HAL
open science

Improving the reliability of genotyping of multigene families in non-model organisms

François Rousset

To cite this version:

François Rousset. Improving the reliability of genotyping of multigene families in non-model organisms. 2020, pp.100092. 10.24072/pci.evolbiol.100092 . hal-02453621

HAL Id: hal-02453621

https://hal.umontpellier.fr/hal-02453621v1

Submitted on 23 Apr 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Peer Community In Evolutionary Biology

Improving the reliability of genotyping of multigene families in non-model organisms

François Rousset based on reviews by Thomas Bigot, Sebastian Ernesto Ramos-Onsins and Helena Westerdahl

Open Access

Published: 23 January 2020

Copyright: This work is licensed under the Creative Commons Attribution-NoDerivatives 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nd/4.0/

A recommendation of:

Gillingham, Mark A. F., Montero, B. Karina, Wilhelm, Kerstin, Grudzus, Kara, Sommer, Simone and Santos, Pablo S. C.. **A novel workflow to improve multi-locus genotyping of wildlife species: an experimental set-up with a known model system** *(2020), bioRxiv, 376756, ver. 3 peer-reviewed by Peer Community in Evolutionary Biology.* 10.1101/638288

Submitted: 15 May 2019, Recommended: 22 January 2020

Cite this recommendation as:

François Rousset (2020) Improving the reliability of genotyping of multigene families in non-model organisms. *Peer Community in Evolutionary Biology, 100092.* 10.24072/pci.evolbiol.100092

The reliability of published scientific papers has been the topic of much recent discussion, notably in the biomedical sciences [1]. Although small sample size is regularly pointed as one of the culprits, big data can also be a concern. The advent of high-throughput sequencing, and the processing of sequence data by

opaque bioinformatics workflows, mean that sequences with often high error rates are produced, and that exact but slow analyses are not feasible. The troubles with bioinformatics arise from the increased complexity of the tools used by scientists, and from the lack of incentives and/or skills from authors (but also reviewers and editors) to make sure of the quality of those tools. As a much discussed example, a bug in the widely used PLINK software [2] has been pointed as the explanation [3] for incorrect inference of selection for increased height in European Human populations [4]. High-throughput sequencing often generates high rates of genotyping errors, so that the development of bioinformatics tools to assess the quality of data and correct them is a major issue. The work of Gillingham et al. [5] contributes to the latter goal. In this work, the authors propose a new bioinformatics workflow (ACACIA) for performing genotyping analysis of multigene complexes, such as self-incompatibility genes in plants, major histocompatibility genes (MHC) in vertebrates, and homeobox genes in animals, which are particularly challenging to genotype in non-model organisms. PCR and sequencing of multigene families generate artefacts, hence spurious alleles. A key to Gillingham et al.‘ s method is to call candidate genes based on Oligotyping, a software pipeline originally conceived for identifying variants from microbiome 16S rRNA amplicons [6]. This allows to reduce the number of false positives and the number of dropout alleles, compared to previous workflows.

This method is not based on an explicit probability model, and thus it is not conceived to provide a control of the rate of errors as, say, a valid confidence interval should (a confidence interval with coverage c for a parameter should contain the parameter with probability c, so the error rate 1- c is known and controlled by the user who selects the value of c). However, the authors suggest a method to adapt the settings of ACACIA to each application. To compare and validate the new workflow, the authors have constructed new sets of genotypes representing different extents copy number variation, using already known genotypes from chicken MHC. In such conditions, it was possible to assess how many alleles are not detected and what is the rate of false positives. Gillingham et al. additionally investigated the effect of using non-optimal primers. They found better performance of ACACIA compared to a preexisting pipeline, AmpliSAS [7], for optimal settings of both methods. However, they do not claim that ACACIA

will always be better than AmpliSAS. Rather, they warn against the common practice of using the default settings of the latter pipeline. Altogether, this work and the ACACIA workflow should allow for better ascertainment of genotypes from multigene families.

References

[1] Ioannidis, J. P. A, Greenland, S., Hlatky, M. A., Khoury, M. J., Macleod, M. R., Moher, D., Schulz, K. F. and Tibshirani, R. (2014) Increasing value and reducing waste in research design, conduct, and analysis. The Lancet, 383, 166-175. doi: 10.1016/S0140-6736(13)62227-8 [2] Chang, C. C., Chow, C. C., Tellier, L. C. A. M., Vattikuti, S., Purcell, S. M. and Lee, J. J. (2015) Second-generation PLINK: rising to the challenge of larger and richer datasets. GigaScience, 4, 7, s13742-015-0047-8. doi: 10.1186/s13742-015-0047-8 [3] Robinson, M. R. and Visscher, P. (2018) Corrected sibling GWAS data release from Robinson et al. http://cnsgenomics.com/data.html [4] Field, Y., Boyle, E. A., Telis, N., Gao, Z., Gaulton, K. J., Golan, D., Yengo, L., Rocheleau, G., Froguel, P., McCarthy, M.I . and Pritchard J. K. (2016) Detection of human adaptation during the past 2000 years. Science, 354(6313), 760-764. doi: 10.1126/science.aag0776 [5] Gillingham, M. A. F., Montero, B. K., Wihelm, K., Grudzus, K., Sommer, S. and Santos P. S. C. (2020) A novel workflow to improve multi-locus genotyping of wildlife species: an experimental set-up with a known model system. bioRxiv 638288, ver. 3 peer-reviewed and recommended by Peer Community In Evolutionary Biology. doi: 10.1101/638288 [6] Eren, A. M., Maignien, L., Sul, W. J., Murphy, L. G., Grim, S. L., Morrison, H. G., and Sogin, M.L. (2013) Oligotyping: differentiating between closely related microbial taxa using 16S rRNA gene data. Methods in Ecology and Evolution 4(12), 1111-1119. doi: 10.1111/2041-210X.12114 [7] Sebastian, A., Herdegen, M., Migalska, M. and Radwan, J. (2016) AMPLISAS: a web server for multilocus genotyping using next-generation amplicon sequencing data. Mol Ecol Resour, 16, 498-510. doi: 10.1111/1755-0998.12453

Reviewed by Thomas Bigot, 2019-12-16 13:13

The authors made a remarkable work on their manuscript.

It is now clear which software version is presented in the article, with which dataset. Some missing points were explained. The instructions were made more complete, and the code was stored on a permanent repository with a DOI. It is now possible to fully test it.

My suggestion to use a pipeline manager was discussed in the response and I agree the solution the authors chose (a homemade manager) is suitable for this kind of pipeline. Il find the fact the answers to the interactive pipeline questions are stored in the configuration file for next executions is a smart way to make it more user friendly.

This pipeline now totally reach the current standards of a bioinformatics tool, and to my mind is suitable for publication.

Some very minor remarks:

L37, L203, L392, L553, L558, L566, L609: naive still does not take an umlaut in English

Enumerations are made using different styles: 1) 2) L73 L77 ; 1.) 2.) L152 L153 and 1. 2. L175 L177. This may be standardized.

Reviewed by Sebastian Ernesto Ramos-Onsins, 2019-12-16 15:35

In this work, the authors propose a new workflow (ACACIA) for performing genotyping analysis of relatively complex muti-locus systems, addressed specially to non-model species. The authors realized of a number of problems in genotyping analysis of multi-locus systems (also detected and reviewed for other authors as referenced in the manuscript), such as MHC, and constructed a workflow in which is key to use a method that call candidate genes based on clustering redundant alleles from other divergent alleles, given the information contained at each position (Olygotyping tool). This workflow allows to reduce the number of false positives and the number of dropout alleles in relation to other available workflows. Although, this key process avoided a threshold decision, these kind of methodologies are not fully probabilistic, and therefore, a posterior

decision also make some errors in discarding possible true alleles. Nevertheless, I find a good and practical solution that improves existent methods.

The authors construct a new set of genotypes of different CNV in order to compare and validate the new workflow, using already known genotypes from chicken. Thus, it is possible to test for example, how many alleles are not detected and what is the rate of false positives. I find it correct and very informative about the possibilities of this methodology.

Finally, the authors have thought about all the suggestions given by previous reviewers and have included most of them. In my opinion, the manuscript and the software has greatly improved. I have no more suggestions.

Revision round #1

2019-07-16

I managed to obtain two reviews. One of the reviews highlights why this ms may be eventually worth recommending by PCI. Nevertheless, it also notes two important weaknesses, and the other review points additional important issues. I summarize these criticisms below to make clear the main revisions that appear required for the ms to be eventually recommended.

From the first review:

"ACACIA might be advantageous to the existing programs / workflows, [but] this is not really fully tested in the manuscript": comparisons should be provided.

"The authors should either have run all settings in one study data-set or one setting in all data sets (or all combinations for all data sets)." Here the issue is : what can be concluded from the different analyses? I guess that the authors will be able to partially rebut this question, but it is not clear what is meant by "test" on l. 183 ("test ACACIA in wildlife species with unknown genotypes of varying CNV").

The second review highlights that ACACIA is not yet really a "pipeline" but rather an interactive script. Most importantly, it expresses concerns about the repeatability of the analyses. I concur with this review that reproducible(s) example(s) should be provided. This review also implies that the version described in the ms should be made permanently accessible. I see the point but I am not sure it is the best way to address the issue of reproducibility. An alternative view is that future versions should be tested against the results of the current version, which brings us back to the issue of providing reproducible examples.

I hope the authors will be able to submit a revised version addressing all these points.

Preprint DOI: https://doi.org/10.1101/638288

Reviewed by Helena Westerdahl, 2019-06-26 16:00

Download the review (PDF file)

Reviewed by Thomas Bigot, 2019-07-10 17:14

This article presents a workflow to improve multi-locus genotyping. They propose an experimental set-up and a pipeline named Acacia to perform the genotyping itself. They chose chicken as a model organism, and try to characterize sequences of MHC B Complex with their tool.

The manuscript is well-written.

According to my skills, I will focus this review on the pipeline and its bioinformatics aspects.

The ACACIA pipeline

Description in the article

- The introduction (L 273) should mention Biopython as a dependency;

- Some step were coded ad-hoc, even being non-trivial (e.g. Trimming low quality ends). The reason why well known methods wer not used should be briefly explained.

- Input data is not explained. In the documentation, three input files are listed. One of them is *A fasta file with 100+ sequences related to those that you expect to have sequenced. This file will be used to setup a local BLAST database.*. This description is not clear and BLAST is not mentionned in the manuscript.

- FLASH and Pandas are used in the script but not mentionned in the manuscript.

The pipeline itself

Reproducibility of the code

I have a major concern about reproducibility: the only code available is the master branch of the git repository. If a user downloads the pipeline in the future, nothing can tell the code available at this time corresponds to the one described in this article, and nothing guarantees the code is still available on Github. Hence, I strongly suggest to:

- create a release number of the code (eg v1.0) and indicate this number in the article;
- create an archive of this release and upload it to zenodo or figshare (or any repository of this kind);
- get a DOI from them, and indicate it in the pipeline description.

Dataset: reproducibility of the analysis

I wish I could test the pipeline, but no example dataset is provided. Moreover, the article describes the analysis of a peculiar one (Chicken HMC), so it should be included. I suggest to upload it (fastq data, primers, “well known sequences”) at zenodo or figshare like explained just above, and indicate the DOI in the article, and in the documentation files as a testing procedure.

Pipeline manager

The program is an interactive script, asking questions to the user who has to wait during the whole time of the analysis. No argument can be provided to the pipeline at the launching time. The files must be at certain places with certain names. Moreover, all the steps are performed in one run: if one step fails, it has to be restarted from the beginning.

This script should be transformed as a real pipeline, using a dedicated software. As authors seem to have a good level of Python, I suggest them to choose Snakemake (https://snakemake.readthedocs.io/en/stable/). It is a Python tool: each step code chunk could be simply copied/pasted in the Snakemake recipie.

Other remarks

L 216, L 217: “Naive” and “naively” do have an umlaut in English.

Author's reply:

Download author's reply (PDF file)