Applications of a generic model of genomic variations functional analysis

Sarah Mapelli, Uberto Pozzoli✉

Scientific Institute I.R.C.C.S. "E. Medea", Bosisio Parini (LC)

Motivation and Objectives

Deep sequencing techniques, as well as the inherent equipment, are dramatically increasing their popularity in many scientific communities such as computational biology, "omics" and clinical research groups. The reasons are both scientific and budgetary: new experiments can be performed at a steadily decreasing cost. Nevertheless there are technical issues still to be addressed to make the results really useful in all the communities. We limit our interest to the "final" results of these experiments, usually a set of DNA/RNA variants or annotations relative to some reference assemblies. Except for those who are studying and developing algorithms and tools to produce them, these data are what people in different fields have to deal with. They are necessarily big and organized in a way that doesn't simplify their interpretation in terms of functional effect on phenotype. We speculated that a conceptual model of the connections between these data and the "genomic objects" usually studied (i.e. transcripts, miRNA, chromosomes, and so on) can be useful to make analyses and could greatly simplify the development of tools and programs. After defining such a model we implemented it, as well as a number of utilities. The result of this work is a C++ library (namely GeCo++: Genomic Computation C++ library) still actively developed but already used in our institute.

Methods

In the GeCo++ library, a genomic reference is defined as a portion of DNA identified by a name. A genomic element is then defined as an interval with a given strand along a genomic reference. Positions along a genomic reference are defined as zero based unsigned values. Element positions are defined relatively to the beginning of a given element and therefore are represented by signed zero based values. Intervals for both references and elements are considered as

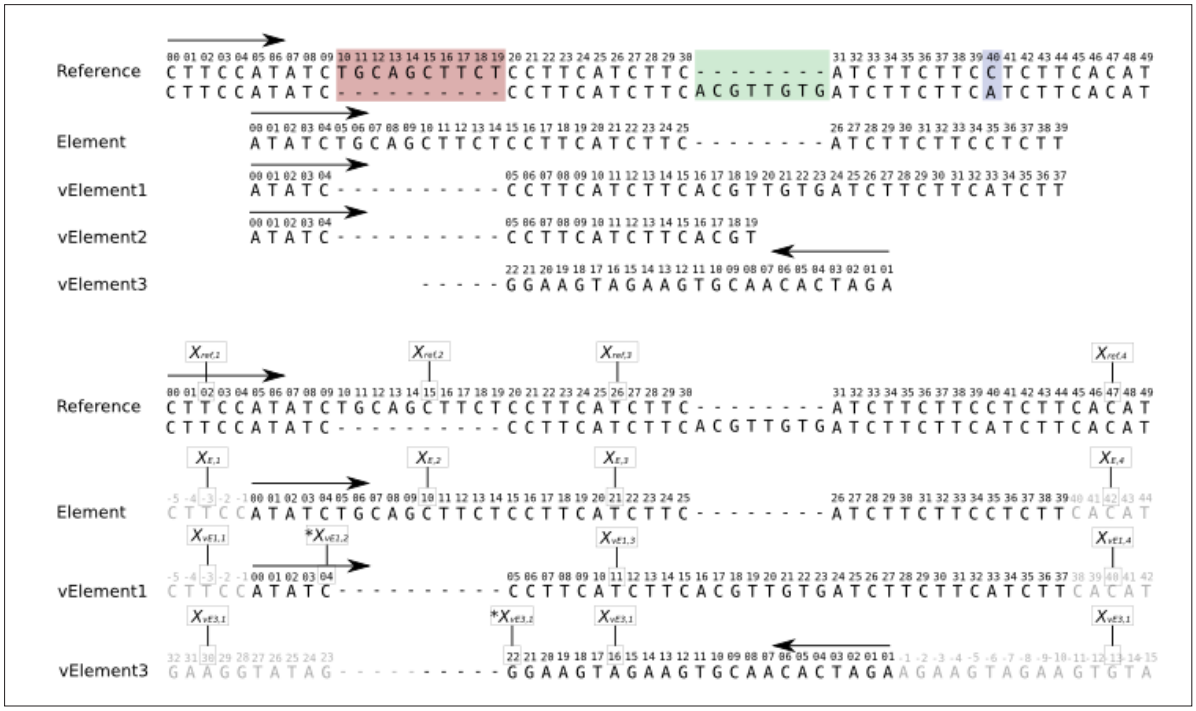

Figure 1: Reference, elements and variated elements definitions (upper panel). Positions and Interval mapping conventions are illustrated in the lower panel.

right open ones. A genomic element instance is defined by its reference boundaries (interval), its strand and possibly by a number of variations (insertions, deletions and substitutions) relative to the reference.

One of our goals was to allow forth and back conversion between element and reference positions. A set of simple and consistent rules has been defined and is applied throughout the library in order to allow interval/position conversion and mapping between reference and variated elements. Some examples of conversion from reference to element positions and intervals are reported in Figure 1.

"Sites" can be added to Element objects. Sites are positions along the element that have a particular biological meaning depending on the application one is developing. As well as sites, "Connections" can be added, representing meaningful directed links between two sites. Finally we define "Features" as numerical properties whose value varies along the element. Let's also define a "Feature Calculator" as an object that implements some algorithms to calculate/retrieve a specific feature along an element.

The objects defined above have been implemented in the GeCo++ library (Cereda et al., 2011). C++ was chosen because it's object oriented, faster than other languages (especially interpreted ones) and because a great number of high quality computational biology C++ libraries do exist and can be readily included in C++ programs.

Further implementations, with respect to the published core, include the definition of the class "Genotypes" as an object intended to represent genotype information deriving from both resequencing or genotyping experiments: it does so in terms of differences from a given reference. This object closely resembles to the kind of information one can find in a VCF (Variant Call Format) file. At this point we can formally define a pseudo function to assess the effect of one or more variations on a given feature along a genomic element as:

"mutated" element = method ("reference" element, genotypes , feature calculator)"

The resulting element allows to easily compare the feature values with the original ones even when insertions and deletions are present. Furthermore, this comparison is independent from the algorithm used to calculate the feature and therefore the subsequent analyses can be performed in the same way for a given element type independently from the algorithm used to calculate the feature. Also, the genotypes object provided can derive from any kind of experiment: in this way, for example, it is straightforward to apply our function to data coming from Sanger sequencing to confirm NGS results.

A series of more specialized classes and functions have also been added to the library to retrieve elements from different sources (i.e. UCSC, Ensembl, gff files), to calculate a variety of features (PWM scores, RNA secondary structure) and to perform statistical tests on genotype information. To this purpose a database structure has been defined to hold genotype information that can be accessed through the genotypes class. In this way we can read genotype information from the VCF file resulting from a whole genome multi-sample experiment, store it in the database and later retrieve the information relative to the region/element and samples of interest.

By using the library it is particularly fast and easy to produce applications that implement complex tasks by using the method abstraction. Since C++ is not the most popular language (especially for those who have a biological background) we also developed a simple and lightweight framework which can produce command line applications that can be called from the R statistical package (R Core Team, 2012) and as web services. In this latter case a simple javascript library implements an Ajax interface.

Results and Discussion

The library is extensively used in our lab, we therefore have a way to store NGS as well as Sanger experiment results in a database. We also can analyze the results in different ways: from genome wide population genetics studies to single gene analyses performed by biologists in the molecular biology lab.

A set of applications has been developed to insert variations in the database and to analyze them. In particular through an application called deLorean it is possible to apply a variety of population genetics statistics to resequencing data. This application has been used in several population genetics studies recently published by our group to analyze the 1000 genomes project data. From the functional analysis point of view, a still provisionally named "testPWM" ap-

plication can evaluate variation effects on PWM scores of any JASPAR (Bryne et al, 2008) PFAM TFBS (Transcription Factor Binding Sites) matrix for any resequenced region in the database. Another application for which a web interface does already exist, allows researchers in our Institute to annotate their NGS sequencing variants with respect to a list of transcripts or to the transcripts overlapping a given genomic region.

The applications developed so far are extensively tested (especially some of the utilities) by all groups in our institute, some part still need to be fully developed and an effort should be made in the near future to exploit the parallel computing opportunities offered by the modern hardware. The library, as well as the applications are available upon request.

Acknowledgements

This research was funded by the Italian ministry of Health. We wish to thank Matteo Cereda for his great ideas and contribution to the library development.

References

1. Cereda M, Sironi M, et al. (2011) GeCo++: a C++ library for genomic features computation and annotation in the presence of variants. Bioinformatics. 1;27(9):1313-5. doi:10.1093/bioinformatics/btr123
2. R Core Team, (2012) R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing Vienna, Austria, ISBN 3-900051-07-0, http://www.R-project.org
3. Bryne JC, Valen E, et al. (2008) JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update., Nucleic Acids Res. 36 (Database issue):D102-6. doi:10.1093/nar/gkm955