Bioinformatics, 36(4), 2020, 1265–1266
doi: 10.1093/bioinformatics/btz706
Advance Access Publication Date: 27 September 2019
Applications Note

Genome analysis

EvolClust: automated inference of evolutionary conserved gene clusters in eukaryotes

Marina Marcet-Houben[1,2,†] **and Toni Gabaldón**[1,2,3,*,†]

[1]Centre for Genomic Regulation (CRG), Bioinformatics and Genomics department, The Barcelona Institute of Science and Technology, Barcelona 08003, [2]Health and Experimental Sciences Department, Universitat Pompeu Fabra (UPF), Barcelona 08003 and [3]ICREA, Barcelona 08010, Spain

*To whom correspondence should be addressed.

[†]Present address: Barcelona Supercomputing Centre (BSC-CNS), Barcelona 08034, Spain and Institute for Research in Biomedicine (IRB), Barcelona 08028, Spain

Associate Editor: Russell Schwartz

Received on March 13, 2019; revised on August 30, 2019; editorial decision on September 6, 2019; accepted on September 25, 2019

Abstract

Motivation: The evolution and role of gene clusters in eukaryotes is poorly understood. Currently, most studies and computational prediction programs limit their focus to specific types of clusters, such as those involved in secondary metabolism.

Results: We present EvolClust, a python-based tool for the inference of evolutionary conserved gene clusters from genome comparisons, independently of the function or gene composition of the cluster. EvolClust predicts conserved gene clusters from pairwise genome comparisons and infers families of related clusters from multiple (all versus all) genome comparisons.

Availability and implementation: https://github.com/Gabaldonlab/EvolClust/.

Contact: Toni.gabaldon.bcn@gmail.com or tgabaldon@crg.es

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Gene order in eukaryotic genomes tends to be poorly conserved through evolution (Dávila López *et al.*, 2010). Despite this trend, certain groups of genes remain close in the genome over long evolutionary distances, which suggests that selection acts to maintain their genomic co-localization. Genes within such conserved clusters may have functional links (Lee and Sonnhammer, 2003; Wisecaver *et al.*, 2014), and can be transcribed in a coordinated manner (Boutanaev *et al.*, 2002; Reimegård *et al.*, 2017). Comparative genomics can be used to uncover groups of genes that remain significantly closer than expected, despite extensive gene order shuffling. In this line, Gecko3 (Winter *et al.*, 2016) was designed to address such questions, but it saturates when a large amount of species is compared. Other programs such as i-ADHore (Proost *et al.*, 2012) or MCScanX (Wang *et al.*, 2012) are prepared to search for collinearity between genomes but do not focus on the presence of gene clusters. We developed EvolClust, an algorithm that detects groups of neighboring genes shared by compared genomes. Unlike the mentioned programs it is able to detect conserved gene clusters, and is prepared to perform large scale comparisons using hundreds of genomes. In addition it is not limited to search for specific types of gene clusters, such as program searching for secondary metabolite clusters (Khaldi *et al.*, 2010; Medema *et al.*, 2011). EvolClust has been tested in Linux and can be ported to other systems using the docker included docker file.

2 Algorithm

2.1 Cluster definition

In EvolClust a conserved cluster is defined as a group of neighboring genes whose homologous genes are also neighboring each other in a different genome. Genes with homologs in the clusters defined in the two genomes are deemed cluster genes. Cluster genes do not need to be strictly in the same order in the two genomes considered, and up to a user-defined number of non-cluster genes (i.e. without homologs in the corresponding cluster defined in the other genome) are allowed in between any two cluster genes. The cluster is also limited by the number of different homologous families it contains as to not have a cluster formed by repeats of the same family.

2.2 EvolClust pipeline

EvolClust is designed to run parts of the pipeline in a computing cluster, enabling the processing of a large number of genomes. For smaller databases it also provides an option to perform all steps with a single command. As an output EvolClust provides lists of inferred clusters grouped into families. One of the inherent drawbacks of EvolClust is that it detects conserved clusters by comparison against background gene order conservation. Therefore, it will not always be able to predict clusters found exclusively in closely related species that have overall conserved gene order throughout the entire genome.

EvolClust algorithm proceeds as follows (see Supplementary Material for details). The starting file is a list of families of homologous proteins. Based on pairs of homologous proteins all possible clusters between a pair of species is calculated. The algorithm used to calculate clusters can be found in Supplementary Figure S3. A conservation score (C) is calculated for all found conserved regions as follows: $C = [(m^2 + n^2)–(o^2+p^2)]/2(m+n+o+p)$; where m and n are the number of shared genes for each region (i.e. those with homologs in the other region), and o and p the genes specific to each region. C is calculated across the whole conserved region and iteratively for smaller subregions so as to account for all possible cluster sizes. The distribution of C values in each pairwise comparison provides a measure of the expected conservation of randomly chosen genes between two species.

Once the threshold values are calculated the algorithm iterates through the predicted regions and selects all possible clusters with C above the threshold. Clusters are predicted for each pair of species. Hence, some redundancy is expected. Such redundancy is lowered by collapsing significantly overlapping gene clusters predicted in a given species. Once all clusters are defined for all species, C values between all of them are calculated as explained above. These scores are then used to group clusters into families using the mcl algorithm (Enright *et al.*, 2002). A subsequent cleaning step trims clusters based on the average length of the family. Finally, the initial list of clusters is sourced again and homologous clusters are added to a family if they share, with any cluster of that family, 90% of homologous proteins and have a $C > 0.95$ indicating they were likely discarded because they were in closely related species.

3 Benchmarking

EvolClust was run over two datasets comprising 341 fungal and 145 insect genomes, respectively (see Supplementary Tables S1 and S2 for full list). The number of predicted clusters was 118 699 (12 120 families) and 28 116 (8778 families), respectively. This was congruent with the difference in the number of species between the two datasets. Clusters can be found in evolclustDB (https://evolclustDB.org). The fungal dataset was processed in 29 h whereas the insect dataset took 149 min to finish, the difference is due to differences in number of species and number of homologous pairs per species between the two datasets. Details and further discussion on scalability can be found in the Supplementary Material.

We compared EvolClust to Gecko3 (Winter *et al.*, 2016), i-ADHore (Proost *et al.*, 2012) and MCScanX (Wang *et al.*, 2012) in 10 sets of fungal genomes of increasing size (Supplementary Material). Note that i-ADHore and MCScanX search for collinear regions and not for conserved clusters and as such they do not distinguish between background gene order conservation and conserved clusters and report back any collinear segment they detect. EvolClust was the second most efficient program behind i-ADHore in the scalability tests, yet i-ADHore was not able to process our largest fungal set due to RAM constraints, something avoided in EvolClust thanks to the use of a computer cluster. Results of cluster families found in EvolClust were comparable to the ones observed in Gecko3 (see Supplementary Material for details).

To assess whether EvolClust is able to recall meaningful gene clusters, we obtained a set of known secondary metabolism gene clusters that are found in at least two fungal genomes (Supplementary Table S3). The final list contained 26 experimentally characterized cluster families that comprised a total of 97 individual gene clusters. EvolClust was able to recover 91% of the clusters, though for some of them the exact boundaries of the cluster were not detected. In total, 31 clusters (32%) were predicted with exact boundaries, while 57 (59%) had discrepancies with the defined benchmark cluster. However, boundary differences were generally low. The median number of extra genes was 1, while the median number of missing genes was 0. In 22 out of the 23 detected families, all clusters were correctly assigned to the same family. Only in one case were the clusters assigned to two different families.

We compared our results with those obtained in the same set of genomes when using the popular SMURF (Khaldi *et al.*, 2010) and ANTISMASH (Medema *et al.*, 2011) algorithms, which are designed to detect this specific kind of gene clusters (see Supplementary Tables S4 and S5, respectively). SMURF was able to find 79 (81%) of the clusters and ANTISMASH found 86 (88%). SMURF predicted 8 correct clusters and had a median of 8 additional genes, ANTISMASH had a single correctly predicted cluster and a median of 11 additional genes. In contrast EvolClust is able to detect the presence of most (91%) known conserved clusters.

Collectively, these results show that EvolClust provides an alternative method to calculate conserved gene clusters that are biologically relevant and is able to process massive amounts of data.

Funding

This work was supported by the Spanish Ministry of Economy, Industry and Competitiveness (MEIC) for the EMBL partnership and grants ‘Centro de Excelencia Severo Ochoa’ SEV-2012-0208 and BFU2015-67107 cofounded by European Regional Development Fund (ERDF); from the CERCA Programme/Generalitat de Catalunya; from the Catalan Research Agency (AGAUR) SGR857 and grant from the European Union’s Horizon 2020 research and innovation programme under the grant agreement ERC-2016-724173 the Marie Sklodowska-Curie grant agreement No H2020-MSCA-ITN-2014-642095. The group also receives support from a INB Grant (PT17/0009/0023–ISCIII-SGEFI/ERDF).

Conflict of Interest: none declared.

References

Boutanaev,A.M. *et al.* (2002) Large clusters of co-expressed genes in the Drosophila genome. *Nature*, **420**, 666–669.

Dávila López,M. *et al.* (2010) Analysis of gene order conservation in eukaryotes identifies transcriptionally and functionally linked genes. *PLoS One*, **5**, e10654.

Enright,A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, **30**, 1575–1584.

Khaldi,N. *et al.* (2010) SMURF: genomic mapping of fungal secondary metabolite clusters. *Fungal. Genet. Biol.*, **47**, 736–741.

Lee,J.M. and Sonnhammer,E.L.L. (2003) Genomic gene clustering analysis of pathways in eukaryotes. *Genome Res.*, **13**, 875–882.

Medema,M.H. *et al.* (2011) antiSMASH: rapid identification, annotation and analysis of secondary metabolite biosynthesis gene clusters in bacterial and fungal genome sequences. *Nucleic Acids Res.*, **39**, W339–W346.

Proost,S. *et al.* (2012) i-ADHoRe 3.0—fast and sensitive detection of genomic homology in extremely large data sets. *Nucleic Acids Res.*, **40**, e11.

Reimegård,J. *et al.* (2017) Genome-wide identification of physically clustered genes suggests chromatin-level co-regulation in male reproductive development in *Arabidopsis thaliana*. *Nucleic Acids Res.*, **45**, 3253–3265.

Wang,Y. *et al.* (2012) MCScanX: a toolkit for detection and evolutionary analysis of gene synteny and collinearity. *Nucleic Acids Res.*, **40**, e49.

Winter,S. *et al.* (2016) Finding approximate gene clusters with Gecko 3. *Nucleic Acids Res.*, **44**, 9600–9610.

Wisecaver,J.H. *et al.* (2014) The evolution of fungal metabolic pathways. *PLoS Genet.*, **10**, e1004816.