BIOINFORMATICS

Vol. 00 no. 00 2011
Pages 1–3

Analyzing microbial ChIP-Seq data with Pique

R.Y. Neches [1,3]*; E.G. Wilbanks [1,3] and M.T. Facciotti [2,3]

[1]Microbiology Graduate Group, University of California, Davis.
[2]Department of Biomedical Engineering, University of California, Davis.
[3]Genome Center, University of California, Davis.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation:

While numerous effective peak finders have been developed for eukaryotic systems, we have found that the approaches used can be suprisingly error prone when run on high-coverage bacterial and archaeal ChIP-Seq datasets.

Results:

We have developed Pique, a conceptually simple, easy to use ChIP-Seq peak finding application for bacterial and archaeal ChIP-Seq data. The software is cross-platform and Open Source, and based on Open Source dependencies. Output is provided in standardized bioinformatic formats, and easily imported by the Gaggle Genome Browser for manual curation and data exploration, or into statistical computing and graphics software such as R for further analysis.

Availability:

The software is available under the BSD-3 license at http://github.com/ryneches/pique.

A tutorial and test data are included with the documentation.

Contact: ryneches@ucdavis.edu

1 INTRODUCTION

Next generation sequencing coupled with chromatin immunoprecipitation (ChIP-Seq) is revolutionizing our ability to genomically map protein-DNA associations for a wide variety of proteins. The growing popularity of ChIP-Seq has spurred the development of over thirty peak picking algorithms (for a nearly completely list see [10]). The relative performance of representative peak detection algorithms on eukaryotic data and methods to assess performance have been recently reviewed by several authors [6, 3, 9, 2, 7].

While ChIP-Seq has been predominantly used to interrogate protein-DNA interactions in eukaryotic systems, there are clear advantages to adopting this technology for studying microbes. Microbial genomes are typically smaller than eukaryotic genomes (the genome of *E. coli* is about 2000 times smaller than that of humans), and have simpler replicon and chromatin structure.

Only one ChIP-Seq analysis tool, CSDeconv [5], has been explicitly developed for microbial data. This MATLAB package successfully finds peaks in microbial ChIP-Seq data, but its application is limited by its dependency on costly proprietary software, slow performance, lack of support for manual curation. Herein we describe Pique, a conceptually simple, Python-based peak finding package that enables easy and rapid peak finding in bacterial and archaeal ChIP-Seq datasets.

*to whom correspondence should be addressed

2 APPROACH

ChIP-Seq in bacteria and archaea yields coverage several orders of magnitude larger than in eukaryotic systems, resulting in continuous coverage rather than the sparse coverage typically present in eukaryotic ChIP-Seq data. This feature of microbial ChIP-Seq experiments permits simpler, faster algorithms to be used. We have based our algorithm on classic noise reduction techniques from signal processing.

Pique is designed for use in systems that have genomic complexities such as IS elements, gene dosage polymorphisms and accessory genomes that cause coverage variations unrelated to ChIP, or in cases where the organism under study is not identical to the reference genome. The resulting enrichment "pedestals" and "holes" can be problematic for detecting peaks and calculating enrichment levels. If the user provides a map of these features, the software will automatically perform a segmented analysis.

The wide variety of microbial systems, target proteins, protocols, and experimental conditions, calls for tailored statistical approaches to ChIP-Seq. Rather than attempting to anticipate each of these (and their combinations) by presenting the user with a very large number of statistical and heuristic parameters, we have chosen to focus on the aspects of the analysis that are common to all ChIP-Seq experiments; finding putative peaks, estimating binding coordinates and binding affinities. The determination of statistical significance is typically straightforward for any particular experiment, but is quite difficult to robustly generalize.

Pique allows users to create high-quality peak lists in two ways. First, we provide quantities for each peak that can be used to ascertain which peaks are statistically significant (usually, this involves little more than sorting the table and choosing a cutoff). Second, we provide integrated support for curation using the Gaggle Genome Browser. This permits interactive curation of the peak list and analysis of the ChIP-Seq data in the context of other Gaggle-enabled resources. Interactive curation of a microbial ChIP-Seq data set can typically be completed in a few minutes.

3 METHODS

Pique requires BAM files as input[4]. Therefore, prior to using Pique, Illumina (Solexa) reads should first be quality filtered, quality trimmed, and aligned to a reference genome (ideally, all contigs of the reference genome should be used as the mapping target).

By default, Pique treats each contig as a single analysis region, but the user may optionally designate regions within a contig for separate analysis by supplying a feature map. This may be useful when coverage levels are systematically altered over large regions. Pique supports three features types; analysis regions, masking regions, and normalization regions. Masking regions are simply masked out of their respective analysis regions, and are useful for removing coverage variation due to repetitive DNA. Normalization regions selected within an analysis region are used to compensate for total coverage discrepancies between the background and ChIP alignments.

The user launches the primary analysis stage by providing alignment an file for the ChIP data, an alignment file for the control data, and a coverage feature map. The primary analysis proceeds thusly :

- The alignment files are digested into coverage tracks, and the analysis regions are initialized in memory. Provided masking regions are applied.
- The coverage noise threshold is measured by comparing the relative total coverage within the normalization regions. (The user should choose normalization regions that contain neither peaks nor coverage level aberrations.)
- High-k noise in coverage is removed using a Wiener-Kolmogorov filter. The filter delay α is chosen to approximate to the expected footprint size of the targeted protein. The choice of filter implies the existence of two inputs; a "true" signal, and a noise source. Both are assumed to be stationary stochastic processes combined additively.
- A Blackman window of a diameter equal to the read length is convolved with the filtered coverage track to remove features smaller than one read. This reduces the effect of fragmentation position bias, and may be especially useful when transposon-based library construction is used.
- The noise threshold in the ChIP coverage track is measured by comparing the coverage distribution in the ChIP track to the control track within user-annotated non-peak regions. Features that exceed the noise threshold are identified.
- The coverage enrichment is distributed differently between the forward and reverse strands because the possible read orientations are constrained by the fragment size. Pique exploits this by requiring that the stop coordinate of the forward strand enrichment envelope must fall between the coordinates of the reverse strand enrichment envelope, and that the start coordinate of the reverse strand enrichment envelope fall between the coordinates of the forward strand enrichment envelope. (We call this the overlap criterion.)

For each putative peak, Pique calculates the enrichment ratio of the ChIP alignment to the control alignment, the binding coordinate, and the enrichment normalization factor for that analysis region.

4 DISCUSSION

Pique does not attempt to filter peaks that are statistically insignificant. We have found that this part of the analysis is rather specific to the data and to the experiment. Pique is designed to achieve a low false-negative rate. Thus, some kind of post-filtering is necessary. Pique provides the user with output that can be used to support a variety of such statistical tests.

Some recommended filtering might include eliminating peaks that are significantly narrower than the size range of the sequencing library, peaks with a normalized enrichment ratio below unity, or peaks that have predicted binding sites that are very skewed from the center of the enriched region. Depending on how many peaks are recovered, the user may wish to try one or all of these, perhaps with clustering. However, if a "perfect" peak list is required, we do not recommend relying on heuristic refinements alone. To facilitate manual curation, Pique outputs a track file of the coverage, a quantitative positional data of the estimated binding sites, and a bookmark file annotating the peaks. These files are simple to process by a variety of tools, and can be loaded directly into the Gaggle Genome Browser.[1]

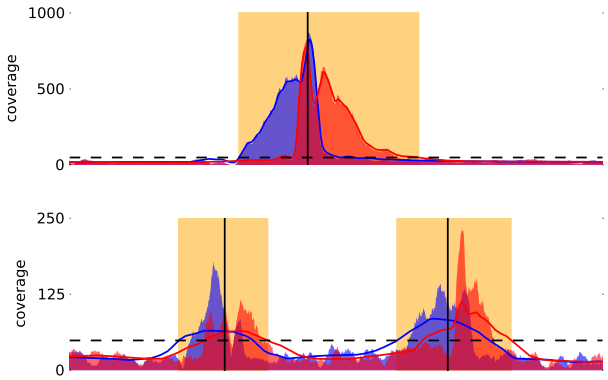

Fig. 1. Peaks found in the included sample dataset, derived from ChIP-seq of tfbD in *Halobacterium salinarum* sp. NRC1. Blue and red shading indicate coverage of reads aligned from the ChIP-derived data to the forward and reverse strands, respectively. Blue and red lines represent the filtered coverage levels for the forward and reverse strands, respectively. The dashed line is the detected noise threshold for the region. Orange indicates a detected peak.

5 CONCLUSION

We conclude that Pique provides a rapid, open source platform for the sensitive detection of transcription factor binding sites in bacterial and archaeal ChIP-seq experiments. We leverage standard signal processing algorithms to rapidly identify binding sites. Downstream analysis is supported via integration with statistical and graphics software such as R, and curation via integration with the user-friendly Gaggle Genome Browser and the suite of Gaggle tools.

We note that Pique should also work well with eukaryotic datasets provided they are gathered with greater coverage than has been previously reported.

ACKNOWLEDGEMENT

Funding: This project was funded by UC Davis startup funds to M.T.F., NSF Graduate Research Fellowship awarded to E.G.W. and DARPA award number HR0011-05-1-0057 to R.Y.N.

[1] See supplementary figure.

REFERENCES

[1]J. Christopher Bare, Tie Koide, David Reiss, Dan Tenenbaum, and Nitin Baliga. Integration and visualization of systems biology data in context of the genome. *BMC Bioinformatics*, 11(1):382+, July 2010.

[2]Xin Feng, Robert Grossman, and Lincoln Stein. PeakRanger: A cloud-enabled peak caller for ChIP-seq data. *BMC Bioinformatics*, 12(1):139+, 2011.

[3]Teemu Laajala, Sunil Raghav, Soile Tuomela, Riitta Lahesmaa, Tero Aittokallio, and Laura Elo. A practical comparison of methods for detecting transcription factor binding sites in ChIP-seq experiments. *BMC Genomics*, 10(1):618+, December 2009.

[4]Heng Li, Bob Handsaker, Alec Wysoker, Tim Fennell, Jue Ruan, Nils Homer, Gabor Marth, Goncalo Abecasis, Richard Durbin, and 1000 Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, 25(16):2078–2079, August 2009.

[5]Desmond Lun, Ashley Sherrid, Brian Weiner, David Sherman, and James Galagan. A blind deconvolution approach to high-resolution mapping of transcription factor binding sites from ChIP-seq data. *Genome Biology*, 10(12):R142+, December 2009.

[6]Shirley Pepke, Barbara Wold, and Ali Mortazavi. Computation for ChIP-seq and RNA-seq studies. *Nature Methods*, 6(11s):S22–S32, October 2009.

[7]Morten B. Rye, Pål Sætrom, and Finn Drabløs. A manually curated ChIP-seq benchmark demonstrates room for improvement in current peak-finder programs. *Nucleic Acids Research*, 39(4):e25, March 2011.

[8]Paul Shannon, David Reiss, Richard Bonneau, and Nitin Baliga. The Gaggle: An open-source software system for integrating bioinformatics software and data sources. *BMC Bioinformatics*, 7(1):176+, 2006.

[9]Adam M. Szalkowski and Christoph D. Schmid. Rapid innovation in ChIP-seq peak-calling algorithms is outdistancing benchmarking efforts. *Briefings in Bioinformatics*, November 2010.

[10]Elizabeth G. Wilbanks and Marc T. Facciotti. Evaluation of Algorithm Performance in ChIP-Seq Peak Detection. *PLoS ONE*, 5(7):e11471+, July 2010.