A flexible ChIP-sequencing simulation toolkit

An Zheng1, Michael Lamkin2, Yutong Qiu1,3, Kevin Ren4, Alon Goren5* and Melissa Gymrek1,5*

Background
Chromatin immunoprecipitation followed by next generation sequencing (ChIP-seq) is a widely used technology for genome-wide mapping of the location of histone modifications (HMs) or DNA-associated proteins such as transcription factors (TFs) and chromatin regulators (CRs) [1]. Dozens of methods have been developed for quantitatively analyzing ChIP-seq data, including peak callers [2, 3] and differential binding tools [4, 5]. A major challenge in training and evaluating these methods as well as interpreting their results is a lack of reliable ground truth data: in most cases, the actual locations and strengths of binding sites or regions enriched for certain histone modifications are not known and cannot be reliably measured using orthogonal experimental techniques.

Computational analysis of ChIP-seq is further complicated by multiple sources of noise introduced during the experimental process, including inefficiency or non-specificity of antibodies, PCR artifacts, and sequencing errors [6, 7].

Abstract
Background: A major challenge in evaluating quantitative ChIP-seq analyses, such as peak calling and differential binding, is a lack of reliable ground truth data. Accurate simulation of ChIP-seq data can mitigate this challenge, but existing frameworks are either too cumbersome to apply genome-wide or unable to model a number of important experimental conditions in ChIP-seq.

Results: We present ChiPs, a toolkit for rapidly simulating ChIP-seq data using statistical models of key experimental steps. We demonstrate how ChiPs can be used for a range of applications, including benchmarking analysis tools and evaluating the impact of various experimental parameters. ChiPs is implemented as a standalone command-line program written in C++ and is available from https://github.com/gymreklab/chips.

Conclusions: ChiPs is an efficient ChIP-seq simulation framework that generates realistic datasets over a flexible range of experimental conditions. It can serve as an important component in various ChIP-seq analyses where ground truth data are needed.

Keywords: Bioinformatics, Epigenomics, ChIP-sequencing, Simulation tool, Command-line program
capture important sources of variation present in real data such as pulldown non-specificity, fragment length variability, or sequencing errors (Additional file 1: Supplementary Table 1). Importantly, existing simulation tools are not capable of inferring model parameters from real ChIP-seq datasets, making it difficult to choose realistic simulation settings.

Here, we present ChIPs (ChIP-seq simulator), a flexible toolkit for rapidly simulating ChIP-seq data based on realistic statistical models. ChIPs is a computationally efficient command-line solution that allows users to easily specify a wide range of parameters modeling key experimental steps and to infer these parameters from existing datasets. We demonstrate the applicability of ChIPs for evaluating the impact of various experimental conditions and for benchmarking computational analysis tools.

Implementation
Framework architecture
ChIPs models each major ChIP-seq step (shearing, immunoprecipitation, pulldown, PCR, and sequencing) as a distinct module (Fig. 1a). It assumes binding sites for the target epitope and their binding scores (probabilities) are known. Notably, for histone modifications, we use binding to refer to genomic localization with the target modification, although the DNA itself is not typically bound by the modification. Importantly, each step is modeled in a way that key parameters can be inferred from existing datasets.

**Step 1: Shearing** Cross-linked DNA is first sheared to a target fragment length, for instance by sonication or enzymatic approaches [12]. ChIPs models fragment lengths using a gamma distribution (Fig. 1a; top) based on empirical observation of fragment distributions which have long right tails. The fragment length distribution parameters are either trivially inferred from paired end read alignments or are approximated from single end data using a heuristic method (Additional file 1: Supplementary Methods, Supplementary Figure 1).

**Step 2: Immunoprecipitation** Sheared cross-linked DNA is subject to immunoprecipitation, during which an antibody is used to enrich the pool of fragments for those bound to the epitope of interest. To model this imperfect process, we quantify the ratio, $\alpha$, of the probability of pulling down a bound versus unbound fragment. This modeled ratio is specific to each ChIP-seq experiment and depends on the antibody specificity as well as the fraction of the genome bound by the factor of interest. Let $f$ be the fraction of the genome bound by the factor of interest and $s$ be the fraction of pulled down reads that originate from true binding sites. We can approximate $\alpha$ using Eq. 1. A detailed derivation of this ratio is provided in Additional file 1: Supplementary Methods.

$$\alpha = \frac{s(1-f)}{(1-s)f}$$ (1)

The parameters $f$ and $s$ can be directly inferred from real data based on binding sites or enriched regions (peaks) identified by various peak-calling methods (Additional file 1: Supplementary Methods, Supplementary Figure 2).

**Step 3: PCR** PCR is used to amplify pulled down fragments before sequencing. Let $n_i$ represent the number of reads (or read pairs) with $i$ PCR duplicates (including the
original fragment). \( n_i \) is modeled using a geometric distribution, where \( p \) gives the probability that a fragment has no PCR duplicates. The parameter \( p \) is estimated as \( 1/\bar{n} \), where \( \bar{n} = \sum_{i=1}^{\infty} (i \bar{n}_i) \).

**Step 4: Sequencing** Finally, amplified fragments are subject to either paired end or single end sequencing. Sequences are based on an input reference genome using the
coordinates of each fragment. We model the per-base pair substitution, insertion, and deletion rates (Additional file 1: Supplementary Table 2).

Implementation details
ChIPs is implemented as an open source C++ project with source code publicly available on Github: https://github.com/gymreklab/chips. It consists of two utilities: simreads and learn (Fig. 1b). The simreads module takes in ChIP-seq model parameters and experimental settings (Additional file 1: Supplementary Table 2), and outputs simulated reads. Input parameters can either be set by the user to mimic a future ChIP-seq experiment or learned from existing data using the learn module. The user must additionally specify the number of simulation rounds, which denotes the number of times the input reference genome is processed by ChIPs. Notably, this number is related, but not directly comparable, to the number of experimentally processed cells, since pulldown efficiency is not directly included in our current model. We have found that in most settings 25–100 and 1000 rounds work well for HMs and TFs, respectively. Full implementation details and methods for benchmarking experiments are provided in Additional file 1: Supplementary Methods.

Results
Comparison of ChIPs simulation results to real ChIP-seq data
We evaluated ChIPs using ChIP-seq data generated by the ENCODE Project [13] for an example histone modification H3K27ac in the GM12878 cell line. To evaluate the effect of varying the number of simulation rounds, we simulated reads on chromosome 22 using parameters inferred from real data over a range of simulation rounds (1–10,000). Run time for chromosome 22 ranged from 11 s (1 round) to 15 min (10,000 rounds). Resulting reads were aligned to the hg19 reference genome using BWA-MEM [14], and duplicates were flagged using Picard [15]. Visual inspection of the resulting coverage profiles shows high similarity between real and simulated data (Fig. 1c).

Next, we compared read counts in bins of 5kb and found high correlation between real and simulated data in bins containing at least one peak (Fig. 1d; Pearson $r = 0.91$; $p < 10^{-200}$; $n = 1,232$ bins; 100 simulation rounds). Further, correlation with ENCODE data increased as a function of the number of simulation rounds but plateaued around 100, suggesting little gain in simulating additional rounds compared to the time tradeoff (Fig. 1e–f). We repeated this analysis on multiple additional HMs and TFs in GM12878 with similar results (Additional file 1: Supplementary Figure 3).

Benchmarking against existing ChIP-seq simulators
We next benchmarked ChIPs against existing ChIP-seq simulators, which are summarized in Additional file 1: Supplementary Table 1. We focused on two recent methods: (1) ChIPulate [9] is a method for simulating TF ChIP-seq data using detailed modeling of locus-specific binding energies. ChIPulate only simulates reads at bound regions, and does not simulate background fragments outside of peak regions, a key feature of real ChIP-seq datasets related to the antibody specificity. (2) isChIP [11] is a command-line method for simulating ChIP-seq data based on a set of input peaks, model parameters, and sequencing parameters. While isChIP performs a similar task to ChIPs, it is not
able to infer model parameters from existing datasets, which is a key feature of ChIPs. A more detailed description of model differences between these tools is provided in Additional file 1: Supplementary Note.

We used ChIPs, ChIPulate, and isChIP to simulate ChIP-seq data based on six different ENCODE datasets including 3 HMs (H3K4me1, H3K4me3, and H3K27ac) and 3 TFs (BCLAF1, IKZF1, and NFYA) (Additional file 1: Supplementary Table 3). For each dataset, we used the three methods to simulate data for chr22 based on ENCODE peaks and with settings meant to capture similar properties of the ENCODE data, including read length and read number. We additionally inferred model parameters using our learn module and used these models to set appropriate simulation options for each tool when possible (Additional file 1: Supplementary Methods). For each tool, we varied the number of simulation rounds (similar to the number of cells) from 1 to 10,000. ChIPulate simulations took approximately 80 min to complete regardless of the number of simulation rounds, although subsequent simulations reused intermediate files and were faster. isChIP consistently achieved the fastest run time (e.g. 0.8 min for 1000 rounds on H3K27ac compared to 4.9 min for ChIPs). For both isChIP and ChIPs, simulation time was far less than the run time of downstream steps of sequence alignment and peak calling.

For each simulated dataset, we compared to real data using two methods. First, similar to above, we aligned simulated reads to the hg19 reference genome and compared read counts in 1kb bins containing at least one peak. As expected, correlation with ENCODE increases for all tools with additional simulation rounds (Additional file 1: Supplementary Figure 4a). In all evaluated conditions we found that ChIPs showed superior correlation with ENCODE data. ChIPs performance was virtually unchanged when using models based on paired versus single end data (Additional file 1: Supplementary Figure 4a).

Second, to evaluate how well each tool captures noise in real data, we examined the distribution of read counts in bins with and without peaks (referred to as peak and background regions, Additional file 1: Supplementary Figure 4b) between simulated and real data. We further visualized these trends using simulated coverage profiles and ENCODE data using the Integrative Genomics Viewer [16] (Additional file 1: Supplementary Figure 5). In all cases, data simulated by ChIPs most closely matches read count distributions in peak versus background regions in the ENCODE data. As expected, almost no reads from ChIPulate align to background regions. For isChIP, we found that using the default background noise level resulted in far higher signal to noise ratios than in the real data. We attempted to more closely match ENCODE data by performing an additional experiment with increased background noise. This in some cases alleviated the bias but still matched less closely than ChIPs data (Additional file 1: Supplementary Figure 4b).

Taken together, our benchmarking results show that ChIPs most accurately captures properties of real ChIP-seq data. Further, whereas ChIPs could learn appropriate model parameters from existing datasets, the alternative tools first required detailed user involvement to determine realistic simulation settings for a particular dataset type. While we cannot rule out that further tuning of parameters for each method could achieve higher correlation, we found that without a method to infer parameters from existing data that it was difficult to choose optimal simulation settings.
Demonstration of ChIPs applications

We next used ChIPs to evaluate the effects of varying experimental parameters on the ability to accurately detect TF or HM peaks. We examined read number, read length, PCR duplicate rate, and antibody specificity, and used ChIPs to simulate a series of datasets by varying each parameter. We generated two sets of simulated data to capture general properties such as peak size distributions characteristic of HMs or TFs (Supplementary Methods). We used MACS2 [2] to call peaks on the resulting datasets after alignment and duplicate marking. Each simulated dataset was evaluated based on the fraction of peaks recovered by the simulated datasets (recall), the fraction of called peaks that were correct (precision), and the combination of precision and recall (F1 score) (Fig. 2a–d, Additional file 1: Supplementary Figure 6). Simulated datasets recapitulate expected trends. Peak calling accuracy increases most dramatically as a function of the total number of reads, and performance decreases for datasets with larger fragment lengths or higher rates of PCR duplicates. Read length and the choice of single versus paired end reads have little impact on peak calling performance in mappable regions of the genome included in our analysis.

Finally, to demonstrate the ability of ChIPs to generate ground truth data for evaluating analysis tools, we compared performance of multiple peak calling methods on simulated datasets. We focused on five representative tools: MACS2 [2], GEM [17], MUSIC [3], BCP [18], and HOMER [19]. We measured peak calling performance using simulated datasets representative of generic HMs or TFs as described above but with varying
degrees of non-specific binding (ChIPs \( s \) parameter, commonly referred to as a SPOT or FRIP score [7]; Fig. 2e–f, Additional file 1: Supplementary Figure 7). As expected, in all settings peak calling performance increased as a function of \( s \). No method achieved superior performance across all datasets or metrics. For TFs, GEM, MACS2, and HOMER showed similarly high F1 scores for datasets with \( s > 0.05 \). For HMs, all tools except BCP showed high F1 scores across a range of \( s \) values. Notably, our analysis captures only a small subset of possible dataset parameters, and it is likely that results will vary depending on specific datasets. Previous work has performed an extensive evaluation of various peak calling methods [20].

**Conclusions**

In summary, we present ChIPs, an efficient command-line program that can rapidly generate realistic ChIP-seq data over a wide range of experimental conditions. ChIPs can infer model parameters from real data and generate simulated data for both TF and HMs. The whole process takes just seconds to minutes for most applications. Our framework is modular, allowing future integration of alternative or improved models at various simulation steps. For example, we can further model multiple types of biases, such as the ones introduced by specific cross-linking steps. Or we can model the biases introduced during pulldown by inherent factors such as GC content or DNA accessibility.

In this study, we benchmarked ChIPs against existing simulation tools and compared simulation results with a broad range of real ChIP-seq datasets as ground-truth. While all these tools could model multiple aspects of ChIP-seq data, we found that ChIPs most closely captures the properties of real ChIP-seq datasets. Another advantage of ChIPs is that, among all simulation tools benchmarked in this study, ChIPs is the only method capable of inferring model parameters from real data, allowing realistic simulation.

We demonstrated the utility of ChIPs in several usage scenarios, including benchmarking peak calling methods and measuring the effects of experimental conditions on peak detection. Some potential future applications include (1) evaluating the effects of genetic variation, such as SNPs, indels, or repeats, on observed ChIP-seq signals, (2) modeling effects of biological processes, such as DNA replication, on ChIP-seq signals, and (3) analyzing effects of spike-in normalization controls. Overall, we envision our framework will serve as a valuable resource for future efforts in ChIP-seq analysis.

**Availability and requirements**

- Project name: ChIP-seq simulator.
- Project home page: https://github.com/gymreklab/chips.
- Operating system: CentOS Linux release 7.8.2003 (Core), macOS Catalina v10.15.7.
- Programming language: C++.
- Other requirements: gcc 4.9.2 or higher.
- License: GNU General Public License v3.0.
- Any restrictions to use by non-academics: None.
Abbreviations

ChIP-seq: Chromatin immunoprecipitation sequencing; TF: Transcription factor; HM: Histone modification; CR: Chromatin regulators; ChIPs: ChIP-seq simulator; PCR: Polymerase chain reaction; ENCODE: Encyclopedia of DNA Elements.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-021-04097-5.

Additional file 1: Supplementary Methods, Supplementary Note, Supplementary Figures and Tables.

Acknowledgements

We thank Drs. Chris Benner and Bing Ren for helpful discussions of the method. We thank Rahel Wachs for assistance in preparing figures. We also thank the American Society of Human Genetics (ASHG) Annual Meeting for publishing the abstract online for a poster session [21].

Authors’ contributions

A.Z. and M.L. developed statistical models and implemented the ChIPs software. Y.Q. conceived statistical models and helped perform benchmarking analyses. K.R. developed a method for estimating fragment length distributions from single-end data. A.G. designed the study and wrote the manuscript. M.G. conceived the study and initial statistical models, designed validation experiments, and wrote the manuscript. All authors have read and approved the manuscript.

Funding

This work was supported in part by NIH/NHGRI Grant 1R21HG010070 to AG and MG. The funding agency did not participate in design or implementation of this work.

Availability of data and materials

The data that support the findings of this study are publicly available from the ENCODE Project [https://www.encodeproject.org/files/]. The accession ID for each dataset can be found in Additional file 1: Supplementary Table 3. The ChIPs source code, installation steps, and usage instructions are available at https://github.com/gymreklab/chips.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

1 Department of Computer Science and Engineering, University of California San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA. 2 Department of Bioengineering, University of California San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA. 3 School of Computer Science, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA. 4 Department of Mathematics, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA. 5 Department of Medicine, University of California San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA.

Received: 8 January 2021   Accepted: 22 March 2021

Published online: 20 April 2021

References

1. Furey, T.S.: ChIP-seq and beyond: new and improved methodologies to detect and characterize protein-DNA interactions. Nat. Rev. Genet. 13(12), 840–852 (2012).
2. Zhang, Y., Liu, T., Meyer, C.A., Eeckhoute, J., Johnson, D.S., Bernstein, B.E., Nussbaum, C., Myers, R.M., Brown, M., Li, W., Liu, X.S.: Model-based analysis of ChIP-Seq (MACS). Genome Biol. 9(9), 137 (2008).
3. Harmanci, A., Rozowsky, J., Gerstein, M.: MUSIC: identification of enriched regions in ChIP-Seq experiments using a mappability-corrected multiscale signal processing framework. Genome Biol. 15(10), 474 (2014).
4. Ross-Innes, C.S., Stark, R., Teschendorff, A.E., Holmes, K.A., Ali, H.R., Dunning, M.J., Brown, G.D., Gojis, O., Ellis, I.O., Green, A.R., Ali, S., Chin, S.F., Palmieri, C., Caldas, C., Carroll, J.S.: Differential oestrogen receptor binding is associated with clinical outcome in breast cancer. Nature 481(7381), 389–393 (2012).
5. Love, M.I., Huber, W., Anders, S.: Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 15(12), 550 (2014).
6. Meyer, C.A., Liu, X.S.: Identifying and mitigating bias in next-generation sequencing methods for chromatin biology. Nat. Rev. Genet. 15(11), 709–721 (2014).
7. Landt, S.G., Marinov, G.K., Kundaje, A., Kheradpour, P., Pauli, F., Batzoglou, S., Bernstein, B.E., Bickel, P., Brown, J.B., Cayting, P., Chen, Y., Marinov, G., Epstein, C., Fisher-Aylor, K.L., Eiserich, G., Gerstein, M., Gertz, J., Hartemink, A.J., Hoffman, M.M., Iyer, V.R., Jung, Y.L., Karmarkar, S., Kellis, M., Kharchenko, P.V., Lim, Q., Liu, T., Liu, X.S., Ma, L., Milosavljevic, A., Myers, R.M., Park, P.J., Ponz, M.J., Perry, M.D., Raha, D., Reddy, T.E., Rozowsky, J., Shores, N., Sidow, A., Slattery, M., Stamatoyannopoulos, J.A., Tolstorukov, M.Y., White, K.P., Xi, S., Farnham, P.J., Lieb, J.D., Lander, E.S., Landt, S. ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. Genome Res. 22(9), 1813–1831 (2012).

8. Humburg, P., Hellwiss, C.A., Bulger, D., Stone, G. ChIPseqR: analysis of ChIP-seq experiments. BMC Bioinformatics 12, 39 (2011).

9. Datta, V., Hannenhalli, S., Siddharthan, R. ChIPulate: A comprehensive ChIP-seq simulation pipeline. PLoS Comput. Biol. 15(3), 1006921 (2019).

10. Zhang, Z.D., Rozowsky, J., Snyder, M., Chang, J., Gerstein, M. Modeling ChIP sequencing in silico with applications. PLoS Comput. Biol. 4(8), 1000158 (2008).

11. Subkhankulova, T., Naumenko, F., Tolmachov, O., Orlov, YL. Novel ChIP-seq simulating program with superior versatility: iChIP Brief Bioinform. 2020.

12. Kidler, B.L., Hu, G., Zhao, K. ChIP-seq technical considerations for obtaining high-quality data. Nat. Immunol. 12(10), 918–922 (2011).

13. Durham, T., Kundaje, A, Aldred, S., Collins, P.J., Davis, C.A., Doyle, F., Epstein, C., Frietze, S., Harrow, J., Kall, R., Khatun, J., Lajoie, B.R., Landt, S.G., Lee, K.B., Pauli, F., Rosenbloom, K.R., Sabo, P.P., Saeki, A., Sanyal, N., Simon, J.M., Song, L., Tinkleind, N.D., Altschuler, R.C., Birney, E., Brown, J.B., Cheng, C., Djebali, S., Dong, X., Dunham, I., Ernst, J., Furey, T.S., Gerstein, M., Giardine, B., Green, M., Hardison, R.C., Harris, R.S., Herrero, J., Hoffman, MM., Iyer, S., Kellis, M., Kharchenko, P.V., Lander, E.S., Zhiping, M., Kidder, B.L., Hu, G., Zhao, K.: ChIP-Seq: technical considerations for obtaining high-quality data. Nat. Immunol. 12(10), 1085–1091 (2011).

14. Zhang, Z.D., Rozowsky, J., Snyder, M., Chang, J., Gerstein, M.: Modeling ChIP sequencing in silico with applications. PLoS Comput. Biol. 4(8), 1000158 (2008).
L, Min R, Mu XJ, Rozowsky J, Yan HK, Yip KY, Birney E. An integrated encyclopedia of D N A elements in the human genome. Nature 2012;489(7414):57-74
14. Li, H.: Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM (2013). arXiv:1303.3997
15. Institute, B. Picard Tools. http://broadinstitute.github.io/picard/. Accessed: 2018-MM-DD; version X.Y.Z. (2018)
16. Robinson, J.T., Thorvalsdóttir, H., Winckler, W., Guttman, M., Lande, E.S., Getz, G., Mesirov, J.P.: Integrative genomics viewer. Nat Biotechnol. 29(1), 34–26 (2011).
17. Guo, Y., Mahony, S., Gifford, D.K.: High resolution genome wide binding event finding and motif discovery reveals transcription factor spatial binding constraints. PLoS Comput. Biol. 8(8), 1002638 (2012).
18. Xing, H., Mo, Y., Liao, W., Zhang, M.Q.: Genome-wide localization of protein-DNA binding and histone modification by a Bayesian change-point method with ChIP-seq data. PLoS Comput. Biol. 8(7), 1002613 (2012).
19. Heinz, S., Benner, C., Spann, N., Bertolino, E., Lin, Y.C., Laslo, P., Cheng, J.X., Murre, C., Singh, H., Glass, C.K.: Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities. Mol. Cell 38(4), 576–589 (2010).
20. Thomas, R., Thomas, S., Holloway, A.K., Pollard, K.S.: Features that define the best ChIP-seq peak calling algorithms. Brief. Bioinformatics 18(3), 441–450 (2017).
21. Lamkin, M., Zheng, A., Qiu, Y., Ren, K., Goren, A., Gymrek, M.: A flexible simulation toolkit for designing and evaluating ChIP-sequencing experiments. Presented at the 69th annual meeting of The American Society of Human Genetics, October 18, 2019 in Houston, TX

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