PIPELINER: a tool to evaluate NGS pipelines and optimize experimental designs for resequencing studies

Bruno Nevado[1], Miguel Perez-Enciso[1,2,3]✉
[1]Centre for Research in Agricultural Genomics (CRAG), Barcelona, Spain
[2]Universitat Autononoma de Barcelona, Bellaterra, Spain
[3]Institut Catala de Recerca i Estudis Avancats (ICREA), Barcelona, Spain

Motivation and Objectives

The choice of technology and bioinformatics' approach is central in the analysis of Next-Generation-Sequencing (NGS) experiments. The pace with which new software and methodological guidelines are published, together with the fact that many of these choices will depend on the particularities of the study-system, mean researchers are often unable to produce informed decisions regarding these central questions. To address these issues, we introduce Pipeliner, a tool to simulate and validate the performance of NGS analysis pipelines, and optimize experimental designs.

Methods

Pipeliner is written in Object-Oriented Perl and is highly customizable, allowing the user to write and test his own bioinformatics' pipelines. A simulation is then performed for each pipeline defined, and statistics describing their performance in variant calling are calculated and reported.

The first step in the analysis performed with Pipeliner is to specify the experimental design, which includes defining the study system, i.e. number of individuals sequenced, population structure, depth and sequencing technology. Pipeliner uses coalescent simulations (with the software ms; Hudson, 2002) to obtain the "true" Single-Nucleotide Polymorphism (SNP) data for the population under study, allowing for specific conditions to be explored such as the effect of the distance between the sampled individuals and the reference genome available, the effect of population subdivision, or different levels of variability or selection. As for the NGS reads, Pipeliner uses the program ART (Huang *et al.,* 2011) to simulate illumina NGS reads (solid and 454 reads can also be simulated with ART), with the user defining the read length, the average depth per individual, paired or single ends run, etc.

Once the experimental design is defined, the next step is to choose the bioinformatics' pipeline with which to analyze the genetic data obtained. The three crucial decisions in this step are (i) how to align the short reads to the reference genome, (ii) how to call variants from the aligned short reads, and (iii) how to filter the variants obtained. Pipeliner implements a wrapper to the commonly used software bwa (Li and Durbin, 2009), and to samtools (Li *et al.,* 2009), which is the default SNP calling tool in Pipeliner. However, Pipeliner also implements a simple interface that allows using any other software for these tasks.

In the final step, Pipeliner calculates a number of statistics that summarize the performance of the defined bioinformatics' pipeline and provide plots to make interpretation easy. The statistics calculated include Recovery (% of SNPs correctly identified in relation to the original SNP number obtained with the coalescent simulations), Power (% of SNPs correctly identified in relation to all SNPs that pass the quality and depth filters set by the user), False Discovery Rate (FDR, % of SNP calls that are incorrect) as well as the frequency with which different errors occur.

Results and Discussion

As an example of the type of analysis possible in Pipeliner, we investigate the effect of individual SNP calling vs. multiple individual SNP calling. Results (Figure 1) show that, while the overall Power increases when joint SNP calling, singletons for the alternative allele actually become more elusive (lower Power), while other SNP sites become easier to detect. This situation is likely to lead to skewed allele frequency spectrum calculations, however such detailed bias has not, to our knowledge, been reported before.

Qualitatively similar results were obtained with higher coverage per individual (12x).

The choice of experimental design and bioinformatics' pipelines are central issues in

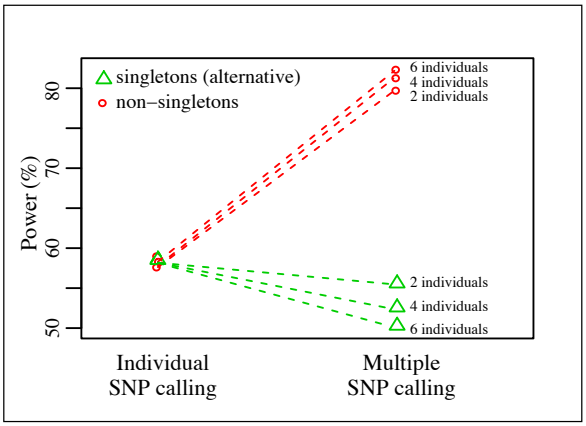

Figure 1. The effect of multiple-sample SNP calling.
Power in identifying heterozygous SNPs when doing individual or multiple SNP calling with samtools, under low coverage (average 6x per diploid individual) and few individuals (2, 4 and 6).

the analysis of NGS datasets. With Pipeliner, we provide the tools that empower researchers to carefully plan their study's sampling design, and compare the suitability of alternative software for their specific study systems. Pipeliner can be obtained from its website: https://github.com/brunonevado/pipeliner.

Acknowledgements

The authors would like to thank S. Ramos-Onsins, W. Sanseverino and R. Tonda for helpful discussions and comments. This work was supported by the Spanish Ministerio de Ciencia e Innovacion [AGL2010-14822 to M.P.E.]; and the Center for Research in Agricultural Genomics [consolider project CSD2007-00036].

References

Huang W, Li L, Myers JR, Marth GT (2011) ART: a next-generation sequencing read simulator. *Bioinformatics* **28** (4): 593-594. doi: 10.1093/bioinformatics/btr708

Hudson RR (2002) Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics* **18** (2): 337–338. doi: 10.1093/bioinformatics/18.2.337

Li H, Durbin R (2009) Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25** (14): 1754-1760. doi: 10.1093/bioinformatics/btp324

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J *et al.* (2009): The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25** (16): 2078–2079. doi: 10.1093/bioinformatics/btp352