SNP-O-MATIC manual

SNP-O-MATIC is software to analyse Solexa reads and compare them to a reference. The core application, *findknownsnps*, looks at perfect matches of single and paired Solexa reads to a given reference sequence. While it will only find perfect matches, it can allow for a list of known SNPs, turning it into a Solexa-based genotyping application.
SNP-O-MATIC works by first creating an index on the reference sequence, then looking up reads in that index.
Most output files are tab-delimited. Parameters have to use the "--" prefix and must be separated from the parameter value by "="; a space will not do.

Parameters

Genome

--genome=FILE [mandatory]
The FASTA file containing the "reference genome" (can also be contigs etc.). This must be present for all operations of SNP-O-MATIC.

--noshortcuts
By default, kmers that occur very often (for example, in repeat regions) are removed from the index, speeding up read scan considerably. This option causes all kmers to be processed, resulting in a slower but more thorough process.

--index=FILENAME
The genome index is usually generated on the fly. To save time, such an index can be stored in a file and read in for subsequent use. Points to consider:

- The index file can get quite large
- The index file varies with the following parameters:
  --mspi, --noshortcuts, --index1, --index2, --snps, --gff
  SNP-O-MATIC will not check if the parameters used to create the index are the same that are used in a subsequent run. If any of these parameters differ, it might segfault or return wrong results without warning.

--mspi=NUMBER
The maximum number of SNPs per chromosomal index. The default value is 8.
When looking for known SNPs, each kmer (default: 26; see --index1 and --index2) is indexed in all possible combinations. If many potential SNPs occur in such a kmer, the number of indices rises exponentially (worst case: 26 "N"s => 4^{26} indices), slowing down index generation and using large amounts of memory. Also, matches will get unspecific. Therefore, no kmers with more than NUMBER SNPs are indexed.

--index1=NUMBER
--index2=NUMBER
The length of the internal index keys for genome indexing. By default, index1 is 10 and index2 is 16, resulting in a kmer index length of 26. Neither index must be 0, nor must index2 exceed 16. It is usually not necessary to alter these settings.

--mismatch=NUMBER
The number of mismatches allowed outside the index (index1+index2)

--chromosome=NAME
Discards all chromosomes but NAME prior to run, effectively only working on that chromosome.

--index_from=NUMBER
Starts indexing at this position on all chromosomes. To be used with *--chromosome*.

--index_to=NUMBER
Stops indexing at this position on all chromosomes. To be used with *--chromosome*.

Reads

--fasta=FASTA_FILE (*--fastq2=FASTQ_FILE*)
--fastq=FASTQ_FILE
The file(s) containing the reads. Use only one or the other. Use fastq together with fastq2 for split files.
One is mandatory, unless --uniqueness is used.

--nono=FILENAME
A file with names of reads to ignore.
One read name per line; names may be prefixed with ">" or "@".

Paired reads

Paired reads have several additional options:

--pair=NUMBER
The length of the first read. Mandatory for paired reads.

--fragment=NUMBER
The fragment length (including the two read pairs). The default is the entire chromosome.

--variance=NUMBER
The allowed variance for the fragment length. If not specified, ¼ of the fragment length is used. If the fragment length is not specified either, the entire chromosome length is used.

--chop=NUMBER
If one read but not the other matches, shorten the other by NUMBER bases.

--multimatch
If a read pair matches in multiple locations, assign it to one of them at random.

--singlematch
Only performs additional output functions (not bins) for single matches [currently paired reads only]

--foum
"Find One Unique Match" – will only map a read pair if at least one of the reads is unique within the genome.

Known SNPs

--snps=SIMPLE_SNP_FILE
--gff=GFF_SNP_FILE
A file containing a list of known SNPs.
The "simple SNP" format contains one known SNP per line:
chromosome_name –tab– position_in_chromosome –tab– reference_allele –tab– snp_allele_or_iupac_code
The GFF format is more complicated; ignore it.

Bins

--bins=FILE_PREFIX
Reads can be written into different "bin" files, depending on how they match the reference:

- FILE_PREFIX_no_match.fast[a|q] for reads that do not match the reference genome
- FILE_PREFIX_single_match.fast[a|q] for reads that match the reference genome exactly once
- FILE_PREFIX_multi_match.fast[a|q] for reads that match the reference genome more than once
- FILE_PREFIX_iupac.fast[a|q] for reads that contain IUPAC codes (e.g., "N")

The file ending will be ".fasta" or ".fastq", depending on the original format of the reads.

--binmask=BINARY_MASK
Use to create only certain bin files. For each bin type, use a "1" to create it and a "0" to prevent its creation, in the order of the above list. The default is "1111", that is, all bins are created.

Misc output

--uniqueness=OUTPUT_FILE
Output a uniqueness data file for the reference. As it acts on the reference sequence only, no read file has to be supplied. The option activates --noshortcuts.

--pileup=OUTPUT_FILE
Outputs a detailed pileup view of the reads. The maximum pileup depth is 70, deeper pileups are cut off. Quality scores are shown when using a fastq read file.

--snpsonly
When using --pileup, only output lines with SNPs.

--coverage=FILENAME
Outputs how many As, Cs, Gs, and Ts are found at each position. Less detailed than --pileup, but works for unlimited depth.

--cigar=OUTPUT_FILE
Outputs read alignments in CIGAR format.

--gffout=OUTPUT_FILE
Outputs read alignments in GFF format.

--fragmentplot=OUTPUT_FILE
Outputs a plot of fragment size distribution.

--snpsinreads=OUTPUT_FILE
Outputs a list of found known SNPs (that is, either reference or alternate allele). Each line contains a single found SNP, the name of the read it was found in, and miscellaneous metadata and quality scores. A header line explains the individual columns.

--indelplot=OUTPUT_FILE
Outputs the fragment length found for each match. For paired reads only.
See also --fragment and --variance.

--inversions=FILENAME
For paired reads, output a list of possible inversions. A large fragment/variance is recommended, as inversions would appear to be far apart.

--wobble= OUTPUT_FILE
Outputs a list of possible variations (paired reads only)

--wobblemax=NUMBER
Maximum number of mismatches allowed for wobble-match

--sqlite=FILENAME
Creates a sqlite text file with alignment data

--sam=FILENAME
Creates a SAM alignment file

--spancontigs=FILENAME
Outputs read pairs where "half" reads map uniquely to different contigs

Experimental

The following options are under development. It is not recommended to use them unless you know what you're doing.

--regions=REGION_FILE
Region file for finding new SNPs

--memory_save=NUMBER
Indexes the genome every NUMBER of positions; saves memory and runtime, but may have strange side effects.

Examples

Example 1

- Reference genome in `ref.fasta`
- Fastq reads in `reads.fastq`
- Reads are `36` bases, fragment size is `200-300`
- Task : Get all the reads that do *not* match perfectly within the fragment size

Command:

```
findknownsnps --genome=ref.fasta --fastq=reads.fastq --pair=36
              --fragment=250 --variance=50 --bins=out --binmask=1000
```

This will create `out_no_match.fastq`, containing the non-matching reads.

Example 2

- Reference genome in `ref.fasta`
- Fastq reads in `reads.fastq`
- SNP list in `snps.tab`
- Reads are `36` bases, fragment size is `200-300`
- Task : Get calls for all the SNP positions

Command:

```
findknownsnps --genome=ref.fasta --fastq=reads.fastq --pair=36
              --fragment=250 --variance=50 --snpsinreads=snps.out
```

This will create `snps.out`, containing all instances of a read covering a position from the `snps.tab` file.