SigProfilerMatrixGenerator

Sept 19, 2018

INTRODUCTION

The purpose of this document is to provide a guide for using the sigProfilerMatrixGenerator framework to generate mutational matrices for a set of samples with associated mutational catalogues.

PREREQUISITES

The framework is written in PYTHON, however, it also requires the following additional software with the given versions (or newer) and access to BASH:

PYTHON	version 3.4 or newer
-PANDAS [MODULE]	version (any)
WGET	version 1.19

By default, the installation process will save the FASTA files for all chromosomes for the default genome assemblies (GRCh37, GRCh38, mm10, mm9). As a result, XXX Gb of storage must be available for the downloads.

QUICK START GUIDE

This section will guide you through the minimum steps required to create mutational matrices:

1. Run the install_catalogue_ref.py script in python3.

2. Once the installation is completed successfully, matrices can now be generated.

3. Create a new folder for each project/job that you run within the *references/vcf_files/* folder. Use a unique name for each project.

4. Separate your INDEL mutations from your SNV mutations if they are present in the same files, and create a folder for each mutation type (ex: *references/vcf_files/[project]/SNV/* or *references/vcf_files/[project]/INDEL/*).

5. Place your *vcf* files within these new folders (either *references/vcf_files/[project]/SNV* or *references/vcf_files/[project]/INDEL/*).

6. From within the *scripts* folder, run the *sigProfilerMatrixGenerator.py* script in python3, specifying the reference genome, project name, indel (optional), and exome (optional) if desired:

```
python3 sigProfilerMatrixGenerator.py –g GRCh37 –p BRCA –e -i
```

*NOTE: only include –e if you would like to generate matrices using only mutations found within the exome, and only include –i or -ie if you would like to create the matrix for INDELS. (See available commands below).

7. The final matrices and the strand bias test results will be placed within the *references/matrix/[project]/* folder. By default, the script will generate the matrices for 6, 12, 96, 192, 1536, 3072, and DINUC context. The strand bias test results are generated for the 12, 192, and 3072 context.

8. Final log files will be placed within the *logs/* folder.

COMMANDS

-g or --genome -> required: Followed by the reference genome (ex: GRCh37, GRCh38, mm10, mm9).
-p or --project -> required: Followed by the project name (ex: BRCA).
-e or --exome -> optional: Creates the matrix based solely on the exome.
-i or --indel -> optional: Creates the matrix for the limited list of INDELs (classifies insertions at micro-homologies as insertions at repeats of 0).
-ie or –extended_indel -> optional: Creates the matrix for the complete list of INDELs.
-b or --bed -> optional: Followed by the name of the BED file. Creates the matrix based on a user supplied list of chromosome ranges saved in a BED file. This file must be saved under the "*/references/vcf_files/BED/[project]/*" folder.
-ch or --chromosome -> optional: Creates a single matrix for each chromosome

FOLDER STRUCTURE

The base framework consists of a *script*s folder, a *transcripts_original* folder, an *exome/* folder, a *BRCA_example* folder, an *example_test* folder, the *install_catalogue_ref.py* script, and this README file. The *scripts* folder contains all of the python scripts necessary to generate the reference files for the matrix generation as well as the *sigProfilerMatrixGenerator.py* script. The *transcripts_original* folder contains all protein-coding transcripts for each reference genome that is currently supported. These are necessary for generating the transcriptional strand bias matrices. The *exome* folder contains the exome BED files for the supported genomes. The *BRCA_example* folder contains an example simple text file format (see EXAMPLE). The *example_test* folder contains 10 example vcf files.

After running the *install_catalogue_ref.py* script, there will now be a *references* folder that contains all of the necessary reference files for creating the matrices. The user must place their input files within the *vcf_files* folder.

INPUT FILE FORMAT

This tool currently supports *maf, vcf,* and *simple text* file formats. The user must provide variant data adhering to one of these three formats. If the user's files are in *vcf* format, each sample must be saved as a separate file.

SUPPORTED GENOMES

This tool currently supports the following genomes:

GRCh38.p12 [GRCh38] (Genome Reference Consortium Human Reference 37), INSDC Assembly GCA_000001405.27, Dec 2013. Released July 2014. Last updated January 2018. This genome was downloaded from ENSEMBL database version 93.38.

GRCh37.p13 [GRCh37] (Genome Reference Consortium Human Reference 37), INSDC Assembly GCA_000001405.14, Feb 2009. Released April 2011. Last updated September 2013. This genome was downloaded from ENSEMBL database version 93.37.

GRCm38.p6 [mm10] (Genome Reference Consortium Mouse Reference 38), INDSDC Assembly GCA_000001635.8, Jan 2012. Released July 2012. Last updated March 2018. This genome was downloaded from ENSEMBL database version 93.38.

GRCm37 [mm9] (Release 67, NCBIM37), INDSDC Assembly GCA_000001635.18. Released Jan 2011. Last updated March 2012. This genome was downloaded from ENSEMBL database version release 67.

SIMULATING ADDITIONAL GENOMES

If the user desires to use a genome other than those currently supported (GRCh37, GRCh38, or mm10), they must:

1) Download the individual FASTA files for each chromosome
2) Place them into the *references/chromosomes/fasta/[genome]/* folder
3) Download the transcriptional data following the format presented in the *transcripts_original* folder.
4) Place this file within the *transcripts_original* folder
5) Download the exome ranges for the given genome and place in the *references/chromosomes/exome* folder.
6) Once you have performed the initial install for the baseline genomes, rerun the installation script and specify the name of the new genome:

```
python3 install_catalogue_ref.py –g mm37
```

EXAMPLE

Within the *references/vcf_files/BRCA_example/* folder, there is a simple text file (*BRCA_example_subs_simple.txt*) with four breast cancer sample and their associated mutational catalogues saved within a single file. To see an example of the output, run the following command from within the *scripts* folder:

```
python3 sigProfilerMatrixGenerator.py –g GRCh37 –p BRCA_example
```

The final matrices will be saved within the *references/matrix/* folder. To generate the INDEL matrix, place the INDEL *vcf* file within the *references/vcf_files/[project]/INDEL/* folder and run the command:

python3 sigProfilerMatrixGenerator.py –g [genome] –p [project] –i

LOG FILES

All errors and progress checkpoints are saved into *sigProfilerMatrixGenerator_[project]_[genome].err* and *sigProfilerMatrixGenerator_[project]_[genome].out*, respectively.
For all errors, please email the error and progress log files to the primary contact under CONTACT INFORMATION.

COPYRIGHT

CONTACT INFORMATION

Please address any queries or bug reports to Erik Bergstrom at ebergstr@eng.ucsd.edu