kmc_tools documentation

for v. 2.3.0

Contents

Introduction

This document contains description of kmc_tools software. kmc_tools is a program which allows to work easily with sets of k-mers and their counters generated as output of KMC. KMC is an efficient k-mer counter described in: http://bioinformatics.oxfordjournals.org/content/early/2015/02/18/bioinformatics.btv022. kmc_tools can work with databases produced by KMC2 as well as by KMC1 (there is a little difference between those formats). kmc_tools always generates results in KMC1 database format as it is a little faster (in sense of searching k-mers) than in case of KMC2 database.

1 kmc_tools usage

kmc_tools provides a number of operations that can be used to work with k-mer sets. The number of input sets is depended on operation itself. Configuration of kmc_tools is done via command-line parameters.

The general syntax is:
kmc_tools [global_params] <operation> [operation_params]

Global parameters are independent of operation type. There are:

- -t<value> — total number of threads (default: no. of CPU cores),
- -v — enable verbose mode (shows some information) (default: false),
- -hp — hide percentage progress (default: false).

Avaiable operations:

- intersect — intersection of 2 k-mer sets,
- kmers_subtract — subtraction of 2 k-mer sets,
- counters_subtract — subtraction of 2 k-mer sets,
- union — uinon of 2 k-mer sets,
- complex — complex set operations for 2 or more input k-mer sets,
- sort — sort k-mers in input set,
- reduce — filter out k-mers with counters below (abow) specified threshold,
- compact — store only k-mers without counters,
- histogram — produce histogram of k-mers occurrences,
- dump — produce ASCII representation of k-mers and counters,
- filter — filter out reads with too small number of k-mers.

First four operations are typical set operations with 2 input sets. They are described in the next section (with explanation of difference between kmers_subtract and counters_subtract). The 5th operation (complex) is also a set operation, but it is more flexible (see section 3). Next 5 operations work with single input set and are described in section 4. The last operation takes as input KMC database and set of reads (e.g. FASTQ files) and keep only reads that contains at least specified number of k-mers (see section 5).

2 Set operations

Sets operations are executed from the command-line as follow:
kmc_tools [global_params] <set_operation> <input1 [input1_params]> <input2 [input1_params]> <output [output_params]>

where:

- set_operation — one of: intersect, kmers_subtract, counters_subtract, union,
- input1, input2 — paths to the databases generated by KMC (KMC generates 2 files with the same name, but different extensions — here only name without extension should be given),
- output — path to the output database.

For each input there are additional parameters which can be set:

- -ci<value> — exclude k-mers occurring less than <value> times,
- -cx<value> — exclude k-mers occurring more of than <value> times.

If additional parameters are not given they are taken from the apropriate input database.
For output there are also additional parameters:

- -ci<value> — exclude k-mers occurring less than <value> times,
- -cx<value> — exclude k-mers occurring more of than <value> times,
- -cs<value> — maximal value of a counter.

If they are not specified they are deduced based on input parameters.
The following subsections describe all available operations with examples.

2.1 intersect

The output database will contain only k-mers that are present in **both** input sets. The counter value in the output database is equal to the lower counter value in the input.

example

kmc -k28 file1.fastq kmers1 tmp
kmc -k28 file2.fastq kmers2 tmp
kmc_tools intersect kmers1 -ci10 -cx200 kmers2 -ci4 -cx100 kmers1_kmers2_intersect -ci20 -cx150

2.2 kmers_subtract

The output database will contain only k-mers that are present in the first input set but absent in the second one. The counter value is equal to the value in the first input set.

example

kmc -k28 file1.fastq kmers1 tmp
kmc -k28 file2.fastq kmers2 tmp
kmc_tools kmers_subtract kmers1 kmers2 kmers1_kmers2_subtract -cs200

2.3 counters_subtract

The output database will contain only k-mers that are present in the first input set and have counters higher than the apropriate k-mers in the second set. For each k-mer the counter is equal to the difference between the counter in the first set and the counter in the second set.

example

kmc -k28 file1.fastq kmers1 tmp
kmc -k28 file2.fastq kmers2 tmp
kmc_tools counters_subtract kmers1 kmers2 kmers1_kmers2_counters_subtract

2.4 union

The output database will contain each k-mer present in both input sets. For the same k-mers in the first and the second input the counter in the output is equal to the sum of the values in the inputs.

example

kmc -k28 file1.fastq kmers1 tmp
kmc -k28 file2.fastq kmers2 tmp
kmc_tools union kmers1 -ci3 -cx70000 kmers2 kmers1_kmers2_union -cs65536

3 Complex operation

Complex operation allows to define operations for more than 2 input k-mer sets.

Command-line syntax:
kmc_tools [global_params] complex <operations_definition_file>

where operations_definition_file is a path to the file which define input sets and operations. It is a text file with the following syntax:

```
INPUT:
<input1> = <input1_db_path> [params]
<input2> = <input2_db_path> [params]
...
<inputN> = <inputN_db_path> [params]
OUTPUT:
<out_db_path> = <ref_input> <oper> <ref_input> [<oper> <ref_input> [...]]
[OUTPUT_PARAMS:
<output_params>]
```

where:

- input1, input2, ..., inputN — names of inputs used to define operation,
- input1_db_path, input2_db_path, inputN_db_path — paths of k-mer sets,
- out_db_path — path to output database,
- ref_input is one of input1, input2, ..., inputN, oper is one of {*,-,∼,+}, the meaning is as follow:
  - * — intersect (see section 2.1)
  - - — kmers_subtract (see section 2.2)
  - ∼ — counters_subtract (see section 2.3)
  - + — union (see section 2.4)

For each input there are additional parameters which can be set:

- -ci<value> — exclude k-mers occurring less than <value> times,
- -cx<value> — exclude k-mers occurring more of than <value> times.

If additional parameters are not given they are taken from the apropriate input database. Operator * has the highest priority. Other operators has equals priorities. Order of operations can be changed with parenthesis.

output_params are:

- -ci<value> — exclude k-mers occurring less than <value> times,
- -cx<value> — exclude k-mers occurring more of than <value> times,
- -cs<value> — maximal value of a counter.

If they are not specified they are deduced based on input parameters.

example

INPUT:
set1 = kmc_o1 -ci5
set2 = kmc_o2
set3 = kmc_o3 -ci10 -cx100
OUTPUT:
result = (set3 + set1) * set2

4 One input operations

One input operations take a single KMC database and produce the output which may be other (transformed) KMC database or text file depending on the operation type.

General syntax:
kmc_tools [global_params] <oper [oper_params]> <input [input_params]> <output [output_params]>

where:

- oper — one of: sort, reduce, compact, histogram, dump,
- input – path to databases generated by KMC (KMC generates 2 files with the same name, but different extensions — here only name without extension should be given),
- output — path to the output file.

For input there are additional parameters which can be set:

- -ci<value> — exclude k-mers occurring less than <value> times,
- -cx<value> — exclude k-mers occurring more of than <value> times.

If additional parameters are not given they are taken from the apropriate input database.

output_params are only available for sort and reduce operation:

- -cs<value> — maximal value of a counter.

If this parameter is not specified it is deduced based on input database.
oper_params is only avaiable for dump operation (see section 4.5)
The following subsections describe all available operations with examples.

4.1 sort

Output database will contain k-mers in the sorted order. This operation converts KMC2.x database to KMC1.x database.

example

kmc_tools sort out_kmc2 -ci3 -cx1000 out_kmc1 -cs255

4.2 reduce

Exclude too rare and too frequent k-mers. Output is in KMC1.x format.

example

kmc_tools reduce out_kmc2 -ci30 -cx1200 out_kmc1 -cs255

4.3 compact

Remove counters of k-mers. Output database will contain only k-mers (without counters).

example

kmc_tools compact out_kmc2 -ci1 -cx10000 out_kmc1

4.4 histogram

Produce histogram of k-mer occurrences as a text file.

example

kmc_tools histogram kmers_db -ci3 -cx1000 histo.txt

4.5 dump

Produce text dump of KMC database.
For this operation the optional operation_param is:

- -s — force sorted output (default: false).

For KMC1.x this parameter is irrelevant as k-mers are stored in sorted order and this order will be preserved in produced text file. For KMC2.x when this parameter is on k-mers will be sorted before dump to text file.

example

kmc_tools dump out_kmc2 -ci3 -cx1000 dump.txt

5 filter

This operation works on input FASTQ/FASTA files and a database produced by KMC. It removes from the input read set those reads which does not contain specified number of k-mers.

Syntax:
kmc_tools filter <kmc_input_db> [kmc_input_db_params] <input_read_set> [input_read_set_params] <output_read_set> [output_read_set_params]

where:

- kmc_input_db — path to database generated by KMC,
- input_read_set — path to input set of reads,
- output_read_set — path to set output of reads.

For k-mer database there are additional parameters:

- -ci<value> — exclude k-mers occurring less than <value> times,
- -cx<value> — exclude k-mers occurring more of than <value> times.

For the input set of reads there are additional parameters:

- -ci<value> — remove reads containing less k-mers than value,
- -cx<value> — remove reads containing more k-mers than value,
- -f<a/q> — input in FASTA format (-fa), FASTQ format (-fq); default: FASTQ.

For input set of reads integer or floating number can be given as -ci<value> and -cx<value>. Integer values are used to define strict thresholds, which means only reads that contain at least ci_{value} and at most cx_{value} k-mers will be kept in the output read set. Floating numbers for -ci<value> and -cx<value> parameters are used to define thresholds depending on read lenght. It should be in range [0.0;1.0]. Let r be a length of a read. The read will be kept in the output read set only if it contains at least $\lfloor (r-k+1) * ci_{value} \rfloor$ and at most $\lfloor (r-k+1) * cx_{value} \rfloor$ k-mers which are present in KMC database.

For the output set of reads there are additional parameters:

- -f<a/q> — output in FASTA format (-fa), FASTQ format (-fq); default: same as input

input_read_set may be a single file or a file which contains a list of input files (one file per line).

example

kmc_tools filter kmc_db -ci3 input.fastq -ci0.5 -cx1.0 filtered.fastq
kmc_tools filter kmc_db input.fastq -ci10 -cx100 filtered.fastq
kmc_tools filter kmc_db @input_files.txt -ci10 -cx100 filtered.fastq