alfie

ALFIE: python package for the acquisition of anonymous or anchor loci datasets!

Thank you for downloading alfie!

SUMMARY

1. Introduction

1.1. What is alfie?

Alfie (Anonymous/Anchor Loci FIndEr) is an open-source python package containing a pipeline that searches complete genome sequences for the maximum number of ideal anonymous loci (i.e., single-copy, neutral, and independent loci) or for anchor (AE/UCE) loci that also meet the single-copy and independence assumptions. It also extracts orthologous sequences from other genomes, performs multiple sequence alignments, and outputs ready-to-analyze datasets in commonly used formats.

These anonymous loci datasets are ideal for phylogenomic studies that use multi-locus coalescent analyses (e.g., phylogeography). AE- and UCE-anchor loci are best for phylogenomic studies involving moderate to deep divergences. The goal of alfie is to be a complete, user-friendly, and flexible phylogenomic loci dataset-generating software.

1.2. How it works?

Alfie utilizes a sequential modular approach, with each step of the pipeline being stored in a single module that can be executed with greater flexibility in standalone mode. Many of these steps require external programs, such as BLAST or CLUSTALW. To reduce the complexity of the steps the whole pipeline can be run automatically using the afie.py script.

The program contains two main modules: Anonymous loci (AL) finding module and an anchor (AE/UCE) loci finding module.

1.2.1 Anonymous loci

The AL finding module uses as input a single query genome in FASTA format and its associated GTF file. The user must also input one complete genome sequence (can be un-annotated) for each individual or species in the study. Alfie finds all anonymous regions distant from genomic features (> 200 kb distance, by default) to avoid the effects of direct or indirect natural selection (i.e., to satisfy the neutrality assumption).

The program also uses a distance filter (> 200 kb distance, by default) to select loci that are likely to be genealogically independent of other sampled loci (i.e., satisfy the independent loci assumption). Alfie performs an exhaustive search (i.e., it finds every possible anonymous regions for the specified parameters in the genomes analyzed).

After the initial target regions have been found, they are split in small loci (default 1 kb). Each piece is searched against subject and query genomes to filter duplications and find orthologous regions in all genomes.

The sequences are then aligned using ClustalW (for each locus) and datasets are output in NEXUS (single and multi-locus files), PHYLIP (single and multi-locus files), and FASTA formats (single files). These files can then be used in a variety of single and multi-locus phylogenomic analyses (e.g., species tree analyses).

The number of output anonymous loci will depend on the two distance filter settings (i.e., longer inter-locus distances = fewer output loci and vice-versa).

1.2.2 Anchor loci

To use the anchor (AE/UCE) loci finding module, the user must input genome sequences (can be un-annotated) for each individual or species in the study. Program then finds the locations of target AEs/UCEs in a reference human genome with a coordinate file that currently contains 512 vertebrate AEs[1] (included in package). The module retrieves flanking regions around each AE/UCE element with user-defined length (e.g., 500 bp). User also specifies distance (bp) between flanking sequences and their AEs/UCEs. In general, evolutionary conservation of nucleotide sites declines with physical distance from each AE or UCE core region (i.e., location in the center of each AE/UCE element). Thus, if the user would like more conserved sequences, then the distance from the AE/UCE should be minimal, whereas less conserved sequences can be obtained by using longer distance settings (e.g., > 500 bp away from the AE/UCE). Paired flanking sequences (i.e., candidate AE/UCE loci) are then saved in FASTA files.

The next step is very similar to the second step of the AL finding module: AE/UCE candidate

loci are used as query sequences in BLAST searches against target genomes. Single-copy loci are retained and subsequently aligned. A user-specified distance filter retains loci that are likely independent from other sampled loci. Each pair of AE/UCE flanking sequences is concatenated to form independent loci. Lastly, ALFIE outputs ready-to-analyze datasets in the aforementioned formats.

[1] 512 vertebrate AE coordinates source: Lemmon AR, Emme SA, Lemmon EM. 2012. Anchored hybrid enrichment for massively high-throughput phylogenomics. Syst Biol 61: 727–744.

1.3. What to use alfie for?

Alfie was developed to find hundreds to thousands of single-copy, neutral, and independent nuclear markers among whole complete genomes and facilitate population genomics studies. A usual user case would use four to ten closely related genomes (we recommend no more than 20 million years of divergence to allow precise ortholog assignment). As an example, we were able to find about 300 anonymous loci with orthologs in all four hominoid genomes used (human, chimpanzee, gorilla and orangutan).

It is strongly recommended that at least one of the genomes have been extensively studied and annotated, presenting a comprehensive GTF file that describes the precise location of the main features targeted by natural selection, such as genes and regulatory elements.

2. Installation

2.1. Downloading alfie

Users can download and install the entire source code from github (https://github.com/igorrcosta/alfie/archive/master.zip) or download the docker container with all the dependencies included (available soon). To install, simply extract the donwloaded repository file in a folder.

2.2. Installation requirements

Working versions for each required program can be found on the following links:

- BLAST (https://ftp.ncbi.nlm.nih.gov/blast/executables/legacy/2.2.26/)
- ClustalW (http://www.clustal.org/download/current/)
- PhyML (http://www.atgc-montpellier.fr/phyml/binaries.php)
- ModelTest (https://code.google.com/p/jmodeltest2/)
- Biopython (http://biopython.org/wiki/Download)

2.3. Testing your installation

You can test your installation by running alfie on the human and chimpanzee Y Chromosomes' test case. To do that, go to the alfie folder and run the following command:

```
$>python alfie.py -r test/homo_y.fasta -i test/pan_y.fasta -g test/y.gtf
```

This command might take more than a minute to finish, depending on your computer speed. A successful run will output 42 candidate loci in the test.fasta file, of which only loci 11, 27 and 40 will be selected and aligned.

3. Quick Run

To run alfie, you will need a reference genome and GTF file (there are several available at Ensembl (http://www.ensembl.org/info/data/ftp/index.html)) and some genomes to compare against the reference. Example command line:

```
$> python alfie.py –r "reference_genome.fasta" -i "genome2.fasta" "genome3.fasta" -g "annotation_file.gtf" -o "output_folder"
```

Here the ".fasta" represents the path to the genome files in FASTA format, "annotation_file.gtf" is the path to the GTF file and "output_folder" is where the loci will be saved.

This command will find several candidade loci in the reference genome, which will be stored at a test.fasta file. This candidade loci will be blasted against all genomes and the final loci will be saved in several formats.

4. Advanced parameters

While alfie can be executed with good results without any additional configuration, there are several options that can be implemented in your analysis:

Flag	Description
-r, --reference	Path to the reference genome FASTA file.
-i, --genomes	Path to the other genome files in the FASTA format.
-g, --gtf	Path to the GTF file relative to the reference genome.
-o, --outpath	Path where all output files will be saved.
-f, --skip_formatdb	Skip making BLAST databases, use databses from the last run. Can significantly improve analysis speed.
--uce_coordinate	File with the conserved elements' coordinates, alfie comes with an example file containing the coordinates for 512 vertebrate anchored elements in the human genome. Necessary for Anchored Loci search.
--uce_distance	Distance between the flanking locus and the beggining of the conserved element. Only used for Anchored Loci search.
--locus_length	Length of the anonymous loci (default 1000bp). You may increase this length if you are planning to predict primers for these loci.

Flag	Description
--max_n	Maximum percentage of N's in the AL sequence (default 0%). Increase this if you want to find more loci in a low quality reference genome.
--inter_distance	Minimum distance between ALs (defaut 200000bp).
--gene_distance	Minimum distance between ALs and genes (default 200000bp). You can use negative numbers to find loci close to the gene regions, for example, -2000 will find loci between 0-2000bp from genes.
--end_distance	Minimum distance between ALs and the telomeres (default 10000bp).
--gene_locus	Use this flag to find loci inside the gene regions (will ignore the gene_distance flag).
--cds	Only considers the CDS features of GTF files, ignoring all pseudogens, miRNA, etc.
--duplication_cutoff	ALs with 2 hits with identity higher than this will be considered duplicated(default: 50%).
--identity_cutoff	ALs with a identity higher than this will be considered homologous(default: 90%).
--coverage_cutoff	BLAST hits must have at least this much %coverage to be considered hits (default: 90%).
--chromossomes	Chromossomes to be excluded. We recommend excluding all sex chromossomes.
--min_align	Minimum final alignment length (default 900bp). This will exclude loci that have many Ns in genomes other than the reference.
--remove_gaps	Remove gaps from the final alignment.

5. Citing us

If you use this program, please cite the following paper:

Costa IR, Prosdocimi F, Jennings WB. 2016. In silico phylogenomics using complete genomes: a case study on the evolution of hominoids. Genome Res. Published in Advance July 19, 2016, doi:10.1101/gr.203950.115 (http://genome.cshlp.org/content/early/2016/07/19/gr.203950.115.abstract)