crispRtool_v2.0.1.Rscript Documentation

Samuel Lessard, Lettre Lab

October 3rd 2017

1 Overview

This scripts aims to find potential off-target hits in CRISPR/Cas9 experiments. Off-target scores are calculated as in Sanjan NE, Nature Methods 2014, and based on experimentally derived mismatch scores from Hsu et al, Nature biotechnology 2013. It also interogates each match using the Cutting Frequency Determination (CFD) score from Doench et al, Nature Biotechnology 2016.

This script takes a FASTA file containing single-guide RNA sequences and searches a genome for any potential off-target matches adjacent to a user-specified PAM sequence. It reports the overall off-target score, as well as the number of off-targets. The script also creates a file with the sequence and position of the best matches for each guide.

Multiple PAMs can be inputted. In this case, the script will search for off-targets adjacent to any of these motifs. The script can also be used to design non-targeting guides by creating and testing random sequences.

A VCF file can be provided to mask SNP positions using IUPAC ambiguity codes.

2 Dependencies

- R (tested on version 3.2.2)
- Libraries:
  - BSgenome
  - Biostrings
  - optparse
  - VariantAnnotation
  - reshape2

3 Usage

Rscript crispRtool.Rscript [options]

4 Options

-i INPUT, --input=INPUT

Input file of sgRNAs in FASTA format. Sequences must be 20bp long and PAMs should not be included. If this option is omitted, random sequences will be analyzed.

-o OUTNAME, --outname=OUTNAME

Output base name. Default = 'output'

-p PAM, --pam=PAM

PAM sequence [default "NGG"]. Multiple PAMs can be analyzed by passing a comma-separated list of motifs (eg. NGG,NGA).

-s OUTSCORE, --outscore=OUTSCORE

Minimum individal score for off-targets to be outputed [default 5].

-n NMM, --nmm=NMM

Maximum number of mismatches allowed in off-targets [default 4].

-a ASSEMBLY, --assembly=ASSEMBLY

Masked genome to be searched for off-target [default BSgenome.Hsapiens.UCSC.hg38.masked]. ASSEMBLY refers to the name of a BSgenome package to use. Active masks are for assembly gaps and intra-contig ambiguities. Repeats are not masked. The script will try to install the package if it is not installed.

-r RANDOM, --random=RANDOM

Number of random guides to analyze [default 5]. Ignored if input argument is present.

-A VCF, --ambiguous=VCF

Replace ACTG letter by IUPAC ambiguitie codes at SNP locations. Only SNPs are supported. Arguments correspond to the name of a two-column file with header: first column is chromosome names and second is the corresponding VCF files locations containing SNP information. Default is NA. Make sure VCF is on the same build as the assembly used.

-c CFD_REF, --cfd=CFD_REF

Path to CFD matrix file. Default is "CFD_REF.format". This file is provided with this script.

-m INTEGER, --max_ambiguities=INTEGER

Maximum number of ambiguities allowed in matches. Default is 4.

-h, --help

Show this help message and exit

5 Output

The script will output 2 different files:

5.1 .matches.txt

This files contains information on the genomic matches for each guides. It contains the following columns:

1. sgRNA_ID: sgRNA ID
2. sgRNA_Seq: sgRNA sequence
3. chr: Chromosome of match
4. start: Start position of match
5. end: End position of match
6. strand: Strand of match
7. match_Seq: Sequence of match
8. score: Individual score of match
9. CFD: CFD Score
10. mismatches: Number of mismtaches

5.2 .summary.txt

This files contains information on the total number of matches and overall score of each guides. It contains the following columns:

1. sgRNA_ID: sgRNA ID
2. sgRNA_Seq: sgRNA sequence
3. nontargeting_score: Non-targeting guide score. This score assumes that the sgRNA should NOT match any sequence of the genome.
4. best_match_position: Position of best match
5. best_match_score: Score of best match
6. best_match_CFD: CFD Score of best match
7. targeting_guide_score: Targeting guide score. This score assumes that the sgRNA should have one genomic match that is the intended target. If there is no perfect match, this score will be equal to the non-targeting score.
8. mean_CFD: Mean CFD Score

9. median_CFD: Median CFD Score

10. max_CFD: Maximum CFD Score

11. min_CFD: Minimum CFD Score

12. sd_CFD: Standard deviation of CFD Score

13. perc_CFD: 10,25,75, and 90th percentile of CFD score.

14. Total_matches: Total number of matches

15. N_X: These columns contains the number of matches with X mismatches. There will be X+1 columns ranging from N_0 (perfect match) to N_X, where X is the maximum number of allowed mismatches.

6 Supplementary analyses

We provide 2 supplementary scripts, which can be used to investigate specific haplotypes from genetic datasets. The main script is *search_haplotypes.Rscript*:

Rscript search_haplotypes.Rscript [input gRNA fasta] [VCF file] [chromosome] [haplotypes] > [haplotype_output]

The first agrument, <input gRNA fasta> is a fasta file containing gRNA sequences to test. The second file is a VCF file containing genetic variants to test. Haplotypes will be derived from the VCF and local sequences will be constructed based on each different haplotypes. The script can test both indels and SNPs. VCF files should be separated by chromosome and must be phased. The third argument is the chromsome to test. The last argument is a 3-column, tab-separated file describing each region to test with the following header: "chr start end". If this argument is omitted, the script will test all variants contained in the VCF file. In this case, each haplotype will consist of clusters of variants that are spaced 22bp or less apart. Note that the script may take several hours to run.

The script *sampler.Rscript* can be used to relate each haplotype outputed by the *search_haplotypes.Rscript* script to specific samples contained in the VCF. Samples are separated into two haploid genomes. Usage of this script is:

Rscript sampler.Rscript [haplotype_output] [VCF file] [chromosome]