BIOINFORMATICS

Vol. 00 no. 00 2015
Pages 1–6

Zerone: a ChIP-seq discretizer for multiple replicates with built-in quality control

Pol Cuscó [1,2] and Guillaume Filion [1,2*]

[1]Genome Architecture, Gene Regulation, Stem Cells and Cancer Programme, Centre for Genomic Regulation (CRG), The Barcelona Institute of Science and Technology, Dr. Aiguader 88, Barcelona 08003, Spain.
[2]Universitat Pompeu Fabra (UPF), Barcelona, Spain.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: Chromatin immunoprecipitation followed by high-throughput sequencing is the standard method to investigate chromatin protein composition. As the number of community-available ChIP-seq profiles increases, it becomes more common to use data from different sources, which makes joint analysis challenging. Issues such as lack of reproducibility, heterogeneous quality and conflicts between replicates become evident when comparing datasets, especially when they are produced by different laboratories.

Results: Here we present Zerone, a ChIP-seq discretizer with built-in quality control. Zerone is powered by a Hidden Markov Model with zero-inflated negative multinomial emissions, which allows it to merge several replicates into a single discretized profile. To identify low quality or irreproducible data, we trained a Support Vector Machine and integrated it as part of the discretization process. Zerone identified low quality data with 95% accuracy. In terms of performance, Zerone is more than 4 times faster than MACS for a similar accuracy.

Availability: Zerone is available as a command line tool and as an R package. The C source code and R scripts can be downloaded from https://github.com/gui11aume/jahmm.

Contact: guillaume.filion@gmail.com

1 INTRODUCTION

One of the major challenges of biology is to understand how transcription factors and chromatin proteins coordinate genome-dependent processes such as transcription, replication and repair. Massive research efforts are invested into collecting protein-genome interaction data in order to gain insight into the organization of the genome as a whole. Chromatin immunoprecipitation followed by high throughput sequencing (ChIP-seq) emerged as the standard method to identify the targets of a transcription factor or a histone modification in a cell population. However, ChIP is not fully understood and artifacts are still discovered more than 10 years after its adoption (Park *et al.*, 2013; Teytelman *et al.*, 2013). Besides, the constant improvement of sequencing technologies makes analysis of ChIP-seq profiles difficult to standardize. There is thus a continuous need to develop and improve computational tools to analyse ChIP-seq data.

One of the most common analyses performed on ChIP-seq profiles is to discretize the signal, *i.e.* make calls whether the feature is present or absent for every *locus* of the genome. This may seem dubious at first glance because the biological reality is intrinsically quantitative, but there are several good reasons to discretize ChIP-seq profiles: it makes the signal simpler to interpret from the human perspective, it simplifies downstream analyses and it allows to compare or combine profiles of different nature. This raises a challenge at the computational level because discretization has to be carried out uniformly for signals that may have very different properties. For instance compare the megabase scale domains of Lamin (Guelen *et al.*, 2008) to the 6 bp binding sites of typical transcription factors.

Large consortia such as ENCODE have brought to light a more severe type of issue related to the quality of ChIP-seq data. Conflicts between replicates are common, and sometimes laboratory effects are clearly detectable in the data, even when experimentalists use the same material and follow the same protocol (our unpublished observations). The most popular remedy is to use a metric called IDR (Irreproducible Discovery Rate, Li *et al.*, 2011), which allows to weed out poorly reproducible signal. This approach is a significant step forward, but the IDR is undefined when more than two replicates are available. Besides, keeping only the reproducible ChIP peaks is not always the best option. If one of the replicates is mislabelled, for instance, it is more appropriate to reject the dataset than to keep the common ChIP peaks. In summary, how to integrate ChIP-seq data from different sources and with variable qualities is still an open problem.

Here we propose an approach to discretize ChIP-seq data where conflict resolution and quality control are integrated in a tool that we called Zerone. The key idea of Zerone is to combine an arbitrary number of ChIP-seq replicates in a single discretized profile, where conflicts are resolved by maximizing the likelihood of the underlying statistical model. Following discretization, Zerone controls the quality of its output in order to detect potential anomalies, and when applicable rejects the output as a whole. Internally, the first step implements a Hidden Markov Model (HMM) with zero-inflated negative multinomial (ZINM) emissions, and the second implements a Support Vector Machine (SVM)

*to whom correspondence should be addressed

© Oxford University Press 2015.

trained using ENCODE ChIP-seq data. HMM-based discretization is agnostic to the shape of the signal (broad or peaky) and the ZINM distribution captures the essential features of the read count distribution in ChIP-seq data. These two properties provide a unified framework to discretize ChIP-seq data of different kinds.

Zerone is designed for large volume pipelines aiming to combine many ChIP-seq profiles with little human intervention. To this end, it is compatible with the standard SAM and BAM formats (Li *et al.*, 2009), it produces congruent window-based outputs, and it can process hundreds of experiments per day on average hardware. We benchmarked Zerone against MACS (Zhang *et al.*, 2008), BayesPeak (Spyrou *et al.*, 2009) and JAMM (Ibrahim *et al.*, 2015) on the core task of discretizing ChIP-seq profiles of CTCF, H3K36me3 and Pol2. Our results show that Zerone is competitive in terms of speed and accuracy.

2 METHODS

2.1 Emission model

It is natural to model read counts in genomic windows by an unbounded discrete distribution. The Poisson distribution is an obvious candidate, but it is a poor choice because the variance of read counts is typically higher than the mean in ChIP-seq data. The reason is that windows are non homogeneous, which increases the dispersion. More specifically, windows are not equally PCR-prone and not equally mappable. The negative binomial (NB) distribution is thus a better choice because it allows some variation between windows. However, genomes are fraught with repeats, which creates an excess of windows where reads cannot be mapped. Since such windows will always have 0 read count, a natural choice for this distribution is the zero-inflated negative binomial (ZINB), *i.e.* the mixture of a negative binomial distribution and a distribution concentrated at 0.

The ZINB distribution has 3 parameters that can be fitted by maximum likelihood. Zerone uses a custom solver based on the Newton-Raphson method, which converges much faster than the popular routine `zeroinfl` (Zeileis *et al.*, 2008) from the R (R Core Team, 2014) package `pscl` (Jackman, 2015). Fig. 1 shows that the ZINB distribution gives a better fit to ChIP-seq data than Poisson and NB distributions.

The NB distribution can be interpreted as a Gamma-Poisson process, which gives a straightforward extension to a multivariate distribution called the Negative Multinomial (NM) and to its corresponding zero-inflated version the Zero-Inflated Negative Multinomial (ZINM, see supplementary material for detail). In this model, windows have an intrinsic ChIP-seq bias due to their sequence composition, mappability and other inherent properties, which gives a baseline variation present in all ChIP-seq experiments performed in the same conditions. All the replicates of a ChIP-seq experiment can thus be combined with the negative controls in a single multivariate distribution.

2.2 Discretization

Discretization is performed by fitting an HMM with ZINM emissions (see section 2.1). The HMM has three states corresponding to "low", "medium" and "high" abundance of the given chromatin feature. We have observed that in many ChIP-seq profiles, the baseline signal shows block-wise variations of low amplitude but large size (typically 10-100 Kb). This will sometimes be the dominant signal and a two-state HMM will identify these blocks instead of the targets. Dedicating two states to fit the baseline is a way to make sure that the "high" state corresponds to the targets of the chromatin feature.

Fitting is performed with the Baum-Welch algorithm (Baum and Petrie, 1966), which is a special case of EM algorithm (Dempster *et al.*, 1977). Discrete variables take only a small number of distinct values, which allows to save computation time by hashing the observations. With this technique, we need to compute each value of the emission probabilities only once per cycle of the Baum-Welch algorithm. Transition parameters are updated through the forward-backward algorithm, and emission parameters are updated by solving maximum likelihood equations directly with the Newton-Raphson method (see supplementary material for detail). Approximately 3/4 of the computation time is spent in foward-backward cycles, and 1/4 in updating emission parameters (the time spent computing emission probabilities is insignificant). The algorithm stops when parameters reach a stable value, or after a limit number of cycles (100 by default). The state calls are computed by finding the most likely segmentation given the value of the parameters through the Viterbi algorithm (Viterbi, 1967).

The shape parameter and the mixture ratio of the ZINM distribution are fitted directly from the negative control profiles and they are considered constant throughout. Overall, the total number of estimated parameters is $3(r+1)$, where r is the number of replicate experiments (excluding negative controls).

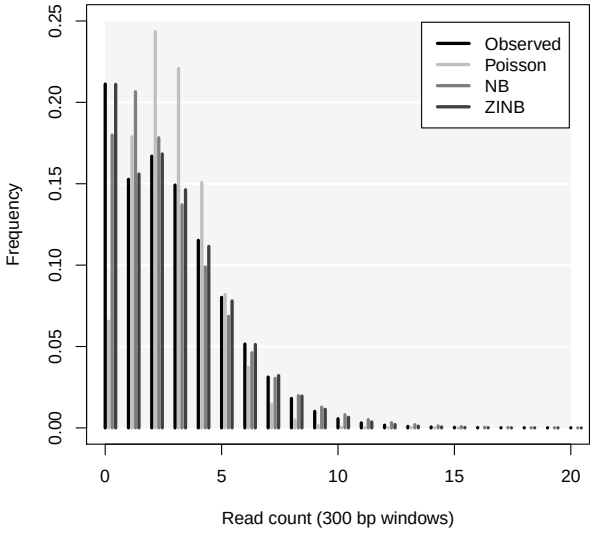

Fig. 1. Using the ZINB distribution to model ChIP-seq data. Reads from the negative control dataset XX were mapped on the human genome and pooled in 300 bp windows after removing duplicates. The histogram of the read counts is shown in black (no immunoprecipitation was performed in this experiment, so this variation corresponds to the 'baseline'). The histograms in gray scales show the maximum likelihood fit of the Poisson, Negative Binomial (NB) and Zero-Inflated Negative Binomial (ZINB) distributions. The fit of the Poisson distribution (light gray) is poor. The NB distribution (medium gray) gives a good fit at the tail, but not for windows with 0 and 1 read. The ZINB distribution (dark gray) gives a good fit over the whole range.

2.3 Classification and training

We used a machine learning strategy to identify discretization failures. We first prepared a high confidence dataset where the output of Zerone was labelled positive (success) or negative (failure). We discretized 144 replicated ChIP-seq experiments, together with their respective input control. We labelled the output for the discretization as positive (91 cases) or negative (53 cases), based visual inspection and on the available literature about the chromatin features. The most common cases of poor data quality in

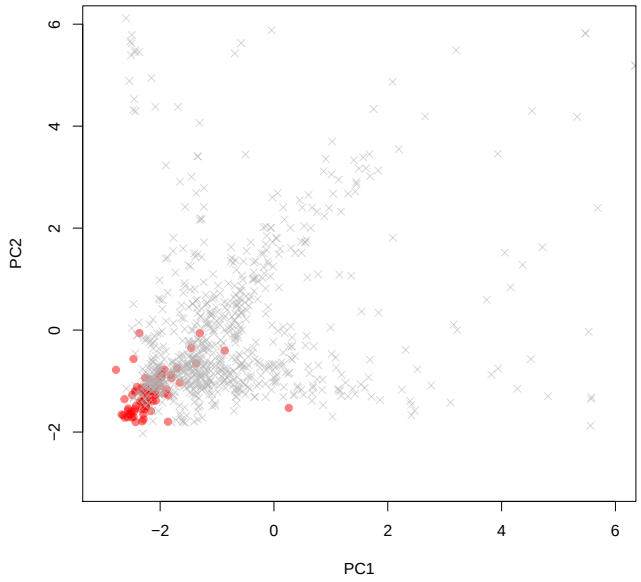

Fig. 2. Principal Component Analysis of the training dataset. Training examples labeled as positive (red circles) appear to be similar to each other, while negative examples (grey crosses) show notable differences between them and with respect to positive examples. There exist though a certain degree of overlap between the two groups that creates an "ambiguous zone", at least on this projection.

ChIP-seq correspond to low signal-to-noise ratio (*e.g.* when the antibody is unspecific), and lack of repdroducibility between replicates (*e.g.* when samples are swapped). To cover these cases, we included in the trusted set 720 cases obtained by discretizing non-replicates (*e.g.* CTCF and Pol2) and controls without immuno-precipitation.

To identify discretization failures, we used the paremeters of the fitted HMM. Based on the transition matrix, the emission paramaters, the Viterbi Path and the posterior probabilities, we computed 19 features expected to vary with the quality of the discretization, and we used them to train a classifier.

Our first attempts with logistic regression suggested that linear classifiers are unable to properly separate these two classes because they overlap in the feature space (Fig. 2). To obtain more complex, nonlinear separation, we used a Support Vector Machine (SVM, Chang and Lin, 2011; Meyer *et al.*, 2014), as this approach guarantees the lowest overfitting upper bound and allows nonlinear classification by mapping the training data to a kernel space. Also, SVMs are fast to train and they require only one hyperparameter to be fitted, which were advantages over other approaches such as neural networks.

We trained the SVM with a radial basis function kernel and selected the hyperparameters that maximized the prediction performance on test sets using a 10-fold cross-validation scheme. The prediction accuracy on the trusted set was 95%. We then implemented a prediction function in Zerone that used the model trained by the SVM to classify the discretizations and suggest whether they should be accepted or discarded.

2.4 Datasets and preprocessing

We used all the ChIP-seq profiles produced by the ENCODE consortium on the human myelogenous leukemia cell line K562. We did not make use of preprocessed mapped reads because they could have been mapped with different software and could therefore disagree on its results (Szalkowski and Schmid, 2011). Instead, we mapped all the raw reads onto the hg19 assembly of the human genome with GEM (Marco-Sola *et al.*, 2012), using the options `--unique-mapping` and `-q ignore` of gem-mapper version 1.376 (beta). The version of gem-indexer was 1.423 (beta). We preprocessed the data in the same way both for training the classifier and for benchmarking. We also used the same genome assembly to generate the datasets used in the benchmarks (see Section 3).

For training the classifier, we binned the mapped reads into 300 bp windows and saved the data in WIG format before discretization.

2.5 Benchmark conditions

To compare Zerone to other discretizers, we analysed three different ChIP-seq datasets: CCCTC-binding factor (CTCF), tri-methylated histone H3 at lysine 36 (H3K36me3) and RNA polymerase II (Pol2)—that represent punctate, broad and mixed type signals respectively. Each dataset consisted of an input profile and two replicate target profiles.

To make the comparisons fair, we merged all contiguous windows that were called as enriched by Zerone, in the same way the other programs do. Otherwise, the number of enriched regions in the genome would be higher and Zerone performance would look poorer than the actual.

We decided to include MACS `callpeak` (version 2.1.0.20140616) in the comparisons because it is the *de facto* standard method for ChIP-seq peak calling; BayesPeak (version 1.20.0) because, as Zerone, it makes use of an HMM with ZINB emissions to estimate the regions of enrichment; and JAMM (version 1.0.7rev1) because it can perform joint analysis on experimental replicates.

All tests were performed on an 8-core Intel Xeon E5606 machine with 48 GB of DDR3-RAM at 1333 MHz. All programs were run on a single core with the default options.

3 RESULTS

3.1 Speed and memory consumption

We compared the running times of the different programs on discretizing three datasets of similar size that represent the three major types of ChIP-seq signal usually observed. The CTCF signal consists of sharp peaks at the transcription factor binding site, the H3K36me3 signal consists of broad domains, and the Pol2 signal consists of peaks at promoters and potentially broad domains on transcribed genes.

The results were similar between experiments, and Zerone was consistenty the fastest tool, with a running time around 5 minutes (Fig. 3, top row). The advantage is only marginal over MACS, wihch ran for around 10 minutes, but it is substantial over BayesPeak and JAMM, which ran in over 9 hours. The results for peak memory usage were more variable between experiments (Fig. 3, bottom row). MACS achieved the best performance with a memory footprint around 0.5 GB, followed by Zerone around 1.0 GB. BayesPeak and JAMM each used more than 1.5 GB.

The benchmark is partly confounded by the fact that BayesPeak and MACS discretize a single input per run, whereas Zerone and JAMM discretize multiple inputs simultaneously. This makes a difference for pipelines where all files have to be processed in parallel with the minimum amount of resources. In our benchmark, Zerone used twice as much memory as MACS, but it also processed twice as many files. In other words, for the same amount of available memory, a Zerone pipeline would run twice faster than a MACS pipeline, and with only half the required amount of CPUs.

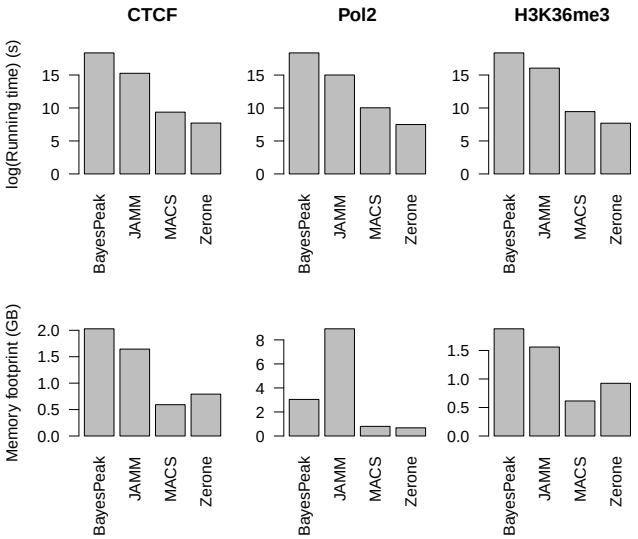

Fig. 3. Running times and peak memory footprint of the discretizers on the three ChIP-seq datasets. For programs that only allow single-profile discretization (*i.e.* BayesPeak and MACS), mean values (not the sum) are shown. Note the logarithmic scale in the running times.

3.2 Discretization benchmark

The purpose of discretization is to simplify and denoise ChIP-seq profiles. The process can be viewed as a lossy compression, a noise reduction or a classification. Intuitively, good discretizers capture a large fraction of the ChIP-seq signal with few target genomic sites. The number of targets and the amount of signal captured are therefore critical characteristics of a discretizer.

Unfortunately, the amount of noise in each experiment is unknown (because of differences between antibody specificities), so there is a large uncertainty about the amount of signal lost during the discretization. As a result, there is no gold standard to estimate the trade-off between false positives and false negatives in ChIP-seq experiments, and thus no gold standard to rank discretizers. However, we can compare discretizers with a partial order, as explained below.

When sorting genomic windows by decreasing order of ChIP-seq reads and plotting the cumulative number of reads, we obtain a curve that forms a Pareto front. It represents the largest amount of reads that can be captured by the given amount of targets, or alternatively the smallest number of targets that can catpure the given amount of reads. A discretization can be represented as a point on the xy plane, the x coordinates is the number of targets, and the y coordinate is the number of reads captured by the discretization.

By construction, no discretization can lie on the upper left side of the Pareto front, and every discretization on the front represents an optimum. Discretizations lying on the lower right side of the front are suboptimal, since the same number of targets can capture more reads, and the the same number of reads can be captured by less targets. With this representation, the quality of a discretization can be appreciated by its distance to the Pareto front.

In spite of the appearances, this representation has little to do with receiver operating characteristic (ROC) curves. First, a discretizer is represented as a point and not a curve. Second, a discretization close to the top-left corner is not always better than one that lays further away. For instance, if the experiment consists of pure noise, the Pareto front is a straight line from the bottom left to the top right corner. On this line, the closest point to the top-left corner is not a better discretization. Comparing two discretizations on the Pareto front is thus a delicate matter. All that can be said is that discretizations are expected to have more false positives and less false negatives as the slide to the right.

The discretizers are compared in Fig. 4. Whereas the discretizations of CTCF have similar characteristics (left panel), those of Pol2 and H3K36me3 differ substantially (middle and right panels).

Table 1. Summary of peak calling on the CTCF dataset. The table lists the total number of peaks found by the different programs, how many of those peaks contained at least one CTCF motif, and the correspondent precision, recall and F_1 score relative to the CTCF motif dataset.

Software	Total	Motif	Precision	Recall	F_1 score
BayesPeak$^{(1)}$	45,316	25,228	0.56	0.29	0.39
BayesPeak$^{(2)}$	45,154	23,428	0.52	0.27	0.36
JAMM	264,410	31,709	0.12	0.37	0.18
MACS$^{(1)}$	48,358	26,449	0.55	0.31	0.39
MACS$^{(2)}$	41,030	23,542	0.57	0.27	0.37
Zerone	50,792	30,972	0.61	0.36	0.45

The numbers in parentheses indicate the results on the two replicates by separate.

3.2.1 Identification of CTCF binding sites. CTCF binding sites are characterized by a specific 20 bp sequence that is highly conserved in vertebrates. In humans, nearly 80% of these sites contain the consensus motif (Kim *et al.*, 2007). In order to determine the capacity of the different programs to call peaks of CTCF binding, we compared the discretized profiles against a dataset of CTCF binding motifs. We used FIMO (Grant *et al.*, 2011) from the MEME suite version 4.10.1 (Bailey *et al.*, 2009), to generate such dataset from the human CTCF motif obtained from the JASPAR database version 5.0_ALPHA (Mathelier *et al.*, 2014). The CTCF motif dataset contained 85,690 entries.

Table 1 shows that Zerone outperforms its competitors: it has one of the highest recall scores, comparable to that of JAMM, the other software capable of combined analysis, while achieving the highest precision among the evaluated programs.

3.2.2 H3K36me3-enriched domains. Unlike in the previous case, there is no known consensus sequence to determine the location of histone modifications. However, it is known that the bodies of active genes are enriched in H3K36me3 (Pokholok *et al.*, 2005; Kimura, 2013). Therefore, the genes that contain peaks or windows determined as enriched in H3K36me3 by the different discretizers should be more expressed than the background.

UPDATE shows that the expression levels of regions detected as enriched in H3K36me3 by Zerone are indeed higher than the ones of non-enriched regions. Zerone detects as not enriched windows with lower expression than other programs (Fig. **??**, left), consistent with having the highest precision score as discussed above (Table 1).

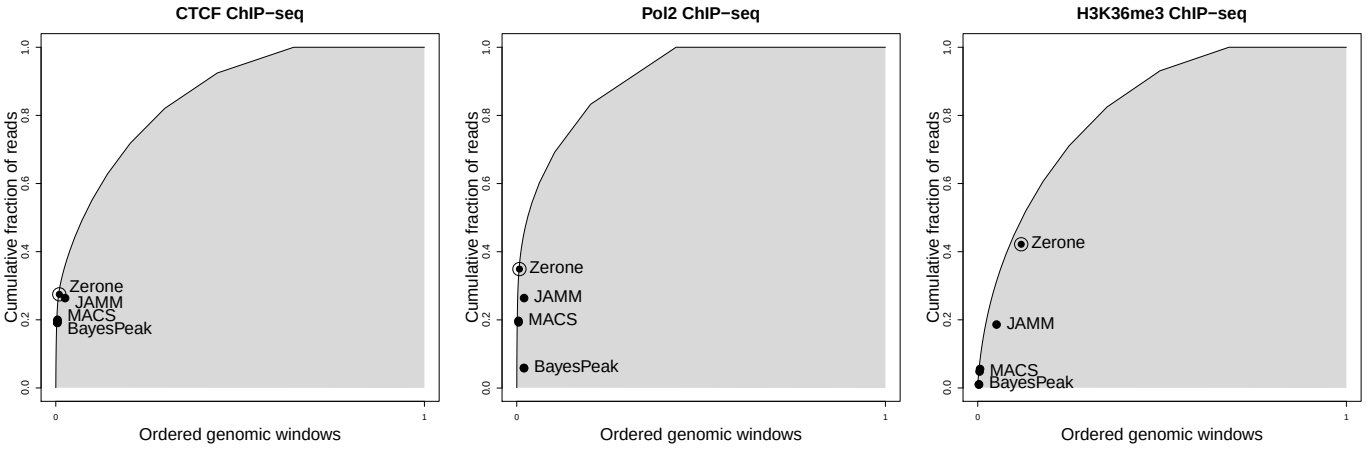

Fig. 4. Left: Update legend.

Table 2. Summary of peak calling on the Pol2 dataset. As in Table 1, The table lists the total number of peaks found by the different programs, how many of those peaks contained at least one TSS, and the correspondent precision, recall and F_1 score relative to the Pol2 dataset.

Software	Total	TSS	Precision	Recall	F_1 score
BayesPeak$^{(1)}$	208,809	2,722	0.01	0.06	0.03
BayesPeak$^{(2)}$	203,438	2,603	0.01	0.05	0.03
JAMM	209,882	6,761	0.03	0.21	0.11
MACS$^{(1)}$	51,313	6,845	0.13	0.22	0.27
MACS$^{(2)}$	49,034	6,546	0.13	0.21	0.26
Zerone	23,976	6,926	0.29	0.24	0.36

The numbers in parentheses indicate the results on the two replicates by separate.

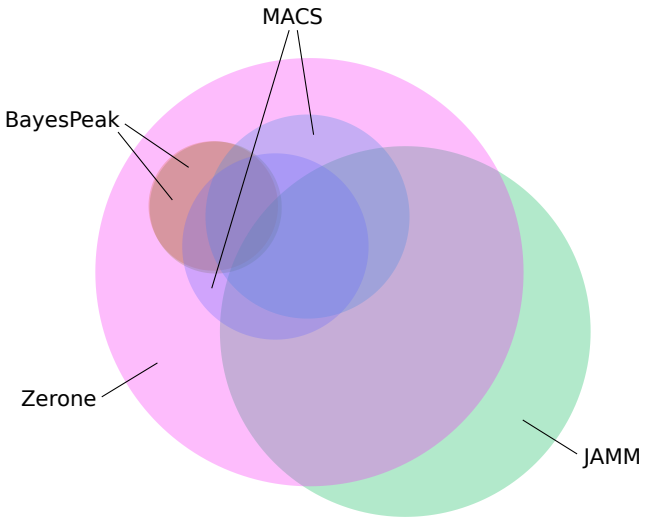

Fig. 5. Enriched genomic windows called by the different software. Circle size represents the enriched portion of the genome according to each of the programs, and the overlap between circles approximates the amount of windows found enriched by two or more programs. Note the almost complete overlap between the BayesPeak discretizations on the two replicates.

However, both JAMM and Zerone appear to detect as enriched windows with a lower mean read count (Fig. **??**, center), which seems to contradict the previous finding that these discretizers have the highest recall scores when analyzing the CTCF dataset (Table 1). This effect can be explained by the fact that Zerone is able to detect more enriched windows: other programs are only able to detect the peaks with the highest intensity and therefore the mean read count in enriched windows could be overestimated (Fig. **??**, right).

Interestingly, the majority of the enriched windows are detected by most programs. This can be visualized in the Venn diagram of Fig. 5, where Zerone is shown to be able to detect most of the windows detected by other software, while discovering new enriched windows not found by the others, which is consistent with the previous observations (Table 1 and Fig. **??**)

3.2.3 Pol2 binding around transcription start sites. To compare the behavior of the different discretizers, we determined how the peaks were distributed around TSSs.

4 DISCUSSION AND CONCLUSIONS

ACKNOWLEDGEMENT

Funding: P.C. fellowship is partly financed by the Spanish Ministry of Economy and Competitiveness (State Training Subprogram: predoctoral fellowships for the training of PhD students (FPI) 2013). We acknowledge support of the Spanish Ministry of Economy and

Competitiveness, 'Centro de Excelencia Severo Ochoa 2013-2017', SEV-2012-0208.

REFERENCES

Bailey, T. L., Boden, M., Buske, F. A., Frith, M., Grant, C. E., Clementi, L., Ren, J., Li, W. W., and Noble, W. S. (2009). MEME SUITE: tools for motif discovery and searching. *Nucleic Acids Res.*, **37**(Web Server issue), W202–8.

Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics*, **37**(6), 1554–1563.

Chang, C.-c. and Lin, C.-j. (2011). LIBSVM : A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, **2**, 1–39.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, **39**(1), 1–38.

Grant, C. E., Bailey, T. L., and Noble, W. S. (2011). FIMO: scanning for occurrences of a given motif. *Bioinformatics*, **27**(7), 1017–8.

Guelen, L., Pagie, L., Brasset, E., Meuleman, W., Faza, M. B., Talhout, W., Eussen, B. H., de Klein, A., Wessels, L., de Laat, W., and van Steensel, B. (2008). Domain organization of human chromosomes revealed by mapping of nuclear lamina interactions. *Nature*, **453**(7197), 948–51.

Ibrahim, M. M., Lacadie, S. A., and Ohler, U. (2015). JAMM: a peak finder for joint analysis of NGS replicates. *Bioinformatics*, **31**(1), 48–55.

Jackman, S. (2015). *pscl: Classes and Methods for R Developed in the Political Science Computational Laboratory, Stanford University*. Department of Political Science, Stanford University, Stanford, California. R package version 1.4.9.

Kim, T. H., Abdullaev, Z. K., Smith, A. D., Ching, K. A., Loukinov, D. I., Green, R. D., Zhang, M. Q., Lobanenkov, V. V., and Ren, B. (2007). Analysis of the vertebrate insulator protein CTCF-binding sites in the human genome. *Cell*, **128**(6), 1231–45.

Kimura, H. (2013). Histone modifications for human epigenome analysis. *J. Hum. Genet.*, **58**(7), 439–45.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and and, R. D. (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, **25**(16), 2078–9.

Li, Q., Brown, J. B., Huang, H., and Bickel, P. J. (2011). Measuring reproducibility of high-throughput experiments. *Ann. Appl. Stat.*, **5**(3), 1752–1779.

Marco-Sola, S., Sammeth, M., Guig, R., and Ribeca, P. (2012). The GEM mapper: fast, accurate and versatile alignment by filtration. *Nat. Methods*, **9**(12), 1185–8.

Mathelier, A., Zhao, X., Zhang, A. W., Parcy, F., Worsley-Hunt, R., Arenillas, D. J., Buchman, S., yu Chen, C., Chou, A., Ienasescu, H., Lim, J., Shyr, C., Tan, G., Zhou, M., Lenhard, B., Sandelin, A., and Wasserman, W. W. (2014). JASPAR 2014: an extensively expanded and updated open-access database of transcription factor binding profiles. *Nucleic Acids Res.*, **42**(Database issue), D142–7.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., and Leisch, F. (2014). *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*. R package version 1.6-4.

Park, D., Lee, Y., Bhupindersingh, G., and Iyer, V. R. (2013). Widespread misinterpretable ChIP-seq bias in yeast. *PLoS ONE*, **8**(12), e83506.

Pokholok, D. K., Harbison, C. T., Levine, S., Cole, M., Hannett, N. M., Lee, T. I., Bell, G. W., Walker, K., Rolfe, P. A., Herbolsheimer, E., Zeitlinger, J., Lewitter, F., Gifford, D. K., and Young, R. A. (2005). Genome-wide map of nucleosome acetylation and methylation in yeast. *Cell*, **122**(4), 517–27.

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Spyrou, C., Stark, R., Lynch, A. G., and Tavar, S. (2009). BayesPeak: Bayesian analysis of ChIP-seq data. *BMC Bioinformatics*, **10**, 299.

Szalkowski, A. M. and Schmid, C. D. (2011). Rapid innovation in ChIP-seq peak-calling algorithms is outdistancing benchmarking efforts. *Brief. Bioinformatics*, **12**(6), 626–33.

Teytelman, L., Thurtle, D. M., Rine, J., and van Oudenaarden, A. (2013). Highly expressed loci are vulnerable to misleading ChIP localization of multiple unrelated proteins. *Proc. Natl. Acad. Sci. U.S.A.*, **110**(46), 18602–7.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, **13**(2), 260–269.

Zeileis, A., Kleiber, C., and Jackman, S. (2008). Regression models for count data in R. *Journal of Statistical Software*, **27**(8).

Zhang, Y., Liu, T., Meyer, C. A., Eeckhoute, J., Johnson, D. S., Bernstein, B. E., Nusbaum, C., Myers, R. M., Brown, M., Li, W., and Liu, X. S. (2008). Model-based analysis of ChIP-Seq (MACS). *Genome Biol.*, **9**(9), R137.