Bioinformatics, 31(18), 2015, 3046–3047
doi: 10.1093/bioinformatics/btv322
Advance Access Publication Date: 20 May 2015
Applications Note

OXFORD

Genome analysis

agplus: a rapid and flexible tool for aggregation plots

Kazumitsu Maehara and Yasuyuki Ohkawa*

Department of Advanced Medical Initiatives, JST-CREST, Faculty of Medicine, Kyushu University, Fukuoka 812-8582, Japan

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on February 24, 2015; revised on April 24, 2015; accepted on May 17, 2015

Abstract

Summary: Aggregation plots are frequently used to evaluate signal distributions at user-interested points in ChIP-Seq data analysis. agplus, a new and simple command-line tool, enables rapid and flexible generation of text tables tailored for aggregation plots from which users can easily design multiple groups based on user-definitions such as regulatory regions or transcription initiation sites.

Availability and Implementation: This software is implemented in Ruby, supported on Linux and Mac OSX, and freely available at http://github.com/kazumits/agplus

Contact: yohkawa@epigenetics.med.kyushu-u.ac.jp

1 Introduction

Aggregation plots are frequently used to convert genome-wide chromatin immunoprecipitation-sequencing (ChIP-Seq) data into a simple curve of averaged ChIP-Seq signal intensities at user-interested points (anchor points), such as regulatory sequences (Maehara *et al.*, 2013). Aggregation plots allow differences in the signal distribution among different ChIP-Seq data to be assessed intuitively. Many ChIP-Seq data analysis toolkits include software for the creation of aggregation plots, including HOMER, bwtools, CGAT, ngsplot and HTSeq (Anders *et al.*, 2015; Heinz *et al.*, 2010; Pohl and Beato, 2014; Shen *et al.*, 2014; Sims *et al.*, 2014). These software are designed to analyze transcription start sites (TSSs), transcription factor binding sites (TFBSs), exon-intron structures, etc. For example, ngsplot takes a configuration file of gene sets with corresponding raw data to produce a visual of different ChIP-Seq signal distributions at gene loci. More recently, ChIP-Seq applications have begun to offer diverse purposes for aggregation plots. However, user definitions of the anchor points required for aggregation plots are not trivial, e.g. comparisons of the signal distribution of transcription start sites around different gene sets (gene ontology database or a set defined by their gene expression profile data) or regulatory regions (promoters, enhancers or other non-coding region). Therefore, a simpler tool for loci definitions and grouping tasks is preferred. Here, we introduce agplus, a flexible tool designed for the creation of aggregation plots using various points of user interest.

2 Feature

agplus was implemented in Ruby to calculate the average signal intensity (aggregated signal) around specific anchor points from ChIPseq data. agplus calculates the average signal intensity of the 'target' signal at each 'reference' region group as defined by an 'assignment' table, therefore agplus script requires reference, target and assignment files. The reference file defines the anchor points of user interest, e.g. gene loci or regulatory regions. The file should be in BED6 format (6 columns BED), since the names and strand information are needed for the calculation. The target file should be in wiggle or BED format with a scores field. The assignment file is a simple tab-delimited, two-column text file. The first column holds the reference name defined in the reference file and the second column holds an arbitral group name of the references. The assignment file defines a group of anchor points, e.g. gene loci corresponding to gene ontology or gene expression levels against a RefSeq gene list, which is used as a reference file. The assignment file is therefore what gives the benefits of agplus. The output file is a tab-delimited text table of averaged signal intensities in each group per 1 base pair. The output file can be easily handled in subsequent analysis using gnuplot, R, Matlab MS Excel, etc. The fast processing time of agplus comes from bedtools (Quinlan and Hall, 2010), which agplus internally uses to calculate overlaps between the target and reference files.

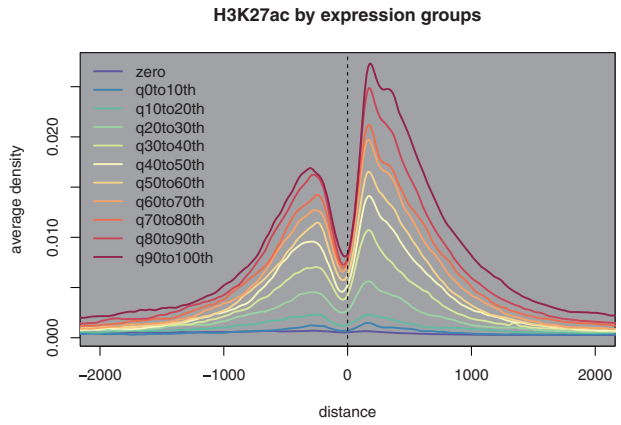

Fig. 1. An example the HeLa-S3 H3K27ac ChIP-Seq signal distribution of gene expression groups around TSS, generated by agplus. Different colors indicate the expression levels of gene groups within the defined percentiles

3 Usage and applications

The installation places scripts anywhere. Ruby (version > 2.0) and BEDTools are required to be in the `$PATH` environment.

The target file is a single wiggle or BED format with scores field and converted through bam2bwshifted to support the bam file. The assignment file should be a single file, and *assignExprGroup* supports the gene expression profile of mRNA-Seq data generated by cufflinks (Trapnell *et al.*, 2012). The script generates an assignment file of gene groups divided by 10 percentile expression levels (default interval length), as shown in Figure 1. The drawing of a smoothed average signal intensity using the output table of agplus is supported by *agpdraw-line*. The supporting utilities of *bam2bwshifted* requires SAMTools (Li *et al.*, 2009) to count read numbers and UCSC's *wigToBigWig* (Kent *et al.*, 2010) for format conversion. *agpdraw-line* requires R to smooth and draw curves.

The basic procedure to create an aggregation plot from mapped reads (bam) (Li *et al.*, 2009) consists of four steps, as shown below:

1. `bam2bwshifted -o sample-shifted.bw -s (half-size of fragment-length) -g (chromosome size definition file) target.bam`
2. `bigWigToWig target-shifted.bw target-shifted.wig`
3. `agplus -b reference.bed -a assignment.txt -d (start, center or end) -o aggregated.txt -r (from),(to) target-shifted.wig`
4. `agpdraw-line -o output.pdf [-c control_aggregated.txt] aggregated.txt`

Step 1 generates an optimal target file for the aggregation. The output file, sample-shifted.bw (bigWig; compressed wiggle format), holds reads per million-normalized (Mortazavi *et al.*, 2008) counts of the mid-point of the mapped DNA fragment on the genomic coordinate. This file conversion step reduces the intermediate file size and processing time of subsequent calculations by agplus. Step 2 decompresses the bigWig file such that the target file can be used for agplus. Step 3 aggregates the read counts in the target file by the groups defined in the assignment file at the anchor-points defined in the reference file. The '-d' option sets the position of the anchor points relative to the reference, and can be either of start (left-most), end (right-most) or center, i.e. '-d start' uses TSSs as center when the RefSeq gene is the reference. Step 4 generates a PDF file of the aggregation plot, in which curves range from bp `from` to bp `to`. The total computation time of these four steps to obtain the result of Figure 1 takes ~4 min under our computational environment (Intel(R) Xeon(R) CPU E5-2687W on x86_64 GNU/Linux). As an option, control_aggregated.txt can be supplied with '-c' to correct the bias of input DNA. The generated curve reports the average density of the reads aggregation, i.e. summation makes total RPM within 1 kb from the anchor points. The same procedure is also applicable to estimate nucleosome densities from MNase-Seq data.

4 Conclusion

We have developed agplus, a tool that enables rapid and flexible creation of aggregation plots. This tool is expected to reduce effort and time for aggregation-style analysis by ChIP-Seq users.

Acknowledgements

We thank Ohkawa lab members for technical feedback, P. Karagiannis for reading the manuscript, and the Research Institute for Information Technology, Kyushu University (tatara) and the National Institute of Genetics (NIG) for providing high-performance computing resources.

Funding

This work was supported by the Core Research for Evolutional Science and Technology (CREST), Japan Society for the Promotion of Science (JSPS) KAKENHI [grant numbers 26290064, 25116010, 25132709, 25118518]. Funding for open access charge: JSPS KAKENHI.

Conflict of Interest: none declared.

References

Anders,S. *et al.* (2015) HTSeq-a Python framework to work with high-throughput sequencing data. *Bioinformatics*, **31**, 166–169.

Heinz,S. *et al.* (2010) Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities. *Mol. Cell*, **38**, 576–589.

Kent,W.J. *et al.* (2010) BigWig and BigBed: enabling browsing of large distributed datasets. *Bioinformatics*, **26**, 2204–2207.

Li,H. *et al.* (2009) The sequence alignment/map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.

Maehara,K. *et al.* (2013) A co-localization model of paired ChIP-seq data using a large ENCODE data set enables comparison of multiple samples. *Nucleic Acids Res.*, **41**, 54–62.

Mortazavi,A. *et al.* (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods*, **5**, 621–628.

Pohl,A. and Beato,M. (2014) bwtool: a tool for bigWig files. *Bioinformatics*, **30**, 1618–1619.

Quinlan,A.R. and Hall,I.M. (2010) BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, **26**, 841–842.

Shen,L. *et al.* (2014) ngs.plot: quick mining and visualization of next-generation sequencing data by integrating genomic databases. *BMC Genomics*, **15**:284.

Sims,D. *et al.* (2014) CGAT: computational genomics analysis toolkit. *Bioinformatics*, **30**, 1290–1291.

Trapnell,C. *et al.* (2012) Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat. Protoc.*, **7**, 562–578.