SOFTWARE

SUPERmerge: ChIP-seq coverage island analysis algorithm for broad histone marks

Bohdan B. Khomtchouk^{1*}, Derek Van Booven² and Claes Wahlestedt¹

*Correspondence:

b.khomtchouk@med.miami.edu

¹Center for Therapeutic

Innovation and Department of

Psychiatry and Behavioral

Sciences, University of Miami

Miller School of Medicine, 1120

NW 14th St, Miami, FL, USA

33136

Full list of author information is
available at the end of the article

Abstract

SUPERmerge is a ChIP-seq read pileup analysis and annotation algorithm for investigating alignment (BAM) files of diffuse histone modification ChIP-seq datasets with broad chromatin domains at a single base pair resolution level. SUPERmerge allows flexible regulation of a variety of read pileup parameters, thereby revealing how read islands aggregate into areas of coverage across the genome and what annotation features they map to within individual biological replicates. SUPERmerge is especially useful for investigating low sample size ChIP-seq experiments in which epigenetic histone modifications (e.g., H3K9me1, H3K27me3) result in inherently broad peaks with a diffuse range of signal enrichment spanning multiple consecutive genomic loci and annotated features.

Keywords: ChIP-seq; epigenetics; bioinformatics; computational biology

Introduction

Standard ChIP-seq wet and dry lab workflows are very well characterized, and their respective tools and protocols frequently and consistently employed [2, 16, 22]. One hallmark shared in common by all ChIP-seq computational workflows is the process of peak calling, whereby differential enrichment of reads is statistically detected across the genome in case-control samples.

It is known that epigenetic histone modifications that have inherently broad peaks with a diffuse range of signal enrichment (e.g., H3K9me1, H3K27me3) differ significantly from narrow peaks that exhibit a compact and localized enrichment pattern (e.g., H3K4me3, H3K9ac) [15]. Likewise, transcription factors that exhibit punctate protein-DNA enrichment patterns resulting in narrow and well-defined peaks differ significantly from the broad peaks frequently detected in diffuse chromatin mark data. Due to these differences in peak topology, various specialized peak calling algorithms have been developed to handle both transcription factor ChIP-seq and histone modification ChIP-seq data analysis [6, 11–14, 20, 24, 28, 29, 31, 32].

Despite these computational advances in identifying putatively enriched genomic regions, peak calling software generally relies on pooling the respective BAM files of each biological replicate within each group (in cases and controls) prior to performing the peak calling step. This is done in order to merge multiple sorted alignment (BAM) files within each group, thereby reliably boosting read count and amplifying the downstream differential enrichment signal obtained during the peak calling process. While pooling certainly achieves the aforementioned goal, it may inadvertently lead to erroneous results due to one or two over-representative samples whose read counts swamp the overall signal. This is especially relevant to experiments with low

sample sizes and yields (e.g., brain tissues [3,19]), where a single biological replicate might drive the overall signal of the (case or control) group. As such, it is strongly recommended to pre-process data with quality control software to ensure that the sequencing phase was successful [27]. A popular tool designed for this purpose is FastQC [1], which can check for biases and other sequencing anomalies, including duplication rate and per-base GC content and sequence quality score. A complementary tool, MGA [10], can check for contamination, un-alignable sequence, and presence of sequencing adapter dimers [27]. In addition, the irreproducible discovery rate (IDR) can be used to measure general consistency between replicates in high-throughput experiments [17] and is made available as an R package [18]. Another popular alternative is the R/Bioconductor package CHIPQC [7], which computes a number of quality metrics relating to the alignments, including duplication rates, mapping quality filtering rates, distribution of pileups (and the associated SSD measure), and an estimated fragment size based on maximizing a relative cross-coverage score [26].

Despite the abundance of quality control and peak calling software available to-date, there has not yet been a tool for quantitatively examining read pileup and annotation information at the level of a single BAM file. This would be useful for identifying read pileup outliers across select individual (un-pooled) samples, as well as investigating the latent impact of broad coverage islands whose summit position, length, and other dimensions are unlikely to be registered as statistically significant (i.e., false-negatives) by existing peak calling software. Since peak caller statistical significance often precludes many genomic loci (and their respective genes) from further consideration, SUPERmerge offers the analyst complete freedom in specifying coverage depth parameters and broadness dimensionality (see [Methods](#) section) – a paradigm-shifting idea that we propose for annotating genomic features and planning subsequent follow-up analyses. As such, we recommend using SUPERmerge for exploratory data analysis purposes (not peak calling), e.g., when investigating annotations of extremely broad histone modification coverage islands that span multiple consecutive genomic loci and annotated features.

Methods

SUPERmerge is a cross-platform command-line program written in the C programming language that takes as input a single BAM file and produces as output a variety of tabulated flat file results (see software manual), as well as graphical data visualization output (see [Results](#) section) generated by the R programming language [23]. SUPERmerge source code and compilation instructions are made publicly available at: <https://github.com/Bohdan-Khomtchouk/SUPERmerge>. A pre-compiled binary executable and software manual are available for immediate download at: <https://sourceforge.net/projects/supermerge/files/SUPERmerge.zip/download>.

SUPERmerge is designed to be used as part of a genomic workflow (Figure 1) prior to, or in conjunction with, the process of peak calling (see [Results](#) and [Discussion](#) sections for rationale). The program is composed of three primary parameters that are specified as user flags:

- *-d*: depth

- $-i$: interval grouping factor
- $-g$: GTF annotation

Depth represents read pileup at a given genomic location (e.g., “ $-d$ 20” returns all single base pair positions in a BAM file harboring at least 20 or more reads). SUPERmerge leverages the mpileup function in the SAMtools software suite [?] to calculate depth. The interval grouping factor ($-i$) represents a genomic distance (in bp) between any two consecutive positions of sufficient depth ($-d$), for which any two neighboring base pair positions are considered to be part of the same genomic interval if they are separated by no more than this distance ($-i$). In practice, this translates to merging any base pair positions that are of sufficient depth $-d$ into one contiguous interval of coverage, as long as the gap between these genomic positions is no greater than $-i$. We refer to these contiguous regions of coverage as “coverage islands”, since there exists no base pair position within a $-i$ radius of the island (in either the 3-prime or 5-prime end) that is of the sufficient depth $-d$. However, within any given coverage island, there will always be a certain percentage of base pairs that do not meet the sufficient depth, simply by virtue of the fact that any two consecutive regions of sufficient depth are merged into a continuous interval whenever there is a gap of less than $-i$ bp between them. The frequency of occurrence of coverage islands whose base pairs meet a certain percentage of sufficient depth per island is counted and graphed in Figure 2A (see Results section for details).

As such, these two thresholds ($-d$ and $-i$) constitute the cutoff parameters by which a user can explore how reads assemble into coverage islands at various levels of exploratory data analysis. Coverage islands are defined as contiguous genomic regions with a certain percentage of read coverage above a set threshold depth and a distance of at least $-i$ bp from their nearest neighboring island. Once these coverage islands are calculated, the $-g$ flag specifies a gene transfer format (GTF) file that annotates them with their respective genomic features (see Results section).

An example of a full command issued from the command-line is:

```
./supermerge -d 20 -i 500 -g gencodeV19annotation.gtf file.bam
```

This command would be repeated by the user across all BAM files in the sample pool, for any desired combinations of $-d$ and $-i$.

Results

SUPERmerge allows users to analyze read pileup distribution across individual alignment (BAM) files of biological replicates, returning a variety of graphical and tabulated output results specific to the parameter space described in Methods. There are four graphs that are produced for each set of replicate BAM files:

- Annotation graph: a bar chart of all coverage islands annotated to GTF features (protein.coding, lincRNA, miR, antisense, and pseudogene)
- Island diameter graph: a histogram of coverage island length vs. frequency of occurrence
- Library size graph: a histogram of coverage island area (total sum of reads per island) vs. frequency of occurrence
- Percentage graph: a histogram showcasing the percentage of bases within an individual coverage island that meet the sufficient depth ($-d$) after merging ($-i$) vs. frequency of occurrence

Figure 2 shows a representative histone modification ChIP-seq sample run produced for the H3K4me3 mark in the human GM12878 cell type downloaded from the ENCODE ChIP-seq Experiment Matrix [5, 9]. This data is composed of two biological replicates, both of which are run under the same set of parameters for their respective flags ($-d$, $-i$, and $-g$):

```
./supermerge -d 20 -i 500 -g gencodeV19annotation.gtf
                wgEncodeBroadHistoneGm12878H3k4me3StdAlnRep1.bam
./supermerge -d 20 -i 500 -g gencodeV19annotation.gtf
                wgEncodeBroadHistoneGm12878H3k4me3StdAlnRep2.bam
```

Figures 2A and 2B depict the number of coverage islands in each biological replicate that meet a sufficient read depth (e.g., “frequency x at percentage y ” means that there exists a total of x coverage islands wherein only y percent of the base pairs in each respective island meet the required depth $-d$). As explained in [Methods](#), this is due to the nature of the $-i$ parameter in merging contiguous genomic regions of sufficient depth ($-d$) that are sufficiently close to each other in distance (within $-i$ bp).

Despite an approximately similar set of annotations in the two biological replicates (Figures 2C and 2D), the distribution of coverage islands that meet the sufficient depth is drastically different (Figures 2A and 2B). In fact, most coverage islands in replicate A have less than 10% of base pairs that meet the sufficient depth ($-d = 20$), suggesting a significantly more diffuse signal enrichment profile relative to replicate B. Likewise, coverage islands in replicate A exhibit a significantly lower variance in length (measured from end-to-end of each island) across the BAM file relative to replicate B (Figures 3A and 3B). Furthermore, the library size graph of replicate A indicates that there are many more outliers of coverage islands that contain extremely large quantities of reads per island, despite having a lower median value than replicate B (Figures 3C and 3D). This suggests that replicate A is dominated by a large number of oversized coverage islands that, taken in consideration with Figure 2A, contain an abundance of shallow reads that do not meet the sufficient depth. It is precisely this abundance of reads failing to meet the sufficient depth that gives rise to such a high number of coverage island outliers with huge library sizes (i.e., total sum of reads per island). Taken together, this suggests that replicate A contains an elevated count of scattered contiguous regions of sufficient depth that are closely separated (within 500 bp of each other), suggesting a more fractured alignment pattern than replicate B. By studying the percentage graph of replicate A at progressively diminishing values of $-i$ (Figure 4A-F), it becomes clear that coverage islands in this replicate sample are separated by approximately 100 bp from each other on average. Comprehensive analyses of percentage, annotation, island diameter, and library size graphs for both biological replicates are provided as supplementary materials for download here: https://github.com/Bohdan-Khomtchouk/SUPERmerge/tree/master/output/interval_figures.

Discussion

The analysis of coverage islands in the histone modification ChIP-seq experiment presented as a case-study in [Results](#) highlights the underlying freedom of the user

in testing different sets of input parameters in the SUPERmerge algorithm for exploratory data analysis purposes. Since many histone modification ChIP-seq experiments are composed of only a few biological replicates per group, this approach is likely to prove useful in making critical downstream analysis decisions, such as whether or not and how to pool or remove replicates during the subsequent peak calling and differential enrichment steps. Considerations involving the specific biology of various chromatin marks will also likely inform the results. For instance, the H3K4me3 study presented above suggests a relatively narrow domain of separation between coverage islands for this specific histone mark, which is in line with its known biological role as the methylation state associated with transcriptional start sites of actively transcribed genes [4].

Since large-scale ChIP-seq analyses are known to be sensitive to the quality of input samples, this has often necessitated the need to filter poor quality data automatically [21]. However, in experiments with naturally low sample sizes and yields (e.g., brain tissue [3, 19]), this is a risky and wasteful procedure. Even though larger numbers of biological replicates are known to establish better concordance and extracted signal [30], this is clearly not always possible. Seeing as the sample size issue will likely continue to impact the ChIP-seq field in many areas of tissue research (e.g., brain), SUPERmerge has been designed to mediate and navigate the aforementioned conflicts by allowing users to explore within-group characteristics of individual alignment (BAM) files, thereby highlighting differences in read alignment topology across single biological replicates of the same group and aiding the user in planning subsequent analysis decisions. All in all, this approach is designed to complement and inform the many peak caller analysis approaches currently in widespread use by the scientific community.

Conclusions

We have designed a novel bioinformatics algorithm implemented as a cross-platform command-line program, SUPERmerge, for the analysis and annotation of coverage islands within individual read alignment (BAM) files of histone modification ChIP-seq datasets harboring broad chromatin domains. Coverage islands are defined as contiguous genomic regions of read pileup that are isolated from their nearest neighbors by some genomic distance (in bp) on both the 3-prime and 5-prime ends. By examining coverage island formations at a single base pair resolution across individual biological replicates, SUPERmerge is designed to be used as an exploratory data analysis tool that complements existing peak calling and quality control algorithms to inform users when to pool or remove samples from analysis based on the consistency and concordance of biological replicates to each other.

Author information

Center for Therapeutic Innovation and Department of Psychiatry and Behavioral Sciences, University of Miami Miller School of Medicine, 1120 NW 14th St., Miami, FL, USA 33136. Correspondence: b.khomtchouk@med.miami.edu.

Competing interests

The authors declare that they have no competing interests.

Author's contributions

BBK conceived the study and wrote the paper. BBK and DV wrote the code. CW participated in the management of the source code and its coordination. All authors read and approved the final manuscript.

Acknowledgements

BBK dedicates this work to the memory of his uncle, Taras Khomchuk. BBK wishes to acknowledge the financial support of the United States Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program: this research was conducted with Government support under and awarded by DoD, Army Research Office (ARO), National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a.

Author details

¹Center for Therapeutic Innovation and Department of Psychiatry and Behavioral Sciences, University of Miami Miller School of Medicine, 1120 NW 14th St, Miami, FL, USA 33136. ²Hussman Institute for Human Genomics, 1501 NW 10th Ave, Miami, USA 33136.

References

- Andrews S. (2017). FastQC: a quality control tool for high throughput sequence data. Available online at: <http://www.bioinformatics.babraham.ac.uk/projects/fastqc>
- Bailey T, Krajewski P, Ladunga I, Lefebvre C, Li Q, Liu T, Madrigal P, Taslim C, Zhang J: *Practical Guidelines for the Comprehensive Analysis of ChIP-seq Data*. PLoS Computational Biology. 2013, 9(11): e1003326.
- Barbier E, Johnstone AL, Khomtchouk BB, Tapocik JD, Pitcairn C, Rehman F, Augier E, Borich A, Schank JR, Rienas CA, Van Booven DJ, Sun H, Nätt D, Wahlestedt C, Heilig M: *Dependence-induced increase of alcohol self-administration and compulsive drinking mediated by the histone methyltransferase PRDM2*. Molecular Psychiatry. 2016, Nature Publishing Group. doi: 10.1038/mp.2016.131.
- Barski A, Cuddapah S, Cui K, Roh TY, Schones DE, Wang Z, Wei G, Chepelev I, Zhao K: *High-resolution profiling of histone methylations in the human genome*. Cell. 2007, 129: 823–837. <http://epigenie.com/key-epigenetic-players/histone-proteins-and-modifications/histone-h3k4/>
- Bernstein BE, Meissner A, Lander ES: *The mammalian epigenome*. Cell. 2007, 128: 669–681.
- Boyle AP, Guinney J, Crawford GE, Furey TS: *F-Seq: a feature density estimator for high-throughput sequence tags*. Bioinformatics. 2008, 24(21): 2537–2538.
- Carroll TS, Liang Z, Salama R, Stark R, de Santiago I: *Impact of artifact removal on ChIP quality metrics in ChIP-seq and ChIP-exo data*. Frontiers in Genetics. 2014, 5(75): 1–11.
- Carroll T, Stark R (2016): *Assessing ChIP-seq sample quality with ChIPQC*. <http://bioconductor.org/packages/devel/bioc/vignettes/ChIPQC/inst/doc/ChIPQC.pdf>
- ENCODE ChIP-seq Experiment Matrix hg19: <https://genome.ucsc.edu/ENCODE/dataMatrix/encodeChipMatrixHuman.html>
- Hadfield J, Eldridge MD: *Multi-genome alignment for quality control and contamination screening of next-generation sequencing data*. Frontiers in Genetics. 2014, 5(31): 1–4.
- Heinig M, Colomé-Tatché M, Taudt A, Rintisch C, Schafer S, Pravenec M, Hubner N, Vingron M, Johannes F: *histoneHMM: Differential analysis of histone modifications with broad genomic footprints*. BMC Bioinformatics. 2015, 16:60.
- Ji H, Jiang H, Ma W, Johnson DS, Myers RM, Wong WH: *An integrated software system for analyzing ChIP-chip and ChIP-seq data*. Nature Biotechnology. 2008, 26, 1293–1300.
- Jothi R, Cuddapah S, Barski A, Cui K, Zhao K: *Genome-wide identification of in vivo protein-DNA binding sites from ChIP-seq data*. Nucleic Acids Research. 2008, 36, 5221–5231.
- Kharchenko PK, Tolstorukov MY, Park PJ: *Design and analysis of ChIP-seq experiments for DNA-binding proteins*. Nature Biotechnology. 2008, 26(12): 1351–1359.
- Khomtchouk BB, Van Booven DJ, Wahlestedt C: *geneXtender: R/Bioconductor package for functional annotation of histone modification ChIP-seq data in a 3D genome world*. bioRxiv. 2016, <http://dx.doi.org/10.1101/082347>.
- Landt SG, Marinov GK, Kundaje A, Kheradpour P, Pauli F, Batzoglou S, Bernstein BE, Bickel P, Brown JB, Cayting P, Chen Y, DeSalvo G, Epstein C, Fisher-Aylor KI, Euskirchen G, Gerstein M, Gertz J, Hartemink AJ, Hoffman MM, Iyer VR, Jung YL, Karmakar S, Kellis M, Kharchenko PV, Li Q, Liu T, Liu XS, Ma L, Milosavljevic A, Myers RM, Park PJ, Pazin MJ, Perry MD, Raha D, Reddy TE, Rozowsky J, Shores N, Sidow A, Slattery M, Stamatoiyannopoulos JA, Tolstorukov MY, White KP, Xi S, Farnham PJ, Lieb JD, Wold BJ, Snyder M: *ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia*. Genome Research. 2012, 22 (9): 1813–1831.
- Li Q, Brown JB, Huang H, Bickel PJ: *Measuring reproducibility of high-throughput experiments*. Annals of Applied Statistics. 2011, 5(3): 1752–1779.
- Li Q (2014). *idr: Irreproducible discovery rate*. R package version 1.2. <https://CRAN.R-project.org/package=idr>
- Maze I, Feng J, Wilkinson MB, Sun H, Shen L, Nestler EJ: *Cocaine dynamically regulates heterochromatin and repetitive element silencing in nucleus accumbens*. 2011. Proceedings of the National Academy of Sciences. 108 (7): 3035–3040.
- Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B: *Mapping and quantifying mammalian transcriptomes by RNA-seq*. Nature Methods. 2008, 5: 621–628.
- Nakato R, Shirahige K: *Recent advances in ChIP-seq analysis: from quality management to whole-genome annotation*. Briefings in Bioinformatics. 2016, 1–12: doi: 10.1093/bib/bbw023.
- Pepke S, Wold B, Mortazavi A: *Computation for ChIP-seq and RNA-seq studies*. Nature Methods. 2009, 6 (11 Suppl): S22–S32.
- R Core Team (2017). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>.
- Rozowsky J, Euskirchen G, Auerbach RK, Zhang ZD, Gibson T, Bjornson R, Carriero N, Snyder M, Gerstein MB: *PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls*. Nature Biotechnology. 2009, 27: 66–75.

25. Sarkar D, Gentleman R, Lawrence M and Yao Z (2016). chipseq: A package for analyzing chipseq data. R package version 1.24.0.
26. Stark R (2014): *Tool: ChIPQC: a package for assessing quality of ChIP-seq samples and experiments.* <https://www.biostars.org/p/98050/>
27. Stark R, Hadfield J: *Characterization of DNA-Protein Interactions: Design and Analysis of ChIP-Seq Experiments.* Field Guidelines for Genetic Experimental Designs in High-Throughput Sequencing. 2016, pp. 223–260: Springer International Publishing Switzerland.
28. Valouev A, Johnson DS, Sunquist A, Medina C, Anton E, Batzoglou S, Myers RM, Sidow A: *Genome-wide analysis of transcription factor binding sites based on ChIP-seq data.* Nature Methods. 2008; 5, 829–834.
29. Wang J, Lunyak VV, Jordan IK: *BroadPeak: a novel algorithm for identifying broad peaks in diffuse ChIP-seq datasets.* Bioinformatics. 2013, 29(4): 492–493.
30. Yang Y, Fear J, Hu J, Haecker I, Zhou L, Renne R, Bloom D, McIntyre LM: *Leveraging biological replicates to improve analysis in ChIP-seq experiments.* Computational and Structural Biotechnology Journal. 2014, 9(13): e201401002.
31. Zang C, Schones DE, Zeng C, Cui K, Zhao K, Peng W: *A clustering approach for identification of enriched domains from histone modification ChIP-Seq data.* Bioinformatics. 2009, 25(15): 1952–1958. <https://github.com/biocompare-ntnu/epic>
32. Zhang Y, Liu T, Meyer CA, Eeckhoutte J, Johnson DS, Bernstein BE, Nusbaum C, Myers RM, Brown M, Li W, Liu XS: *Model-based analysis of ChIP-Seq (MACS).* Genome Biology. 2008, 9(9): R137.

Figures

Abbreviations used

bp: base pairs

ChIP-seq: chromatin immunoprecipitation followed by massively parallel DNA sequencing

IDR: irreproducible discovery rate

QC: quality control

TF: transcription factor