BIOINFORMATICS APPLICATIONS NOTE
Vol. 26 no. 14 2010, pages 1783–1785
doi:10.1093/bioinformatics/btq281

Sequence analysis

Manipulation of FASTQ data with Galaxy

Daniel Blankenberg[1,†,‡], Assaf Gordon[2,†], Gregory Von Kuster[1,‡], Nathan Coraor[1,‡], James Taylor[3,*,‡], Anton Nekrutenko[1,*,‡] and the Galaxy Team[‡]

[1]Huck Institute for the Life Sciences, Penn State University, University Park, PA 16803, [2]Cold Spring Harbor Laboratory, Watson School of Biological Sciences, Howard Hughes Medical Institute, Cold Spring Harbor, NY 11724 and [3]Departments of Biology and Mathematics and Computer Science, Emory University, Atlanta, GA 30322, USA

Associate Editor: John Quackenbush

ABSTRACT

Summary: Here, we describe a tool suite that functions on all of the commonly known FASTQ format variants and provides a pipeline for manipulating next generation sequencing data taken from a sequencing machine all the way through the quality filtering steps.

Availability and Implementation: This open-source toolset was implemented in Python and has been integrated into the online data analysis platform Galaxy (public web access: http://usegalaxy.org; download: http://getgalaxy.org). Two short movies that highlight the functionality of tools described in this manuscript as well as results from testing components of this tool suite against a set of previously published files are available at http://usegalaxy.org/u/dan/p/fastq

Contact: james.taylor@emory.edu; anton@bx.psu.edu

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

Received on April 1, 2010; revised on May 20, 2010; accepted on May 24, 2010

1 INTRODUCTION

The proliferation of next generation sequencing technologies has created numerous data management and analysis issues. The most troubling of these issues stems from the lack of standardized sequencer output and tools. The *de facto* standard, FASTQ, comes in a number of distinct variants (Cock *et al*., 2009), causing much of the difficulty experienced by biologists when working with next generation sequencing data.

The first steps following data acquisition involve preparing and quality checking the produced sequencing data. These steps typically follow the workflow: (i) parsing sequencer output; (ii) calculating and (iii) visualizing summary statistics on quality scores and nucleotide distributions; (iv) trimming reads if necessary; and (v) filtering reads by quality score and other various manipulations. Here, we describe a set of tools, able to handle all known FASTQ variants, that has been integrated into the online data analysis platform Galaxy (http://usegalaxy.org), allowing experimental biologists without any programming experience to easily manipulate sequencing data using a point and click interface. This open-source toolkit has no known data size limitations and was implemented in Python, allowing true cross platform availability. All of the following tools, unless mentioned otherwise, are found under the next generation sequencing (NGS): quality check (QC) and manipulation section within Galaxy. Furthermore, by integrating this toolset into Galaxy, researchers have access to a plethora of other genome analysis software as well as a fully customizable workflow (pipeline) system. Blankenberg *et al*. (2007, 2010), Taylor *et al*. (2007) and http://galaxycast.org are recommended for users to familiarize themselves with genome analysis using Galaxy.

2 A SUITE OF TOOLS FOR PREPARING NEXT GENERATION SEQUENCING READS FOR MAPPING AND ANALYSIS

2.1 FASTQ from FASTA and quality score files

Some sequencing technologies will produce separate files containing sequences and quality scores. These two separate files can be merged together to create a single FASTQ file. For this purpose, the combine FASTA and QUAL into FASTQ tool was developed. Specifying a quality score file is optional and, when not specified, quality score values will be filled with the maximal allowed quality value.

2.2 FASTQ Groomer

The FASTQ Groomer tool is used to verify and convert between the known FASTQ variants. The data created by this tool is guaranteed to conform to the target variant specified by the user, including the enforcement of quality score minimums and maximums. After grooming, the user is presented with some information about the input such as ASCII character and decimal value ranges and a list of FASTQ variants for which the input data is actually valid. Although the output created by this tool is now valid, if the user has selected the wrong presumed input variant, it is possible for the resultant score values not to reflect the values intended by the sequencing technology. Users should utilize the provided summary information as a sanity check before continuing with their analysis; for example, if a user provides a Sanger encoded variant (with ASCII values <59), but specifies the input variant as Solexa, this summary information would state that the input was valid only for Sanger (a direct contradiction of the user's selection).

[*]To whom correspondence should be addressed.

[†]The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

[‡]http://galaxyproject.org

2.3 Quality statistics

As quality scores can vary along the length of sequencing reads, determining how to trim and filter read data involves calculating summary statistics on a per column (base position) basis. The FASTQ Summary Statistics by column tool accomplishes this task. The output of this tool contains read counts, minimums, maximums, sums, means, quartiles with ranges, outliers and nucleotide counts for each base position in a FASTQ file. This statistical summary can be graphed by using the Boxplot tool, found under the Graph/Display Data tool section.

2.4 Read Trimmer

To prevent otherwise high-quality reads from being rejected during quality filtering or from influencing mapping or assembly processes, it can be beneficial to trim bases from poor-quality ends of reads. The FASTQ Trimmer by column tool allows trimming either end of a set of reads by using absolute offsets or by specifying percentage of read length based offsets. Offsets begin at 0 for each end and increase towards the opposing end of the read. For example, to trim the outer 3 bases from each end of a 36 length sequencing read, a user can specify absolute $5'$ and $3'$ offsets of 3 or percentage-based offsets of 8.33 ($0.0833 \times 36 = 2.9988$, rounded to the nearest integer = 3).

2.5 Quality filter

The Filter FASTQ reads by quality score and length tool allows filtering by minimum and maximum read lengths and by minimum and maximum quality score values over the entire read while allowing a configurable number of deviant bases. Complex filters can also be constructed that allow the user to set offsets, just like with the trimmer tool, to use as bounds for performing a selected aggregation action that is compared to a user specified value. Any number of complex filters can be designed and applied to a set of sequencing reads. For example, to only include reads which have no quality score values less than 28 in the first half of a read, a user can use percentage-based offsets of 0 and 50, select the minimum score aggregation and the greater-than-or-equal-to operator ($\geq$) and set a quality score threshold of 28.

2.6 FASTQ Manipulation

Highly configurable complex manipulations can be performed on selected FASTQ reads by using the Manipulate FASTQ reads on various attributes tool. This tool allows the user to define a set of matching criteria to be used to select the reads in a FASTQ file on which to perform a set of manipulations; any number of match directives can be defined and a read must match each directive to be considered for manipulation. Matching is currently limited to user-specified pattern matching (regular expressions) on sequence identifier/name, sequence content and quality score strings, with defaults set to match all (.*); however, additional matching and manipulation options can be easily implemented as needed. When a read does not match, it will be transferred to the output in an unmodified fashion. Reads that pass all matching criteria are subjected to any number of user-specified manipulations. Manipulations are available that act upon sequence identifier/name, sequence content or quality score strings. Beyond allowing the user to remove matching reads or to perform string translations on any of these attributes, additional manipulations are available for sequence content, including: reverse complementing, reversing (without complementing), complementing (without reversing), trimming, *in silico* transcription of DNA to RNA and vice-versa, as well as changing the adapter base within color space sequences. Additionally, separate tools exist that can convert FASTQ files to-and-from a tabular format; this allows FASTQ data to be modified using any of the powerful text manipulation tools, which are prepackaged with Galaxy.

2.7 Paired-end read splitting and joining

FASTQ formatted paired-end sequencing data can come in two common forms, one that utilizes a separate file for each paired-end component or another where a single FASTQ file is used and the two paired-end reads ends have been concatenated together to form a single entry. Two tools exist to facilitate the use of these data: FASTQ Joiner on paired-end reads and FASTQ Splitter on joined paired-end reads. The Joiner tool takes two separate FASTQ files that contain paired-end reads and creates a single file. The Splitter tool does the opposite of the Joiner tool and takes a single FASTQ file and splits each read in half, creating two separate FASTQ files. When splitting, an identifier suffix is added to each paired end; when joining, these differences in identifiers are taken into account.

3 CONCLUSIONS

Although the differences between the FASTQ variants will likely continue to cause difficulty for researchers, it is our hope that the adoption of this toolset will alleviate many of these problems by providing facilities that allow verification and interconversion of these variants and which are available alongside a comprehensive collection of tools. Although this toolset aims to be both simple to use and functionally powerful, ultimately the user is responsible for understanding the analytical requirements of their data; e.g. the difference between fixed read length and variable read length platforms. To help biologists overcome the nominal learning curve associated with this toolset, onscreen help is displayed within each tool interface and a series of screencasts, which demonstrate a typical analysis with this toolset as it appears on a user's screen, is available at http://galaxycast.org.

A description of the results of running this toolset against test files provided in Cock *et al*. (2009) is available in the Supplementary Material. To prevent potential problems from occurring as future enhancements are made to the toolset, these files have been incorporated as functional test cases that are automatically executed whenever the source code is updated. As always, users are encouraged to send comments, suggestions, feature requests and bug reports to galaxy-bugs@bx.psu.edu.

Funding: Beckman Foundation Young Investigator Award (to A.N.); NSF grant DBI 0850103 and NIH grant HG004909 (to A.N. and J.T.); Penn State University and the Huck Institutes for the Life Sciences (to A.N.); Emory University (to J.T.); grant with the Pennsylvania Department of Health using Tobacco Settlement Funds (in part). The Department specifically disclaims responsibility for any analyses, interpretations, or conclusions.

Conflict of Interest: none declared.

REFERENCES

Blankenberg,D. *et al.* (2007) A framework for collaborative analysis of ENCODE data: making large-scale analyses biologist-friendly. *Genome Res.,* **17**, 960–964.

Blankenberg,D. *et al*. (2010) Galaxy: a web-based genome analysis tool for experimentalists. *Curr. Protoc. Mol. Biol*., **Chapter 19**, Unit 19.10.1-21.

Cock,P.J. *et al*. (2009) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Res.*, **38**, 1767–1771.

Taylor,J. *et al*. (2007) Using galaxy to perform large-scale interactive data analyses. *Curr. Protoc. Bioinformatics*, **Chapter 10**, Unit 10.5.