

SpikeFlow: automated and flexible analysis of ChIP-Seq data with spike-in control

Davide Bressan ^{1,*}, Daniel Fernández-Pérez ², Alessandro Romanel ¹ and Fulvio Chiacchiera ^{1,*}

¹Department of Cellular, Computational, and Integrative Biology, University of Trento, Via Sommarive 9, 38123 Povo - Trento, Italy

²Quantitative Stem Cell Dynamics Laboratory, IRB Barcelona, Carrer de Baldiri Reixac 10, 08028 Barcelona, Spain

*To whom correspondence should be addressed. Tel: +39 0461 283422; Email: davide.bressan-1@unitn.it

Correspondence may also be addressed to Fulvio Chiacchiera. Tel: +39 0461 283934; Email: fulvio.chiacchiera@unitn.it

Abstract

ChIP with reference exogenous genome (ChIP-Rx) is widely used to study histone modification changes across different biological conditions. A key step in the bioinformatics analysis of this data is calculating the normalization factors, which vary from the standard ChIP-seq pipelines. Choosing and applying the appropriate normalization method is crucial for interpreting the biological results. However, a comprehensive pipeline for complete ChIP-Rx data analysis is lacking. To address these challenges, we introduce SpikeFlow, an integrated Snakemake workflow that combines features from various existing tools to streamline ChIP-Rx data processing and enhance usability. SpikeFlow automates spike-in data scaling and provides multiple normalization options. It also performs peak calling and differential analysis with distinct modalities, enabling the detection of enrichment regions for histone modifications and transcription factor binding. Our workflow runs in-depth quality control at all the processing steps and generates an analysis report with tables and graphs to facilitate results interpretation. We validated the pipeline by performing a comparative analysis with DiffBind and SpikChIP, demonstrating robust performances in various biological models. By combining diverse functionalities into a single platform, SpikeFlow aims to simplify ChIP-Rx data analysis for the research community.

Introduction

Precise comparison of transcription factor binding and histone modification levels between conditions is crucial in genome biology. It can reveal novel gene regulatory mechanisms and highlight cellular state changes, helping target therapy development (1). Chromatin immunoprecipitation followed by sequencing (ChIP-seq) is considered the standard approach to study protein-DNA interactions due to its successful adoption in determining epigenomic profiles across cell types and tissues (2–4). However, ChIP-seq faces biases that hinder comparisons across experiments, such as antibody efficiency and specificity. These issues were addressed by integrating exogenous chromatin (spike-in) from a different species as an internal control (1,5,6). This modification to ChIP-seq allows for the normalization of sample-to-sample variation, facilitating more reliable comparative analysis.

The first spike-in normalization method, known as reference-adjusted reads per million (RRPM), was introduced by Orlando *et al.* (6). In this case, the scaling factor is obtained from the number of reads that align to the exogenous genome. Subsequently, several research groups modified RRPM to consider also the spike-in reads derived from the input sample in the scaling factor calculation (7,8). This approach, henceforth named Rx-input, enhances the normalization by adjusting for variations in immunoprecipitation efficiency and background noise (i.e. the input). Alternatively, the downsampling (or tag removal) method scales down all the samples to the minimum

number of spike-in reads in the dataset (1,9). Similarly, all the samples can be normalized based on the median number of exogenous reads in the dataset (10). Recently, a new type of normalization based on local regression was proposed, which reduces the spike-in correction in background regions (9). Applying the correct normalization is critical in ChIP-Rx data analysis to ensure a proper interpretation of transcription factors or histone modification changes.

Implementing the aforementioned strategies can be challenging and time-intensive, especially for non-experts in the field. Moreover, spike-in normalizations are not included in standard ChIP-seq data analysis workflows. To overcome these limitations, we introduce SpikeFlow, a novel pipeline for analyzing ChIP-Rx data. Starting from raw data, our workflow performs trimming, alignment, and generation of normalized bigWigs files suited for signal comparison across conditions. The tool implements five types of normalization, namely RPM, RRPM, Rx-input, Downsampling and Median. Additionally, SpikeFlow features three diverse peak calling methods and performs robust quality control at all the key processing steps. For narrow and broad peak calling, SpikeFlow implements both standard and normalized peak calling. Standard peak calling is performed on the bam files without normalization, common in ChIP-seq data analysis (11). The user can, however, choose to perform the peak calling on the spike-in normalized tracks. We implemented this option by taking advantage of *Spiker* (10), a recently developed

Received: June 12, 2024. Revised: August 2, 2024. Editorial Decision: August 19, 2024. Accepted: August 20, 2024

© The Author(s) 2024. Published by Oxford University Press on behalf of NAR Genomics and Bioinformatics.

This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License

(<https://creativecommons.org/licenses/by-nc/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact reprints@oup.com for reprints and translation rights for reprints. All other permissions can be obtained through our RightsLink service via the Permissions link on the article page on our site—for further information please contact journals.permissions@oup.com.

tool that modifies MACS2 behavior. Once peak calling is performed, SpikeFlow will create a consensus set of regions and perform a differential binding analysis based on DESeq2 (12) for the contrasts specified by the user. Finally, the main output of SpikeFlow is a multiqc report, which contains tables and plots with the analysis results.

Other tools have been proposed in recent years to address ChIP-Rx data analysis. For instance, *SpikChIP* (9) and *ChIPSeqSpike* (13). The former introduces a novel normalization approach focused on peak regions and compares it to RRPM and Downsampling scaling. The latter is an R package that we could not test since it is discontinued and, therefore, unavailable with recent versions of R. The main output of *SpikChIP* are text files, which are not signal track files and thus can not readily be used for downstream analysis. A further conversion step needs to be implemented by the user, introducing a layer of complexity to the analysis and potential biases. Moreover, both *ChIPSeqSpike* and *SpikChIP* request the user to input two bam files per sample, obtained by aligning the reads against the endogenous and exogenous genomes. On the contrary, *SpikeFlow* starts the analysis from the raw data and takes care of the alignment step with the related quality controls. A similar approach is implemented in *Spiker* (10). This tool, however, does not allow automatic computation of scaling factors, which is left to the user (Table 1).

DiffBind (14) is a popular tool used for differential binding analysis of ChIP-seq data that also includes spike-in normalization. This tool requires a minimum of two replicates per condition, which are not always available for ChIP-Rx experiments. In SpikeFlow, we handle both single or multiple replicates per biological condition. Finally, none of the aforementioned published tools implements the Rx-input normalization strategy.

To our knowledge, SpikeFlow is the first pipeline to implement all the steps of ChIP-Rx data processing, from the alignment to the differential peaks analysis. This greatly simplifies the researcher's work and favors a better interpretation of biological results. Furthermore, our tool implements a broad range of normalizations and performs standard and normalized peak calling. We tested SpikeFlow on multiple datasets, including MV4-11 leukemia cells (6), breast cancer cell lines (15), mouse embryonic cortex (16), and mouse hepatocytes (17) validating its efficacy and applicability in various scenarios. As a proof-of-concept, we compared SpikeFlow's performance on ChIP-Rx data of ER binding (15) to *SpikChIP* and *DiffBind*, obtaining highly similar results. Finally, SpikeFlow is implemented with Snakemake, making it suited for multi-threading and highly portable to all operating systems. Its modular design enables easy integration into larger workflows, and it can be run with a docker or singularity container (see materials and methods).

Materials and methods

SpikeFlow allows the researcher to analyze ChIP-Rx data with minimal parameter configuration required. It is implemented with Snakemake (v7.31), a python-based workflow engine available within the conda package manager (18). The workflow uses several bioinformatics tools, organizing them in separate conda environments, thus isolating dependencies and preventing conflicts, ensuring the workflow can be reproduced in different computational environments. Moreover, all the required conda environments are also installed in a Docker im-

age that can be pulled from Dockerhub during the execution (available at this link <https://hub.docker.com/r/davidebrex/spikeflow>). The key features of SpikeFlow are sample normalization with related quality controls, peak calling, annotation, signal tracks generation, and differential peak analysis. **Supplementary Figure S1** visualizes a flowchart with the main analysis steps. The pipeline requires the ChIP-Rx fastq files, a sample sheet, and a configuration file as input. Both single and paired-end Illumina short-read sequencing technologies are supported. Firstly, if present, lanes from the same sample are merged into a single file. Next, trimming is performed with *fastp* to remove adapters and low-quality reads (19). At this point, if not already provided by the user, an index is generated with *bowtie2* (20) from a synthetic genome made by concatenating the endogenous and exogenous (spike-in) genomes. After the alignment, duplicate reads are removed with *sambler* (21). A custom Python script based on samtools is then applied to separate the reads into two bam files based on the chromosome name. To generate the bigWig files, SpikeFlow takes advantage of *deeptools* (22). We implemented five distinct normalization methodologies. The first utilizes the conventional Reads per Million (RPM) normalization strategy. This technique does not incorporate the spike-in for the signal correction. Next, the Orlando (RRPM) normalization applies the scaling by dividing the reads mapped to the sample endogenous genome by the number of reads mapped to the exogenous genome (5,6). RRPM assumes consistent spike-in read quantities across various samples, ensuring an accurate comparison. The normalization factor is calculated as:

$$\alpha = 10^6 \times \frac{1}{N(\text{spike-in in ChIP})}$$

where α denotes the normalization factor, and $N(\text{spike-in in ChIP})$ represents the number of spike-in reads in the ChIP sample. Another normalization extending RRPM was proposed in the works of Niu *et al.* (7) and Fursova *et al.* (8). Rx-input normalization adjusts the scaling factors based on the ratio of spike-in reads relative to the total reads in corresponding input samples. The normalization factor is calculated with:

$$\alpha = 10^6 \times \frac{1}{N(\text{spike-in in ChIP})} \times \frac{N(\text{spike-in in Input})}{N(\text{reference in Input})}$$

We implemented two additional normalization methodologies. The downsampling normalization technique adjusts the sample read counts by reducing them to match the sample with the fewest spike-in reads. This method ensures uniformity by setting the reference sample as the one with the minimum reads, calculated as:

$$\alpha = \frac{\text{Min}(\text{all samples spike-in reads})}{N(\text{current sample reads})}$$

where α denotes the downsampling factor. On the other hand, the median normalization strategy scales all samples to the median value of the spike-in reads across the dataset. This method balances the read counts by adjusting each sample to the median, calculated as:

$$\alpha = \frac{\text{Median}(\text{all samples spike-in reads})}{N(\text{current sample reads})}$$

While downsampling reduces variability by equalizing to the smallest sample, median normalization balances the dataset to a central value.

Table 1. Table comparing the distinct features of the tools currently available for ChIP-Rx data analysis. *ChIPSeqSpike* is included even though it is not currently available on Bioconductor

	<i>ChIPSeqSpike</i>	<i>SpikChIP</i>	<i>DiffBind</i>	<i>Spiker</i>	<i>SpikeFlow</i>
Quality control	✗	✗	✓	✗	✓
Alignment and reads separation	✗	✗	✗	✓	✓
Narrow and broad peak calling	✗	✗	✗	✓	✓
Very-broad peak calling	✗	✗	✗	✗	✓
Signal tracks generation	✗	✗	✗	✓	✓
Automatic scaling factors calculation	✓	✓	✓	✗	✓
Preset normalization options	✗	✓	✓	✗	✓
Differential Peaks Analysis	✗	✗	✓	✗	✓
Peaks annotation	✗	✗	✗	✗	✓
Parallel execution	✗	✗	✓	✗	✓
Containerized workflow	✗	✗	✗	✗	✓
Results plotting	✓	✓	✓	✗	✓

In parallel with the bigWig file generation, SpikeFlow performs peak calling in narrow or broad standard modalities. Optionally, the users can also choose to perform normalized peak calling. Standard peak calling is achieved with *MACS2* (11) and *epic2* (23) for narrow and broad peaks, respectively. If normalized peak calling is selected, a function from *Spiker* (10) scales to spike-in reads both the sample and its matched input based on the chosen normalization type before performing peak calling with *MACS2*. For very-broad peaks, SpikeFlow performs standard peak calling with *EDD* (24), as *Spiker* handles only narrow and broad modes. The default parameters as the p-value thresholds for these tools are set, but the user can adjust them in the Snakemake configuration file. The obtained narrow and broad peaks are then annotated with the R package *ChIPseeker* (25). Differential peak analysis can be performed by specifying the contrasts in the workflow configuration file. This will create a consensus peak set for each antibody with *bedtools merge* (26). Next, the number of reads aligning on these regions is calculated, generating a consensus count matrix. Finally, a custom R script runs *DESeq2* (12), adjusting the size factors based on the specified normalization type (RPM, RRPm, Rx-Input, Downsampling, Median). Differential peaks are stored along with PCA and volcano plots. If only one sample per condition is available, the p-value can not be calculated, so SpikeFlow returns a log₂FC between single samples and generates a scatterplot. Finally, several checks are run by the pipeline to assess the quality of the experiments, namely *fastqc*, *phantompeakqualtools*, and *plotFingerprint* (22,27). Together with other QC plots, all the main outputs are listed in the report generated by *multiqc* (28). Details about the source code and the documentation to run SpikeFlow are in the data and code availability section. SpikeFlow results were compared to *DiffBind* v3.8.4 (14) and *SpikChIP* v1.0 (9). The *dba.normalize* function from *DiffBind* was run with ‘normalize=DBA_NORM_RLE, spikein = TRUE’ parameters. *SpikChIP* was run with ‘-p 2 -k 1000’. All the other parameters were left to default.

Results

To validate SpikeFlow, we selected two studies applying ChIP-Rx to quantify histone modifications or transcription factor binding in breast cancer and leukemia cells (Supplementary Tables S1 and S3). Other studies have used these datasets as a benchmark to observe the effects of spike-in normalization (9,14). In the first dataset, MV4-11 leukemia cells were treated

with increasing concentrations of EPZ5676, an inhibitor of DOT1L. This enzyme catalyzes the methylation of H3K79, a histone mark correlating with actively transcribed genes. A general reduction of H3K79me₂ on the gene bodies was observed by the authors at 5 nM of EPZ5676. As a control, they profiled H3K4me₃, which should not change upon EPZ5676 administration (6). We downloaded the fastq files of these experiments from the Gene Expression Omnibus database and analyzed them through SpikeFlow. On average, the samples had ≈14 million aligned reads, and the percentage of spike-in reads over the total ranged from 10% to 40% among the samples (Supplementary Table S2). For the differential peak analysis, we set the contrast to compare the 5 nM versus 0 nM samples and set RPM normalization (no spike-in). Since this dataset contains only one replicate per condition, SpikeFlow calculates a fold change and generates scatter plots but does not produce volcano plots due to the inability to perform the statistical test with *DESeq2*. With RPM normalization, we identified 238 regions exhibiting a log₂ fold change lower than -1.5 between the EPZ5676 (5 nM) and control conditions (Figure 1A). In contrast, applying Orlando (RRPM) normalization revealed 15 729 regions with decreased levels of H3K79me₂ (Figure 1B). As anticipated, no significant differences were observed for H3K4me₃ under either normalization method, verifying that H3K4me₃ levels remain unaffected by EPZ5676 treatment (Figure 1A, B). These analyses confirmed the published results, demonstrating that SpikeFlow could successfully identify the reduction in H3K79me₂ levels upon EPZ5676 treatment (Figure 1C).

After testing SpikeFlow on a dataset with one sample per biological conditions, we extended our analysis to an experiment with multiple replicates. We analyzed data from Guertin *et al.* (15), where they monitored estrogen receptor (ER) binding upon treatment with fulvestrant in MCF-7 cells. Fulvestrant functions as an ER antagonist, leading to cellular loss of ER protein and compromising ER binding genome-wide. This dataset serves as an ideal model to compare spike-in normalization methods. In this analysis, we ran SpikeFlow, which performed standard peak calling with *MACS2* for each replicate, merged the consensus peaks, and conducted differential peak analysis with Orlando (RRPM) normalization (Supplementary Table S4). The pipeline generated a volcano plot showing 4848 regions with a significant decrease in ER binding upon fulvestrant treatment (adj. P-value ≤ 0.05 and log₂FC ≤ -1, Figure 2 A). Along with the volcano, SpikeFlow produced a PCA plot where the four samples per condition clustered separately along the first principal component,

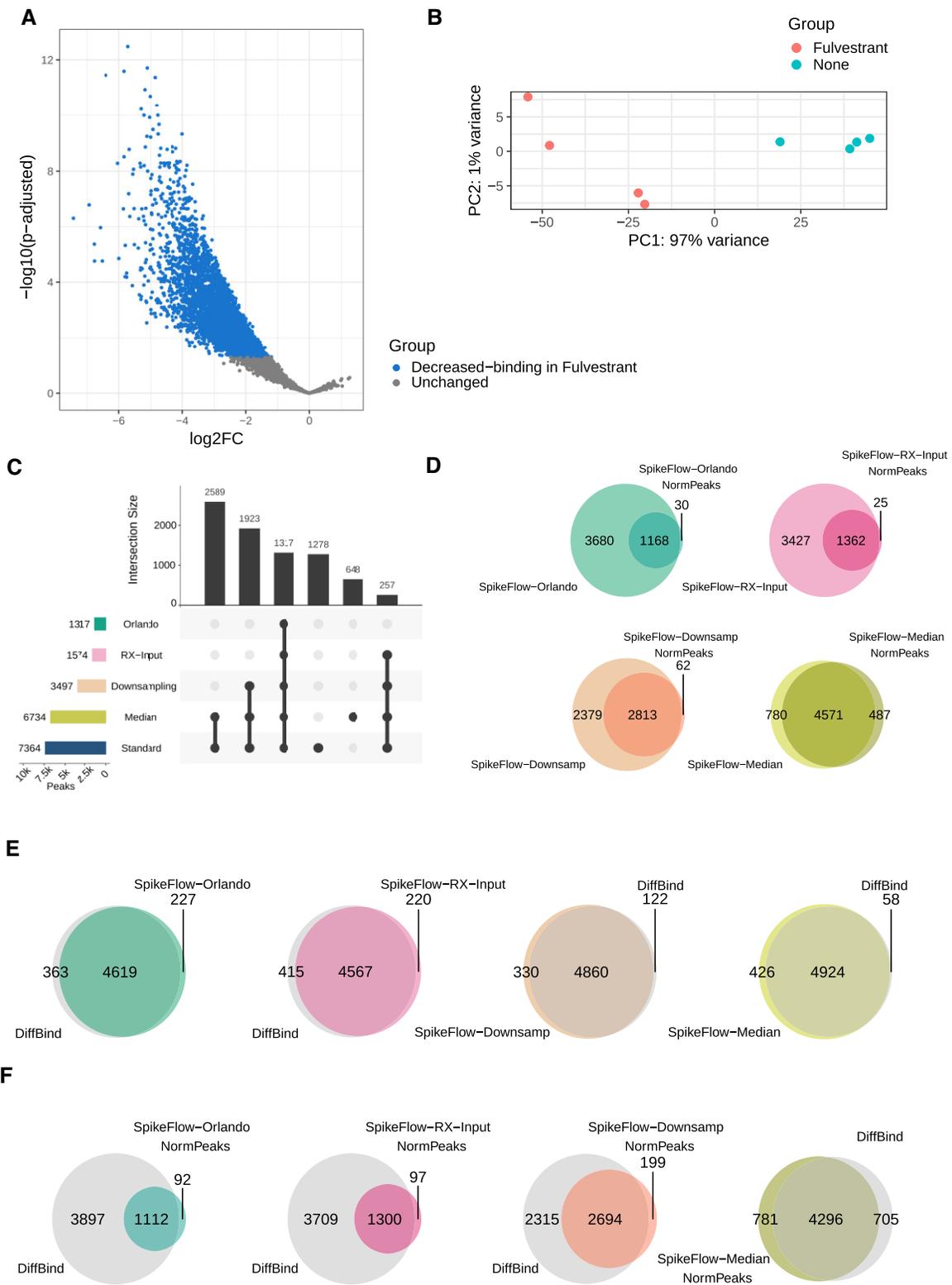


Figure 2. (A) Volcano plot generated by SpikeFlow by setting the Fulvestrant vs Control (None) contrast in the differential peak analysis. Blue dots are regions considered significantly decreased in ER binding (adj. P -value ≤ 0.05 and $\log_2FC \leq -1$). (B) PCA plot generated by SpikeFlow over the consensus regions obtained by merging the peaks of each replicate. (C) UpSet plot displaying the number of significant peaks (q -value ≤ 0.05) obtained with standard and normalized peak calling (Orlando, RX-Input, Downsampling, and Median). (D) Overlap of regions with a significant decrease in ER binding (adj. P -value ≤ 0.05 and $\log_2FC \leq -1$) between standard and normalized peak calling. (E) Overlap between DiffBind (gray) and SpikeFlow outputs of regions with a significant decrease in ER binding for standard peak calling. (F) As in E but with normalized peak calling.

indicating high similarity between replicates (Figure 2B). Our analysis successfully recapitulated the findings observed in the original paper, further validating SpikeFlow's effectiveness in handling datasets with multiple replicates per condition.

Along with the standard peak calling, which is performed on the raw BAM file, SpikeFlow allows the application of spike-in normalization factors while identifying the peaks (i.e. normalized peak calling). To test this feature, we compared the number of consensus peaks (q -value ≤ 0.05) from the samples in the Guertin dataset among all the types of spike-in normalization implemented in SpikeFlow (Orlando, Rx-input, Downsampling, and Median). We observed that the different types of spike-in normalizations reduced the number of significant peaks called compared to the standard method, and these regions were largely a subset of those identified by the standard peak calling (Figure 2C).

Next, we examined how the normalized peak calling strategy impacted the differential peak analysis. We computed the overlap of regions with a decrease in ER binding upon fulvestrant treatment between standard and normalized peak calling for each normalization type (Figure 2D). Although the peaks called with the normalized methods were fewer, they were more frequent among the regions with a significant decrease in the binding of ER. Specifically, 90% of the regions identified by Orlando normalized peak calling showed a significant decrease in ER binding, 88% for Rx-input, 82% for downsampling, and 75% for median normalization. To further investigate this, we examined the position of the regions identified with normalized peak calling in the volcano plot shown in Figure 2A. We observed that peaks identified using Orlando spike-in normalization were among those with the lowest adjusted P -values (Supplementary Figure S2). This suggests that normalized peak calling selects more conservative regions across all fold change levels. SpikeFlow provides researchers with a streamlined way to compare and evaluate different normalization methods, thereby aiding in selecting the most appropriate approach.

Additionally, we compared the differential peak regions identified by SpikeFlow with those identified by the published tool DiffBind (14), operating in spike-in mode. Using standard peak calling, we found that SpikeFlow median normalization identified an additional 426 regions with decreased ER binding that were not found by DiffBind (Figure 2E). Moreover, a high degree of overlap between the regions identified by SpikeFlow and DiffBind was observed, even when using normalized peak calling methods (Figure 2E, F). This consistency suggests that SpikeFlow is reliable and effective for differential peak analysis, regardless of the normalization approach.

To assess the effect of the different normalization strategies, we compared the normalized signals of ER binding over the consensus peaks regions in the Guertin dataset (Figure 3A). Median and downsampling normalizations exhibited greater variability between replicates than Orlando and Rx-input normalization. Still, the overall reduction of ER binding can be appreciated for all the normalization types. In this dataset, Rx-input is comparable to Orlando normalization since all the replicates share the same input. Next, we sought to compare the normalization strategies implemented in SpikeFlow with an already established tool, i.e. SpikChIP (9). SpikChIP computes Orlando, Downsampling, and local normalization approaches (referred as SpikChIP in the fig-

ures). The latter yielded a signal over peak regions comparable to the Orlando normalization (see Figure 3B). Orlando normalization implemented with SpikeFlow produced higher normalized signal values than SpikChIP, while Downsampling was similar (Figure 3B). Despite these differences, there was a significant correlation between the normalizations implemented in SpikeFlow and those in SpikChIP (Figure 3C, Supplementary Figure S3). The high correlation suggests SpikeFlow's normalization methods are robust and comparable to SpikChIP.

We further tested and validated SpikeFlow for the analysis of other histone modifications, both in vitro and in vivo (16,17,29), confirming its portability to diverse biological contexts. The reports generated by SpikeFlow for the MV4-11 and MCF-7 datasets are available in the supplementary material (Supplementary Files 1, 2).

Discussion

SpikeFlow is a new tool to perform ChIP-Rx data analysis, implementing diverse normalization strategies. It performs all the processing steps, from alignment to scaling factor calculations and differential peak analysis, making ChIP-Rx data processing accessible to users without specialized expertise.

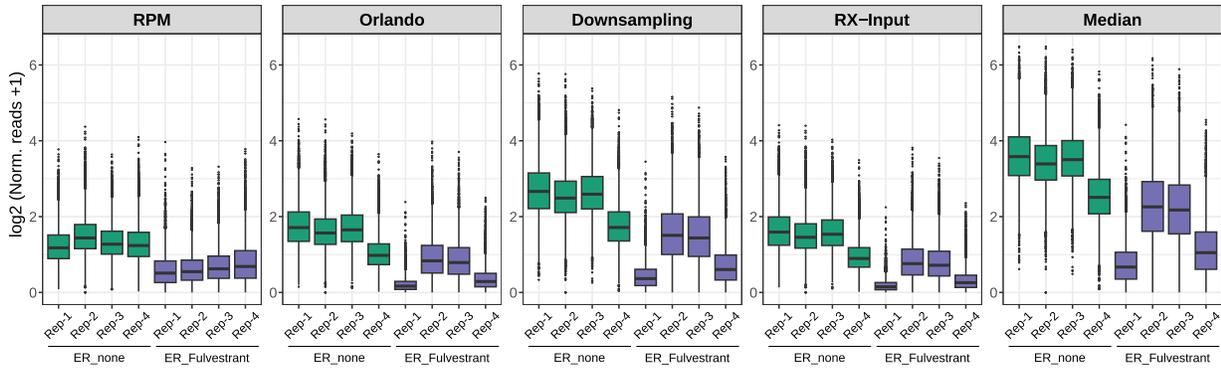
We demonstrated the effectiveness of SpikeFlow for differential peak analysis using datasets with single and multiple replicates per condition. Our findings show that SpikeFlow's ability to normalize sample signal before peak calling can provide meaningful advantages. Specifically, we observed that normalization methods, such as Orlando (RRPM) and Rx-input, tend to select more conservative regions with stronger contrasts, particularly useful for focused investigations. Although normalized peak calling is not widely adopted, our comparison with the standard method establishes a foundation for further analyses and comparisons across various datasets. We believe this work will allow future research to investigate the benefits and limitations of different normalization approaches. SpikeFlow simplifies this process, providing an efficient tool for exploring, validating, and applying various normalization techniques.

Additionally, our analysis matched the results observed in the work of Guertin *et al.* (15), where a reduction in ER binding upon fulvestrant treatment was observed, further validating SpikeFlow accuracy (Figure 2A). However, the choice of normalization type can heavily influence the outcome of the differential analysis and has to be taken into account while interpreting the results (Figure 2D).

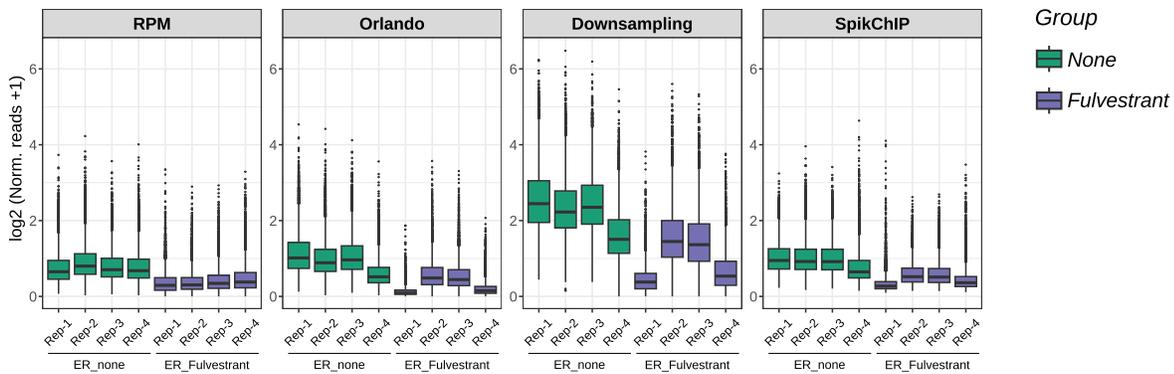
Comparative analysis with DiffBind (14) revealed that median normalization in SpikeFlow identified more significant regions with decreased ER binding, and there was a high overlap between the differential peak regions identified by both tools. Further, our examination of the Guertin dataset highlighted the differences in scaled signals introduced by changing normalization methods (Figure 3). Median and downsampling normalizations exhibited greater variability between replicates compared to Orlando and Rx-input normalizations. When compared to SpikChIP, we observed a high correlation between the normalization methods, underscoring the robustness of SpikeFlow's normalization techniques.

Overall, SpikeFlow offers comprehensive normalization options and the ability to perform complete analysis from

A SpikeFlow



B SpikChIP



C

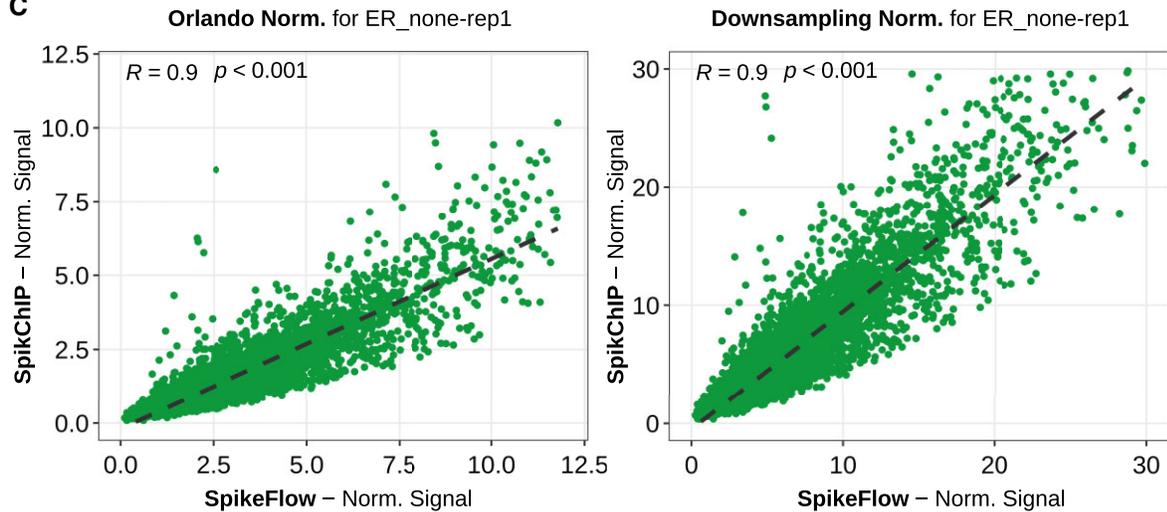


Figure 3. (A) Boxplots displaying the signal for each normalization applied by SpikeFlow and each replicate in the Guertin dataset (Supplementary Table S3) (15). The signal is calculated over the consensus regions, obtained with standard peak calling in SpikeFlow. **(B)** As in A, the normalized signals have been obtained by SpikChIP. **(C)** Scatter plot showing the signal correlation between SpikChIP and SpikeFlow for ER rep1 sample. The person correlation coefficient and the p-value are indicated at the panel's top.

raw data to downstream processing, including quality control and detailed plotting. Its support for parallel and containerized execution further enhances its utility, making it a valuable tool for ChIP-Rx data analysis. Finally, SpikeFlow allows researchers to achieve more precise and reliable results in histone modifications and transcription factor binding studies.

Data availability

SpikeFlow code and documentation are available on GitHub at <https://github.com/DavideBrex/SpikeFlow>. Moreover, SpikeFlow source code (version v1.2.1) was deposited on Zenodo (<https://zenodo.org/records/12189456>) for reproducibility. Optionally, SpikeFlow can be run in a containerized environment, so we also deposited the Docker image on Zenodo at <https://zenodo.org/records/11544955>. The GEO accession numbers for the analyzed datasets are available in [Supplementary Table S1](#) and [S3](#).

Supplementary data

[Supplementary Data](#) are available at NARGAB Online.

Acknowledgements

We thank Riccardo Scandino, Leonardo Morelli, and members of the Demichelis laboratory for the fruitful discussions and useful feedback.

Funding

The work of the Chiacchiera Laboratory is supported by the Italian Association for Cancer Research, AIRC [MFAG-20344]; Worldwide Cancer Research, WWCR [23-0321]; European Union under NextGenerationEU, PRIN 2022 [2022PWKZXE]; PRIN PNRR [P2022YP3HZ].

Conflict of interest statement

Fulvio Chiacchiera is a consultant for Dompé farmaceutici S.p.A.

References

- Egan,B., Yuan,C.-C., Craske,M.L., Labhart,P., Guler,G.D., Arnott,D., Maile,T.M., Busby,J., Henry,C., Kelly,T.K., *et al.* (2016) An alternative approach to chip-seq normalization enables detection of genome-wide changes in histone H3 Lysine 27 trimethylation upon EZH2 inhibition. *PLoS One*, **11**, e0166438.
- The ENCODE Project Consortium (2012) An integrated encyclopedia of DNA elements in the human genome. *Nature*, **489**, 57–74
- Furey,T.S. (2012) ChIP-seq and beyond: new and improved methodologies to detect and characterize protein–DNA interactions. *Nat. Rev. Genet.*, **13**, 840–852
- Park,P.J. (2009) ChIP-seq: advantages and challenges of a maturing technology. *Nat. Rev. Genet.*, **10**, 669–680
- Bonhoure,N., Bounova,G., Bernasconi,D., Praz,V., Lammers,F., Canella,D., Willis,I.M., Herr,W., Hernandez,N., Delorenzi,M., *et al.* (2014) Quantifying ChIP-seq data: a spiking method providing an internal reference for sample-to-sample normalization. *Genome Res.*, **24**, 1157–1168.
- Orlando,D.A., Chen,M.W., Brown,V.E., Solanki,S., Choi,Y.J., Olson,E.R., Fritz,C.C., Bradner,J.E. and Guenther,M.G. (2014) Quantitative ChIP-seq normalization reveals global modulation of the epigenome. *Cell Rep.*, **9**, 1163–1170.
- Niu,K., Liu,R. and Liu,N. (2018) Quantitative ChIP-seq by adding spike-in from another species. *Bio-Protocol*, **8**, e2981.
- Fursova,N.A., Blackledge,N.P., Nakayama,M., Ito,S., Koseki,Y., Farcas,A.M., King,H.W., Koseki,H. and Klose,R.J. (2019) Synergy between variant PRC1 complexes defines polycomb-mediated gene repression. *Mol. Cell*, **74**, 1020–1036.
- Blanco,E., Di Croce,L. and Aranda,S. (2021) SpikChIP: a novel computational methodology to compare multiple ChIP-seq using spike-in chromatin. *NAR Genom. Bioinform.*, **3**, lqab064.
- Wu,D., Wang,L. and Huang,H. (2021) Protocol to apply spike-in ChIP-seq to capture massive histone acetylation in human cells. *STAR Protoc.*, **2**, 100681.
- Zhang,Y., Liu,T., Meyer,C.A., Eeckhoutte,J., Johnson,D.S., Bernstein,B.E., Nusbaum,C., Myers,R.M., Brown,M., Li,W., *et al.* (2008) Model-based analysis of ChIP-Seq (MACS). *Genome Biol.*, **9**, R137.
- Love,M.I., Huber,W. and Anders,S. (2014) Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol.*, **15**, 550.
- Descostes,N., Tsigirgos,A. and Reinberg,D. (2018) ChIPSeqSpike: A R/Bioconductor Package for ChIP-Seq Data Scaling According to Spike-in Control. bioRxiv doi: <https://doi.org/10.1101/269118>, 22 February 2018, preprint: not peer reviewed.
- Stark,R. and Brown,G. (2011) DiffBind: Differential binding analysis of ChIP-Seq peak data. R package version 100 4.3.
- Guertin,M.J., Cullen,A.E., Markowitz,F. and Holding,A.N. (2018) Parallel factor ChIP provides essential internal control for quantitative differential ChIP-seq. *Nucleic Acids Res.*, **46**, e75.
- Pathania,M., De Jay,N., Maestro,N., Harutyunyan,A.S., Nitarska,J., Pahlavan,P., Henderson,S., Mikael,L.G., Richard-Londt,A., Zhang,Y., *et al.* (2017) H3.3K27M cooperates with Trp53 loss and PDGFRA gain in mouse embryonic neural progenitor cells to induce invasive high-grade gliomas. *Cancer cell*, **32**, 684–700.
- D’Ambrosio,A., Bressan,D., Ferracci,E., Carbone,F., Mulè,P., Rossi,F., Barbieri,C., Sorrenti,E., Fiaccadori,G., Detone,T., *et al.* (2024) Increased genomic instability and reshaping of tissue microenvironment underlie oncogenic properties of Arid1a mutations. *Sci. Adv.*, **10**, eadh4435.
- Mölder,F., Jablonski,K.P., Letcher,B., Hall,M.B., Tomkins-Tinch,C.H., Sochat,V., Forster,J., Lee,S., Twardziok,S.O., Kanitz,A., *et al.* (2021) Sustainable data analysis with Snakemake. Technical Report 10:33, F1000Research.
- Chen,S., Zhou,Y., Chen,Y. and Gu,J. (2018) fastp: an ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics*, **34**, i884–i890.
- Langmead,B. and Salzberg,S.L. (2012) Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.
- Faust,G.G. and Hall,I.M. (2014) SAMBLASTER: fast duplicate marking and structural variant read extraction. *Bioinformatics*, **30**, 2503–2505.
- Ramírez,F., Dündar,F., Diehl,S., Grüning,B.A. and Manke,T. (2014) deepTools: a flexible platform for exploring deep-sequencing data. *Nucleic Acids Res.*, **42**, W187
- Stovner,E.B. and Sætrom,P. (2019) epic2 efficiently finds diffuse domains in ChIP-seq data. *Bioinformatics*, **35**, 4392–4393.
- Lund,E., Oldenburg,A.R. and Collas,P. (2014) Enriched domain detector: a program for detection of wide genomic enrichment domains robust against local variations. *Nucleic Acids Res.*, **42**, e92.
- Yu,G., Wang,L.-G. and He,Q.-Y. (2015) ChIPseeker: an R/Bioconductor package for ChIP peak annotation, comparison and visualization. *Bioinformatics*, **31**, 2382–2383.

26. Quinlan, A.R. and Hall, I.M. (2010) BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, **26**, 841–842.
27. Landt, S.G., Marinov, G.K., Kundaje, A., Kheradpour, P., Pauli, F., Batzoglou, S., Bernstein, B.E., Bickel, P., Brown, J.B., Cayting, P., *et al.* (2012) ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Res.*, **22**, 1813–1831.
28. Ewels, P., Magnusson, M., Lundin, S. and Käller, M. (2016) MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*, **32**, 3047–3048.
29. Müller, S., Sindikubwabo, F., Cañeque, T., Lafon, A., Versini, A., Lombard, B., Loew, D., Wu, T.-D., Ginestier, C., Charafe-Jauffret, E., *et al.* (2020) CD44 regulates epigenetic plasticity by mediating iron endocytosis. *Nat. Chem.*, **12**, 929–938.