











SOFTWARE

Open Access



GenomicDistributions: fast analysis of genomic intervals with Bioconductor

Kristyna Kupkova^{1,2†} , Jose Verdezoto Mosquera^{1,2†} , Jason P. Smith^{1,2} , Michał Stolarczyk¹ 
, Tessa L. Danehy¹ , John T. Lawson^{1,3} , Bingjie Xue^{1,3} , John T. Stubbs IV^{1,2} , Nathan LeRoy^{1,3}  and
Nathan C. Sheffield^{1,2,3,4*} 

Abstract

Background: Epigenome analysis relies on defined sets of genomic regions output by widely used assays such as ChIP-seq and ATAC-seq. Statistical analysis and visualization of genomic region sets is essential to answer biological questions in gene regulation. As the epigenomics community continues generating data, there will be an increasing need for software tools that can efficiently deal with more abundant and larger genomic region sets. Here, we introduce GenomicDistributions, an R package for fast and easy summarization and visualization of genomic region data.

Results: GenomicDistributions offers a broad selection of functions to calculate properties of genomic region sets, such as feature distances, genomic partition overlaps, and more. GenomicDistributions functions are meticulously optimized for best-in-class speed and generally outperform comparable functions in existing R packages. GenomicDistributions also offers plotting functions that produce editable ggplot objects. All GenomicDistributions functions follow a uniform naming scheme and can handle either single or multiple region set inputs.

Conclusions: GenomicDistributions offers a fast and scalable tool for exploratory genomic region set analysis and visualization. GenomicDistributions excels in user-friendliness, flexibility of outputs, breadth of functions, and computational performance. GenomicDistributions is available from Bioconductor (<https://bioconductor.org/packages/release/bioc/html/GenomicDistributions.html>).

Keywords: Genomic regions, Region set summary, Data visualization, R package, Bioconductor

Background

Sets of genomic regions are a fundamental data type for biological data analysis. They result from a variety of epigenome analysis experiments, such as ChIP-seq or ATAC-seq. Genomic region sets consist of genomic coordinates that specify regions with a shared property. Unlike genes, whose functions are better defined, the functional importance of non-coding genomic regions has been harder to

interpret. To address this issue, multiple tools have been recently developed for a variety of analyses on genomic region sets, such as enrichment analysis (LOLA [1], LOLAweb [2], GIGGLE [3], IGD [4], GREAT [5], epiCOLOC [6]), visualization (chromPlot [7], karyoploteR [8]), region set comparison (BEDTools [9], Bedshift [10], ALList [11], regioneR [12]), or region annotation (Goldmine [13], annotatr [14], ChIPpeakAnno [15], ChIPseeker [16]). Other tools have been developed to classify and infer biological function of region sets (word2vec-based embedding [17], Avocado [18]), identify regulatory activity of regions (MIRA [19]), or to analyze heterogeneity across samples (COCOA [20]). Existing R packages provide some region-based analytical approaches, such

*Correspondence: nsheffield@virginia.edu

[†]Kristyna Kupkova and Jose Verdezoto Mosquera contributed equally to this work.

²Department of Biochemistry and Molecular Genetics, University of Virginia, Charlottesville, USA

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



as visualizing the distribution of genomic regions across chromosomes or annotations (chromPlot [7], karyoplotR [8], annotatr [14]), or for particular types of region sets, (e.g. ChIPpeakAnno [15], ChIPseeker [16] for ChIP-seq data). However, there is no general-purpose R package for extensive visualization and statistical analysis of genomic region sets from any source (Additional file 1: Table S1). Furthermore, many existing packages are not optimized to deal with the growing scale of region data, which now exceeds hundreds of thousands of publicly available region sets and hundreds of billions of individual regions [21–23].

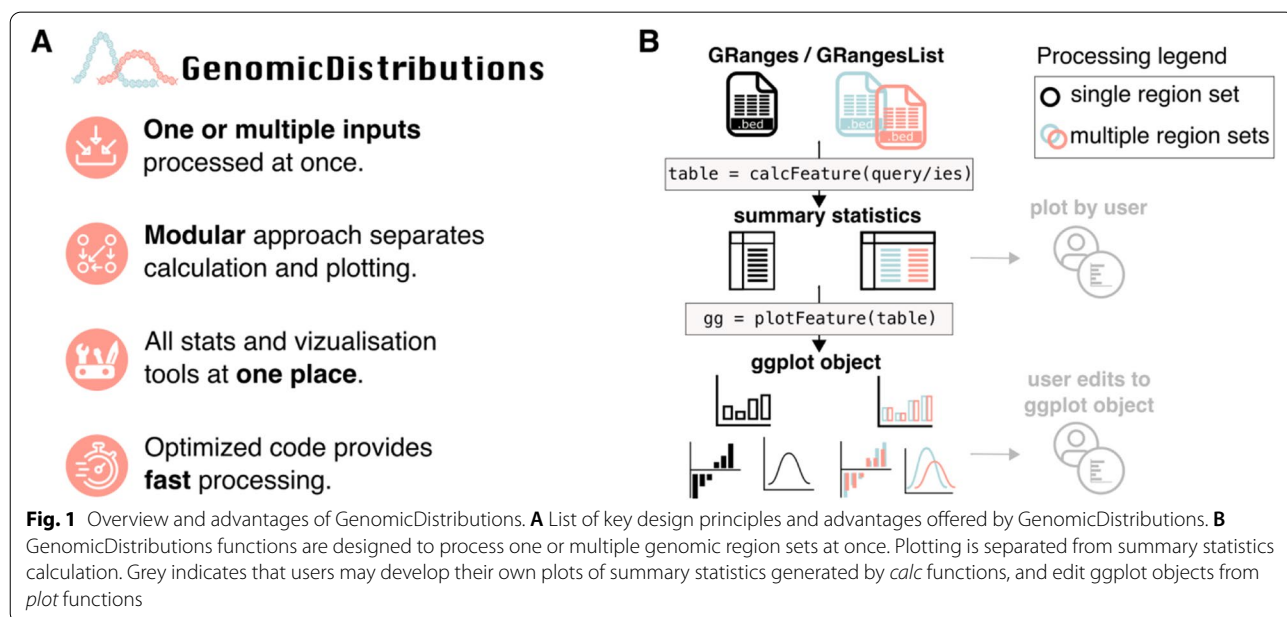
To this end, we introduce the GenomicDistributions R package. GenomicDistributions specializes in computing basic statistics and visualizing distributions of genomic region sets from any experimental source. GenomicDistributions functions compute a variety of statistics and plot results to explore genomic region data. These include *chromosome distribution plots, feature distance plots, neighbor region distance plots, GC content plots, signal summary plots, genomic partition overlap plots, peak width quantile trimmed histogram plots, and dinucleotide frequency plots*. GenomicDistributions offers several key advantages over existing approaches (Fig. 1a): First, we carefully designed the package functions to have a simple, uniform, and flexible interface (Fig. 1b). GenomicDistributions functions all take the same input: a Bioconductor GenomicRanges or GenomicRangesList object, creating a unified interface for the user to summarize one or more region sets with the same line of code. We also separated calculation and plotting functions, enabling users to run calculations for reporting statistics

without directly summarizing these only as plots. The outputs for every calculation function are the inputs for plotting functions. Furthermore, all plotting functions return ggplot2 objects, making it easier for users to adjust style of images. We also put considerable effort into optimizing performance, so GenomicDistributions scales better with large inputs than other R packages for related computations (Additional file 1: Fig. S1). Finally, GenomicDistributions provides a broad array of available analysis types. Its scope is more universal than many existing packages, targeting genomic interval data from any source (Additional file 1: Table S1).

Results and discussion

GenomicDistributions functions calculate and plot summary statistics for single or multiple genomic region sets. These include functions for calculating distribution of regions across chromosomes, distances between neighboring regions, GC content, overlaps with genomic features, nearest distances from genomic features, widths of regions, and dinucleotide frequency, as well as functions for summary of user-provided signal values from different conditions.

Several functions require reference genome feature annotations. Users can either provide their own feature annotations or use wrapper functions (indicated with a “Ref” function name suffix), which require only a string specifying genome assembly as an input, and which automatically extract the feature annotations from the associated GenomicDistributionsData package. The wrapper functions are available for hg19, hg38, mm9, and mm10 reference genome assemblies; for other reference

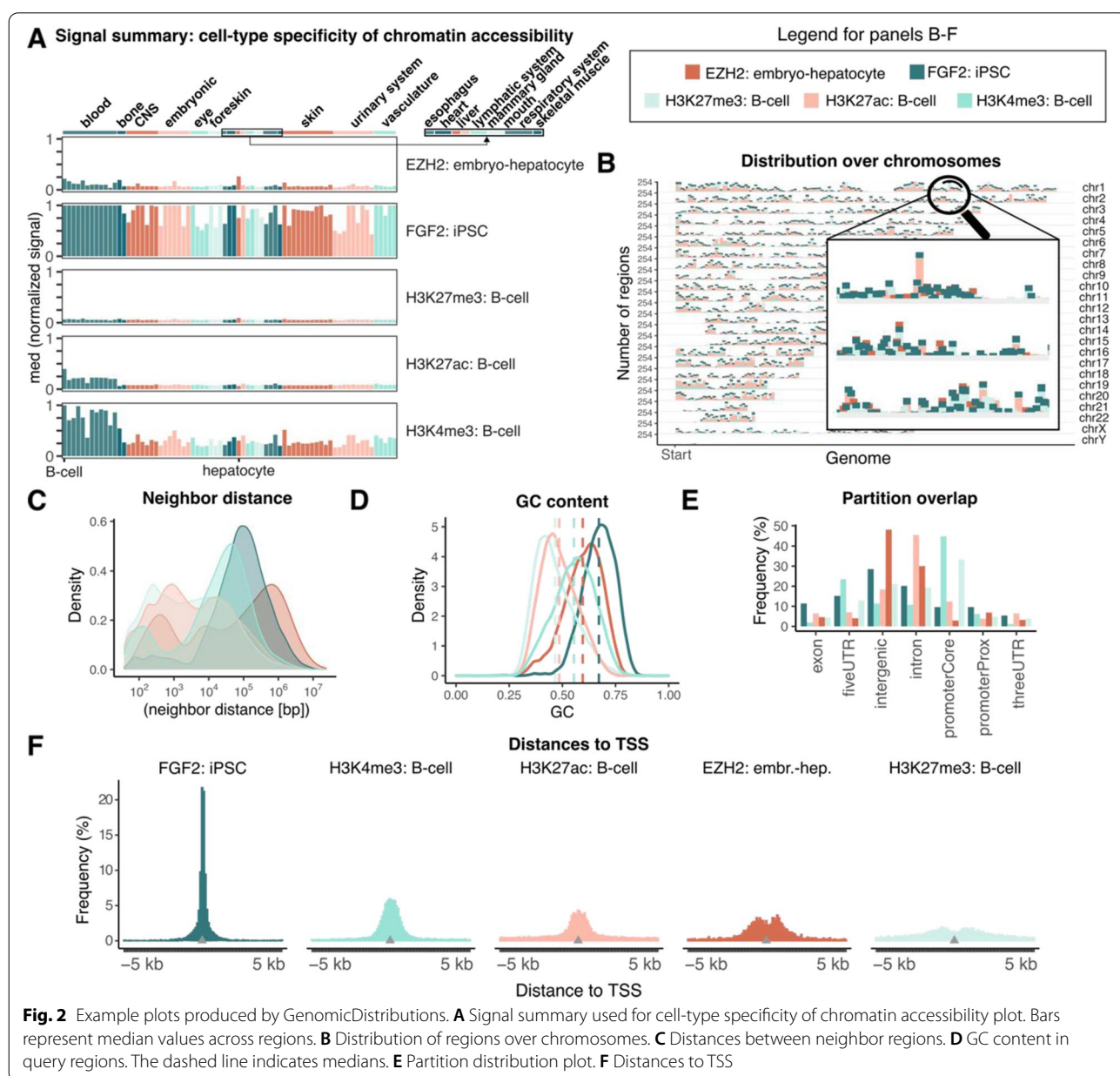


genomes, users can still use GenomicDistributions functions, but will need to supply the annotation data (Additional file 2: Table S2).

To demonstrate the use of GenomicDistributions, we put together a dataset of 5 genomic region sets of various types: EZH2 (Enhancer Of Zeste 2 Polycomb Repressive Complex 2 Subunit) regions in embryonic hepatocyte cells; FGF2 (Fibroblast Growth Factor 2) differentiation factor regions in iPSC (induced Pluripotent Stem Cells); and three sets of histone marks in B-cells, namely H3K27me3, which is associated with heterochromatin; H3K27ac, which is enriched in active

enhancers; and H3K4me3, which is associated with active promoters. For this test set, we applied each GenomicDistributions calculation and plotting function. Here, we show examples of the resulting plots.

First, the signal summary function summarizes external signal data across query regions. This function requires a user-provided matrix with normalized signal values across a genome. GenomicDistributionsData provides a pre-constructed matrix of normalized chromatin accessibility signal values across genomes (hg19, hg38, mm10) of different cell types, which can be used to infer cell-type specificity of chromatin accessibility in the query regions



(Fig. 2a). The output represents a summary of chromatin accessibility signal within test regions across different cell types. Next, the chromosome distribution plot (Fig. 2b) helps to visualize how the regions are distributed across chromosomes. The neighbor region distance plot (Fig. 2c) shows the distribution of distances between two consecutive regions in a sorted region set. The GC content plot (Fig. 2d) displays the distribution of GC content percentage within genomic regions of interest. The genomic partition distribution plot (Fig. 2e) shows how regions are distributed across genome annotation classes. Users can either provide features of interest or use the *calcPartitionsRef* function with pre-defined elements including core promoters, proximal promoters, exons, introns, 5' UTRs (untranslated regions), 3' UTRs, and intergenic regions (Additional file 2: Table S2). In addition to raw partition distributions plots (Fig. 2e), GenomicDistributions also offers expected partition distribution plots (Additional file 1: Fig. S2). Expected partition distribution plots correct the raw overlap counts (observed) by dividing those by expected overlaps, which depend on the size of the partition. This correction is particularly important, since the sizes of individual partitions such as exons vs. introns are considerably different, and therefore we expect more regions to overlap introns than exons by chance. The corrected overlap values are presented as the $\log_{10}(\text{observed}/\text{expected})$ overlap count for each partition. We then use the Chi-square independence test to calculate the *p*-values inferring the significance of the observed overlaps compared to expected (Additional file 1: Table S3). GenomicDistributions also provides a novel plot type that further extends this concept, called cumulative partition distribution plots (Additional file 1: Fig. S3, Supplementary methods). The cumulative partition distribution plot extends this concept in two ways: 1) it shows not only the total counts, but how they accumulate in total genome coverage when regions are ordered by size; and 2) instead of the fraction of regions in each feature, it shows a combined enrichment score, which is the average of the fraction of regions in each feature and the fraction of the features covered by regions. These two changes make the plots more informative, as they include the total size of bases covered in each fraction, and a more balanced enrichment score that naturally accounts for the different total coverage of each partition (Additional file 1: Fig. S3, Supplementary methods).

Another plot produced by GenomicDistributions is the feature distance plot, which shows how the regions are distributed with respect to the nearest feature of interest. Due to the common use of distances to nearest transcription start site (TSS), we provide the *calcTSSDistanceRef* function for convenience (Fig. 2f). The width distribution plot (Additional file 1: Fig. S4) shows widths of genomic

regions as a histogram with clipped top percentiles, a feature that allows enhanced visual comparison by eliminating long tails. Finally, dinucleotide frequency plots (Additional file 1: Fig. S5) shows the distribution of dinucleotide content within genomic regions of interest.

Our design philosophy for GenomicDistributions included several key concepts that provide advantages over other tools with similar purpose: First, we separated calculation functions from plotting functions (Fig. 1b). While many visualization-based analysis functions run calculations and plotting at the same time (Additional file 1: Table S1), by decoupling them, GenomicDistributions provides the flexibility to use the two independently so the intermediate results can be used for other purposes in downstream tools. Each analysis is thus done in two steps: first, by calling a *calc* function, which returns a summarized set of computed statistics; and second, by passing this result to a *plot* function.

We also designed a consistent user interface for GenomicDistributions functions. Each *calc* function can accept either a GenomicRanges object representing a single set of intervals, or a GenomicRangesList object with multiple sets of intervals (Fig. 1a, b). This provides a single interface for either individual region set exploration or to compare among region sets. The result of all *calc* functions can then be used as direct input into a corresponding *plot* function. In turn, each *plot* function returns a ggplot object which can then be further styled by the user. This consistency across functions makes it simple to learn how to use the package and to run multiple analyses on a single input.

Most of the *plot* functions offer multiple plotting options. For example, the feature distance distribution plot offers: 1) the default histogram option (Fig. 2f); and 2) a heatmap option (Additional file 1: Fig. S6a). Similarly, the signal summary plot can be visualized as a bar plot with any defined groups included (Fig. 1a), or, for example, as a violin plot showing a subset of predefined groups, such as tissues or cell types (Additional file 1: Fig. S6b). In addition to the carefully designed user interfaces, we also took considerable effort to optimize GenomicDistributions for speed. By leveraging the highly optimized code in the R data.table package [24] and using fast rolling joins, GenomicDistributions calculation functions are much faster than alternatives in other R packages.

To showcase the speed efficiency and scalability of GenomicDistributions, we carried out a running time benchmark against relevant R packages with related functions (Additional file 1: Table S1). Specifically, we tested the speed of four analyses: 1) distribution of regions across genomic partitions, 2) distance of regions to TSSs 3) distribution of regions across chromosomes and 4) distance of regions to user-defined features.

To perform this benchmark, we assembled a collection of six ChIP-seq region sets displaying variability in terms of region number (less than 10,000 to more than 300,000 regions) and region widths. Narrow regions are represented by transcription factor region sets (TCF12: Transcription Factor 12, ATF3: Activating Transcription Factor 3; MEF2C: Myocyte Enhancer Factor 2C), while broader regions originate from histone mark region sets (for region set details see Additional file 1: Table S4, Supplementary methods). The running time of GenomicDistributions *calc/plot* functions is generally lower, and often much lower, than competing packages (Additional file 1: Fig. S1). Furthermore, GenomicDistributions tends to scale much better with increasing number of regions.

Conclusions

GenomicDistributions is an R package with a broad set of functions available in one place with the purpose to explore genomic regions. While other currently available tools provide some of the summary functions, GenomicDistributions is the most feature-rich, in terms of the number and type of statistics/plots that are produced. GenomicDistributions also takes a step forward on ease-of-use through our modular, consistent programming design. Multiple region sets can be analyzed as easily as a single one, and for common reference genomes, GenomicDistributions can make use of our pre-compiled genome annotations. When more flexibility is needed, GenomicDistributions is also adaptable enough to be used with any reference assembly. Not only can users provide their own annotations, or features of interest, we also give them the freedom to visualize the results in their own way either by using the output of *calc* functions, or by editing ggplot objects returned by *plot* functions. Lastly, GenomicDistributions is substantially faster and more scalable than other publicly available R packages. Together, these strengths make GenomicDistributions a fast, flexible, powerful, and easy-to-use package for analysis of genomic region set data.

Methods

The package is available from <https://bioconductor.org/packages/release/bioc/html/GenomicDistributions.html>, or <https://github.com/databio/GenomicDistributions>. Detailed methods description is available in Additional file 1: Supplementary methods.

Abbreviations

EZH2: Enhancer of Zeste 2 Polycomb Repressive Complex 2 Subunit; FGF2: Fibroblast Growth Factor 2; H3K27me3: Histone H3 lysine 27 trimethylation; H3K27ac: Histone H3 lysine 27 acetylation; H3K4me3: Histone H3 lysine 4 trimethylation; H3K4me1: Histone H3 lysine 4 monomethylation; iPSC: Induced

Pluripotent Stem Cells; TSS: Transcription start site; UTR: Untranslated regions; TCF12: Transcription Factor 12; ATF3: Activating Transcription Factor 3; MEF2C: Myocyte Enhancer Factor 2C; hESC: Human Embryonic Stem Cell.

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12864-022-08467-y>.

Additional file 1. Contains supplementary methods, supplementary Figs. (S1–S6), supplementary Tables (S1, S3 and S4) and supplementary references.

Additional file 2: Supplementary Table S2.

Acknowledgements

We acknowledge support, contributions, and helpful discussions from other Sheffield lab members. We also appreciate helpful insights from David Auble.

Authors' contributions

Conceived of the study and supervised the project: NCS. Wrote the paper: KK, JVM and NCS, with contributions from JPS, MS, TLD, JTL, BX, JTS and NL. Led the software development: KK, JVM, and NCS. Contributed code used in the package: KK, JVM, NCS, JPS, MS, TLD, JTL, BX, JTS and NL. The authors read and approved the final manuscript.

Funding

This work was supported by the National Institute of General Medical Sciences grant GM128636 (NCS). KK was supported by UVA Wagner fellowship. JTL was partially supported by the UVA Cancer Center. The funding bodies played no role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials

The test dataset was obtained from ENCODE [23] (accession numbers: ENCF-F869UBZ, ENCF539PUL, ENCF969HEX), and CISTROME Data Browser [21, 22] (GEO accession numbers: GSM2698625, GSM2439179). The benchmark region sets were obtained from ENCODE (accession numbers: ENCF264OCQ, ENCF543TAQ, ENCF145FYQ, ENCF647QUI, ENCF720VWD, ENCF130BPH).

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

NCS is a consultant for InVivoCell Research, LLC.

Author details

¹Center for Public Health Genomics, University of Virginia, Charlottesville, USA. ²Department of Biochemistry and Molecular Genetics, University of Virginia, Charlottesville, USA. ³Department of Biomedical Engineering, University of Virginia, Charlottesville, USA. ⁴Department of Public Health Sciences, University of Virginia, Charlottesville, USA.

Received: 25 October 2021 Accepted: 13 March 2022

Published online: 12 April 2022

References

1. Sheffield NC, Bock C. LOLA: enrichment analysis for genomic region sets and regulatory elements in R and bioconductor. *Bioinformatics*. 2016;32(4):587–9.

2. Nagraj VP, Magee NE, Sheffield NC. LOLAweb: a containerized web server for interactive genomic locus overlap enrichment analysis. *Nucleic Acids Res.* 2018;46(W1):W194–9.
3. Layer RM, Pedersen BS, Disera T, Marth GT, Gertz J, Quinlan AR. GIGGLE: a search engine for large-scale integrated genome analysis. *Nat Methods.* 2018;15(2):123–6 [cited 2021 Jun 25]. Available from: <https://www.nature.com/articles/nmeth.4556>.
4. Feng J, Sheffield NC. IGD: high-performance search for large-scale genomic interval datasets. *Bioinformatics.* 2021;37(1):118–20.
5. McLean CY, Bristor D, Hiller M, Clarke SL, Schaar BT, Lowe CB, et al. GREAT improves functional interpretation of cis-regulatory regions. *Nat Biotechnol.* 2010;28(5):495–501.
6. Zhou Y, Sun Y, Huang D, Li MJ. epiCOLOC: integrating large-scale and context-dependent Epigenomics features for comprehensive Colocalization analysis. *Front Genet.* 2020;11:53.
7. Oróstica KY, Verdugo RA. chromPlot: visualization of genomic data in chromosomal context. *Bioinformatics.* 2016;32(15):2366–8.
8. Gel B, Serra E. karyoploteR: an R/bioconductor package to plot customizable genomes displaying arbitrary data. Hancock J, editor. *Bioinformatics.* 2017;33(19):3088–90.
9. Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics.* 2010;26(6):841–2.
10. Gu A, Cho HJ, Sheffield NC. SHORT REPORT Bedshift: perturbation of genomic interval sets. *bioRxiv.* 2020;12:2020.11.11.378554.
11. Feng J, Ratan A, Sheffield NC. Augmented interval list: a novel data structure for efficient genomic interval search. *Bioinformatics.* 2019;35(23):4907–11.
12. Gel B, Díez-Villanueva A, Serra E, Buschbeck M, Peinado MA, Malinverni R. regioneR: an R/bioconductor package for the association analysis of genomic regions based on permutation tests. *Bioinformatics.* 2016;32(2):289–91.
13. Bhasin JM, Ting AH. Goldmine integrates information placing genomic ranges into meaningful biological contexts. *Nucleic Acids Res.* 2016;44(12):5550–6.
14. Cavalcante RG, Sartor MA. Annotatr: genomic regions in context. Valencia A, editor. *Bioinformatics.* 2017;33(15):2381–3.
15. Zhu LJ, Gazin C, Lawson ND, Pagès H, Lin SM, Lapointe DS, et al. ChIP-peakAnno: a bioconductor package to annotate ChIP-seq and ChIP-chip data. *BMC Bioinformatics.* 2010;11(1):237.
16. Yu G, Wang L-G, He Q-Y. ChIPseeker: an R/bioconductor package for ChIP peak annotation, comparison and visualization. *Bioinformatics.* 2015;31(14):2382–3.
17. Gharavi E, Gu A, Zheng G, Smith JP, Zhang A, Brown DE, et al. Embeddings of genomic region sets capture rich biological associations in lower dimensions. *Bioinformatics.* 2021;37(23):4299–306. <https://doi.org/10.1093/bioinformatics/btab439>.
18. Schreiber J, Durham T, Bilmes J, Noble WS. Avocado: a multi-scale deep tensor factorization method learns a latent representation of the human epigenome. *Genome Biol.* 2020;21(1):1–18.
19. Lawson JT, Tomazou EM, Bock C, Sheffield NC. MIRA: an R package for DNA methylation-based inference of regulatory activity. *Bioinformatics.* 2018;34(15):2649–50.
20. Lawson JT, Smith JP, Bekiranov S, Garrett-Bakelman FE, Sheffield NC. COCOA: coordinate covariation analysis of epigenetic heterogeneity. *Genome Biol.* 2020;21(1):1–23.
21. Mei S, Qin Q, Wu Q, Sun H, Zheng R, Zang C, et al. Cistrome data browser: a data portal for ChIP-Seq and chromatin accessibility data in human and mouse. *Nucleic Acids Res.* 2017;45(D1):D658–62.
22. Zheng R, Wan C, Mei S, Qin Q, Wu Q, Sun H, et al. Cistrome data browser: expanded datasets and new tools for gene regulatory analysis. *Nucleic Acids Res.* 2019;47(D1):D729–35.
23. Davis CA, Hitz BC, Sloan CA, Chan ET, Davidson JM, Gabdank I, et al. The encyclopedia of DNA elements (ENCODE): data portal update. *Nucleic Acids Res.* 2018;46(D1):D794–801.
24. Dowle M, Srinivasan A, Gorecki J, Chirico M, Stetsenko P, Short T, et al. data.table: Extension of `data.frame` [Internet]. 2021. Available from: <https://rdatatable.gitlab.io/data.table/>

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

