Kraken - A Computational Universal Genomic Coordinate Platform for Comparative Genomics

Software User Guide

Overview

The emerging need for comparative analyses, involving increasing numbers of organisms poses major computational challenges. Here, we described *Kraken*, software that allows for comparing features across large numbers of genomes. *Kraken* efficiently computes syntenic relationships indirectly, i.e. by mapping regions through intermediate genomes, alleviating the need for direct pairwise genome alignments.

Getting started

Download the package, unzip and run make from the main directory. Use gcc version 4.2 or higher for this. You can start by running the code with the data provided in the samples directory. This is explained below.

Example

The following example performs a mapping of items in the GTF input file "dmel.gtf" from specie "dmel" to "dyak". The data for this example can be found in the sample folder. The output is written to mapped.gtf as default. You can run the command.

```
../runKraken -c dere_dyak_dmel.config -s dmel.gtf -S dmel -T dyak
```

Available arguments

Required

-**c**<string> : Configuration file
-**s**<string> : Source GTF file
-**S**<string> : Source genome id
-**T**<string> : Target genome id

Optional

-**T**<string> : Destination GTF file provided if comparison is required.
-**O**<string> : Output for the mapped results in GTF format (def=mapped.gtf)
-**o**<string> : Output from comparing mapped transcripts to destination GTF
-**l**<string> : Application logging file (def=application.log)
-**L**<bool> : Choose if final boundaries should be set by local alignment (def=0)

Input Parameters

The inputs for running Kraken are as follows:

1. Configuration file for specifying the names and paths of input genomes and available syntenic alignments among them.
2. Source file containing entries that the user wants to transfer onto a target genome (should be in GTF format).
3. Source genome name, which should match the genome name in the configuration file.
4. Target genome name (Name should be the same as the relevant entry in the configuration file).
5. Target annotation file to compare the mapped annotations with (Format should be GTF). This is optional and should be provided only if Kraken is being used to map an annotation and also a reference annotation is available to compare to the mapped results.

Configuration File

The configuration file should include the names and paths for the genomes in FASTA format and the synteny files, as shown in the given example. Note that the synteny alignments should be produced from the same FASTA files given in the configuration file. Below is an example of a configuration file.

```
[genomes]
dere       genomes/dere.fa
dmel       genomes/dmel.fa
dyak       genomes/dyak.fa

[pairwise-maps]
dere dmel   syntenies/dere_dmel.satsuma
dere dyak   syntenies/dere_dyak.satsuma

[multiple-alignments]
# This section would be used if multiple-alignments are available (XMFA format)
```

Pairwise Syntenic Alignments

LASTZ general output format with one line per alignment block and configurable columns. The columns that should be included are ones used by the Satsuma format and are as follows:

1) Name of target sequence

2) Target start coordinate
3) Target end coordinate
4) Name of query sequence
5) Query start coordinate
6) Query end coordinate
7) Score
8) Orientation of target vs. query.

Note, that all the above coordinates are assumed to be zero-based. Also, note that the score is a placeholder and is not used by *Kraken.* An example is shown below.

```
X    153927995    153928014    X    72250140    72250180    1.0    +
X    153928028    153928862    X    72250404    72251261    1.0    +
X    153939728    153941202    X    72269478    72270818    1.0    -
…
```

GTF Format

Input items should be in GTF format. The first column (seqid) must exactly match the sequence names in the FASTA files. The third column (type) determines how we uses the entry. The first row is an example of an annotation item entry and the second line is a generic item entry

```
Annotation  item entry
GL000213.1    protein_coding  CDS    138767  139287 .    -    0    gene_id
"ENSG00000237375"; transcript_id "ENST00000327822";

Generic item entry
GL000213.1   general_coordinate      coordinate_id     139285  139287 .    -    0
```

Output

1. Mapped results in GTF format
2. Output from comparing mapped transcripts to destination GTF

Notes

- The sequence names must *exactly* match those used in the annotation files (first column), or the annotations will not work.
- For annotation GTF files, the key for transcript and gene Ids is assumed to be gene_id and transcript_id (this is the case in most standard annotations)
- If multiple alignments are available, they can also be used, in place of or in conjunction with pairwise alignments. The format for the multiple alignment should be XMFA (eXtended Multi FASTA) and provided in the configuration file.
- The GTF format uses 1-based coordinates and the Satsuma format synteny maps uses 0-based coordinates.