Applications Note

Sequence Analysis

AdaptSearch: A Galaxy pipeline for sequence comparison and the search of positive selection from orthologous groups derived from RNAseq datasets

Mataigne Victor[1,2], Le Corguillé Gildas[1], Berthelier, Charlotte[2], Fontanillas Eric[3], Baffard Julie[1], Brun Pierre-Guillaume[2], Monsoor Misharl[1], Corre Erwan[1] & Jollivet Didier[2,*]

[1] ABiMS Bioinformatic platform, FR2424 CNRS-SU, Station Biologique de Roscoff, Place Georges Teissier, 29680 Roscoff, cedex, France.

[2] DISEEM Team, UMR 7144 CNRS-SU, Station Biologique de Roscoff, Place Georges Teissier, 29680 Roscoff, cedex, France.

[3] INGENOMIX, Pôle de Lanaud, Limoges, France.

*To whom correspondence should be addressed.

Abstract

Motivation: AdaptSearch is a tool dedicated to positive selection detection from orthologous groups of transcriptome sequences.

Results: This Galaxy workflow use as input transcriptome assemblies for *n* species and proceeds to find putative orthogroups. Multiple alignments are then produced with the corresponding full-length transcripts within each orthogroup. After finding the coding sequences in the correct frame and indels removal, the suite reconstructs a phylogenomic tree from a concatenated set of the coding-sequence alignments, and search for positively-selected genes along the branches of the tree and positively-selected codons along both the whole sequence alignment or each transcript, separately. In addition, AdaptSearch estimates base, codon and amino-acid frequencies from the selected set of genes and their products as well as all the substitutions from one species to another and tests *a priori* ecological hypotheses using a codon resampling procedure or binomial tests on the sequence composition and/or d_N and d_S substitution rates.

Availability: Implemented in Python and R. Freely installable on the Galaxy Tool_Shed. The Open-Source code is available on https://github.com/abims-sbr/adaptsearch. The functional workflow can be used on galaxy.sb-roscoff.fr (an ABiMS account is required).

Contact: abims@sb-roscoff.fr

1 Introduction

Over the past decade, a tremendous effort has been made in the production of massive amounts of sequences thanks to the help of the continuously improving Next Generation Sequencing (NGS) technology with the aim of performing *de novo* assembly and scaffolding of reads to obtain draft eukaryotic genomes or transcriptomes from DNAseq and RNAseq datasets [1]. Most of the effort has been then dispensed to develop methods for mapping reads and calling SNPs or analysing differential gene expression but pipelines able to treat a large number of assemblies for phylogenomics and evolutionary purposes are at their infancy. Before this period, several powerful gene-based softwares were also developed in order to detect positive selection based on tree reconstruction with sets of closely related species at the scale of a gene [2-4] but none of them were directly applicable to the scale of a genome. Detecting positively-selected genes or codon sites, their proportion and distribution in genomes, and their potential effects on their translated protein represents one of the most important challenges in species evolution and, apart from a few biological models including human and *Drosophila*, are still difficult to evaluate for the vast majority of metazoans. To that end, the Galaxy pipeline AdaptSearch has been developed from Python scripts of previous studies on deep-sea hydrothermal alvinellid worms living under contrasted thermal conditions [5], with the aim of assessing long-term evolution of transcriptomes and their translated proteins between closely-related species.

2 Methods

2.1 Pipeline

The AdaptSearch pipeline is made of a suite of scripts written in Python 2.7 and R (FigS1), managing inputs and outputs of several external tools. It takes as input a series of transcriptomes either assembled by Trinity or VelvetOases to select one variant per transcript, with a preprocessing of contigs using cap3 and FASTX-Toolkit [6] to reduce gene redundancy. Homologous sequences between species are gathered in orthogroups (Putative Orthologous Groups: POGs) using two distinct methods. In the first process, an exhaustive all-versus-all Blast tblastx with a Reciprocal Best Hit Criterion is run between the cured assemblies. Pairs of coding sequences sharing the greatest sequence identity are gathered in orthogroups according to the principle of transitivity. POGs are subsequently filtered to remove paralogs if the number of retrieved sequences is greater than that of the species compared. In the second process, OrthoFinder algorithm [7] is using blastp outputs performed on peptide sequences previously produced by TransDecoder. OrthoFinder works faster and separates outparalogs and inparalogs during the networking process. Multiple alignments are then performed on each orthogroup with BlastAlign [8], which eliminates indels and their flanking regions in coding sequences. The best coding-sequence and its translated sequence are then retrieved from each gene alignment. Orthologous groups (loci) are then concatenated according to the number of species selected by the user, and used to perform a phylogenomic tree with RAxML [9]. Finally, codeml [2] is called to compare different models of selection (Branch and Site models) with the most appropriate nearly neutral model using the RAxML topology tree as user tree. During this step, users can choose to perform the analysis either with the concatenated alignment of coding sequences or with each locus separately. In the specific way of testing each locus, the two selected models are compared over the whole set of gene alignments sequentially. All data files produced by codeml can be then processed with the Python suite ETE toolkit [10] for the calculations of likelihood ratio tests. Base, codon and amino acid frequencies and GC/purine contents are also estimated, as well as substitution rates between codons, amino acids and categories of amino acids within each gene alignment and the concatenated sequence. In addition, a Binomial test with ecological priors [5] is implemented to compare the set of gene variables found in each lineage to an *a priori* ecological group of reference (selected by the user according to a specific ecological/biological trait). Deviations to the null hypothesis of sequence homogeneity are also tested following a resampling bootstrap procedure of codons. The rationale to use binomial test with ecological priors is mainly due to non-applicability of tests assuming the Normal distribution of the datasets because of the very large inter-genic variability in the sequence composition and d_N/d_S ratios over transcriptomes.

2.2 Galaxy Integration

The AdaptSearch suite is available on Galaxy as a beta v2.0 on the ABiMS Galaxy's instance (http://galaxy.sb-roscoff.fr). It is available on the Tool Shed (https://toolshed.g2.bx.psu.edu/) for non ABiMS users. The last version uses Galaxy's dataset collections, a more suited way of accessing and manipulating datasets under Galaxy. The Galaxy wrappers for the external tools OrthoFinder and codeml have been written and validated by the Galaxy-IUC. All wrappers include functional tests, used to track the reliability of the results after any modification in the source code.

2.3 Visualization of results

R scripts have been integrated to the pipeline to produce histograms and heatmaps for observed *vs* expected frequencies of variables together with *p*-values. Tree visualization and exploration of codeml results and more specifically the positions of positively selected sites can be identified with the ease of ETE Toolkit [10]. This Python suite uses the codeml .res files to produce trees where branches under positive selection are highlighted and alignments where positively selected codons are located.

Several Shiny applications are currently under development for a subsequent integration under Galaxy via interactive environments in order to visualize datasets by using multivariate analyses.

Acknowledgements

This article has been written in memory of Christophe Caron, who initiated the transfer of Python scripts first written by EF to Galaxy.

Funding

This work was supported by CNRS and CG29 for funding the two post-doctoral years of EF and the two-years Master Thesis salary of VM.

Conflict of Interest: none declared.

References

[1] Hölzer, M. and Marz, M. (2019). *De novo* transcriptome assembly: a comprehensive cross-species comparison of short-read RNA-seq assemblers. GigaScience, 8, 1-16.

[2] Yang, Z. 2007. PAML4: a program package for phylogenetic analysis by maximum likelihood. *Mol. Biol. Evol.,* **24**, 1586-1591

[3] Pond, S.L.K. and Muse, S.V. (2005). HyPhy: hypothesis testing using phylogenies. In: *Statistical methods in molecular evolution,* pp. 125-181. Springer, New York, NY.

[4] Delport, W. et al. (2010). Datamonkey 2010: a suite of phylogenetic analysis tools for evolutionary biology. *Bioinformatics*, **26**(19), 2455-2457.

[5] Fontanillas, E. et al. (2016). Proteome evolution of deep-sea hydrothermal vent alvinellid polychaetes supports the ancestry of thermophily and subsequent adaptation to cold in some lineages, *Genome Biol. Evol.*, **9**(2), 279–296.

[6] Gordon, A. and Hannon, G. (2010). Fastx-toolkit. FASTQ/A short-reads pre-processing tools. *Unpublished http://hannonlab. cshl. edu/fastx_toolkit*.

[7] Emms, D.M. and Kelly, S. (2015). OrthoFinder: solving fundamental biases in whole genome comparisons dramatically improves orthogroup inference accuracy. *Genome Biol.,* **16**, 157.

[8] Belshaw R. and Katzourakis A (2005). BlastAlign: a program that uses blast to align problematic nucleotide sequences. *Bioinformatics*, **21**, 22-23.

[9] Stamatakis, A. (2014). RAxML version 8 : a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, **30**, 1312–1313.

[10] Huerta-Cepas, J. et al. (2016). ETE 3: reconstruction, analysis, and visualization of phylogenomic data. *Mol. Biol. Evol.*, **33**(6), 1635-1638.

AdaptSearch