipeline with the following command:

```
snakemake -s [path_to_CRISPRroots]/run.smk --use-conda
```

To launch the pipeline in `Slurm`, enter the test sub-folder `QPRT_DEL268T_chr16_10M-40M`; assuming that the `CRISPRroots` folder is under the local path `../../CRISPRroots`, run the following command:

```
[path_to_CRISPRroots]/cluster_run.sh
```

The path from the test folder to `run.smk` is hard-coded in `cluster_run.sh`, and needs to be modified for a path different from `../../CRISPRroots`. You can add the name of a target rule to run only a part of the pipeline as described in the execution examples above. In this case, run:

```
[path_to_CRISPRroots]/cluster_run.sh [target_rule]
```

References

[1] Weissbein, U., Schachter, M., Egli, D., and Benvenisty, N. (July, 2016) Analysis of chromosomal aberrations and recombination by allelic bias in RNA-Seq. *Nature Communications,* **7**(1).

[2] Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T. R. (October, 2012) STAR: ultrafast universal RNA-seq aligner. *Bioinformatics,* **29**(1), 15–21.

[3] Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R., and Salzberg, S. L. (2013) TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biology,* **14**(4), R36.