Accounting for GC-content bias reduces systematic errors and batch effects in ChIP-seq data

Mingxiang Teng and Rafael A. Irizarry

Genome Res. 2017 27: 1930-1938 originally published online October 12, 2017
Access the most recent version at doi:10.1101/gr.220673.117

Method

Accounting for GC-content bias reduces systematic errors and batch effects in ChIP-seq data

Mingxiang Teng[1,2,3] and Rafael A. Irizarry[1,2]

[1]*Department of Biostatistics and Computational Biology, Dana-Farber Cancer Institute, Boston, Massachusetts 02215, USA;* [2]*Department of Biostatistics, Harvard T.H. Chan School of Public Health, Boston, Massachusetts 02115, USA;* [3]*School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China*

The main application of ChIP-seq technology is the detection of genomic regions that bind to a protein of interest. A large part of functional genomics' public catalogs is based on ChIP-seq data. These catalogs rely on peak calling algorithms that infer protein-binding sites by detecting genomic regions associated with more mapped reads (coverage) than expected by chance, as a result of the experimental protocol's lack of perfect specificity. We find that GC-content bias accounts for substantial variability in the observed coverage for ChIP-seq experiments and that this variability leads to false-positive peak calls. More concerning is that the GC effect varies across experiments, with the effect strong enough to result in a substantial number of peaks called differently when different laboratories perform experiments on the same cell line. However, accounting for GC content bias in ChIP-seq is challenging because the binding sites of interest tend to be more common in high GC-content regions, which confounds real biological signals with unwanted variability. To account for this challenge, we introduce a statistical approach that accounts for GC effects on both nonspecific noise and signal induced by the binding site. The method can be used to account for this bias in binding quantification as well to improve existing peak calling algorithms. We use this approach to show a reduction in false-positive peaks as well as improved consistency across laboratories.

[Supplemental material is available for this article.]

Chromatin immunoprecipitation followed by NGS (ChIP-seq) is widely used for detecting the genomic locations of transcription factor binding and histone modifications. ChIP-seq is widely used, with the majority of data provided by the ENCODE (The ENCODE Project Consortium 2012) and modENCODE (Celniker et al. 2009) projects produced with this technology. After mapping the NGS reads, the main part of the quantitative analysis is to infer the genomic sites where the protein of interest binds by finding regions with an enrichment of mapped reads. The regions reported by this analysis are referred to as peaks due to the appearance of the coverage plots (Pepke et al. 2009). Several competing peak detection algorithms have been described in the literature (Ji et al. 2008; Jothi et al. 2008; Kharchenko et al. 2008; Valouev et al. 2008; Zhang et al. 2008; Rozowsky et al. 2009; John et al. 2011; Rashid et al. 2011). Although details of these competing approaches vary, most follow similar general principles. First, after reads are mapped, coverage is calculated for binned regions of the genome. In principle, only regions including binding sites should have counts larger than zero. However, due to nonspecificity of the experimental protocol, we observe a *background level*. This background level is then modeled, and statistical inference is used to distinguish between count levels that can be explained with the background model and those that are higher than expected by chance. The latter are reported as peaks.

GC-content bias has been reported for several NGS applications (Dohm et al. 2008; Alkan et al. 2009; Cheung et al. 2011; Benjamini and Speed 2012; Jiang et al. 2015). For genomic DNA data, PCR amplification of DNA fragments during library preparation is one factor that introduces this bias (Aird et al. 2011; Ross et al. 2013). The bias has also been observed in RNA-seq data (Love et al. 2016). Solutions to this bias have been published for genomic DNA (Benjamini and Speed 2012; Jiang et al. 2015) and RNA-seq data (Hansen et al. 2012; Love et al. 2016). However, below we explain why these approaches are not directly applicable to ChIP-seq data.

Using ENCODE (The ENCODE Project Consortium 2012) data, we show that GC-content bias is also present in ChIP-seq technology. Furthermore, we demonstrate that the way in which GC content affects coverage varies across samples and laboratories and that this unwanted variability is substantial enough to result in different laboratories calling different regions as peaks. Unfortunately, solutions for GC-bias correction, published for other NGS applications, are not directly applicable to ChIP-seq experiments. This is because, in many instances, binding sites are expected to occur in or near high GC-content regions such as gene promoters. If we naively correct for GC content, we may erase the biologically relevant signals we are interested in detecting. Here, we present an approach based on a mixture model, which accounts for GC-content bias separately for effects related to protein binding and differential nonspecific binding. We demonstrate how this approach greatly reduces false-positive peaks and improves agreement across laboratories. The approach was developed for and tested on punctate ChIP-seq for transcription factors.

Corresponding author: rafa@jimmy.harvard.edu
Article published online before print. Article, supplemental material, and publication date are at http://www.genome.org/cgi/doi/10.1101/gr.220673.117.

27:1930–1938 Published by Cold Spring Harbor Laboratory Press; ISSN 1088-9051/17; www.genome.org

Results

GC affects coverage and it does so differently in different labs

To demonstrate the challenges presented by GC-content bias and the advantages presented by our method, we downloaded and processed raw ENCODE (The ENCODE Project Consortium 2012) ChIP-seq data measuring transcription factor binding (*CTCF*, *POLR2A*, *YY1*, *EP300*, and *GATA2*) in the GM12878, HeLa-S3, HepG2, HUVEC, K562, and NHEK cell lines (see Methods section and Supplemental Table S1). Unless otherwise mentioned, all the examples described in this section are based on the *CTCF* data set. We use this particular binding protein as an example because data is available for all six cell lines and experiments performed by three independent laboratories each running at least two replicated experiments (one laboratory ran three replicates for cell lines GM12878 and K562) with at least one million mapped reads.

To explore the extent and characteristics of the bias, we segmented the genome into 10K base-pair bins. After GC content was computed for each of these bins, for each sample of the HUVEC cell line, we counted reads for each bin. Plotting counts versus GC content reveals two clusters in each of the samples (Fig. 1A–F; Supplemental Fig. S1). The presence of two clusters is in agreement with the previously noted observation that ChIP-seq reads can result from either (1) a background level or (2) protein-binding regions (Zhang et al. 2008), with the latter associated with peaks. In both replicates from one laboratory, we observe that counts increase with GC content in both background and signal clusters (Fig. 1A,B). Of particular concern is the fact that the way GC content affects coverage is different in another laboratory (Fig. 1C,D) in which counts decrease with GC content. In a third laboratory, the GC-content bias is only present in one of the two replicates (Fig. 1E,F). A similar result has been previously reported for genomic DNA (Benjamini

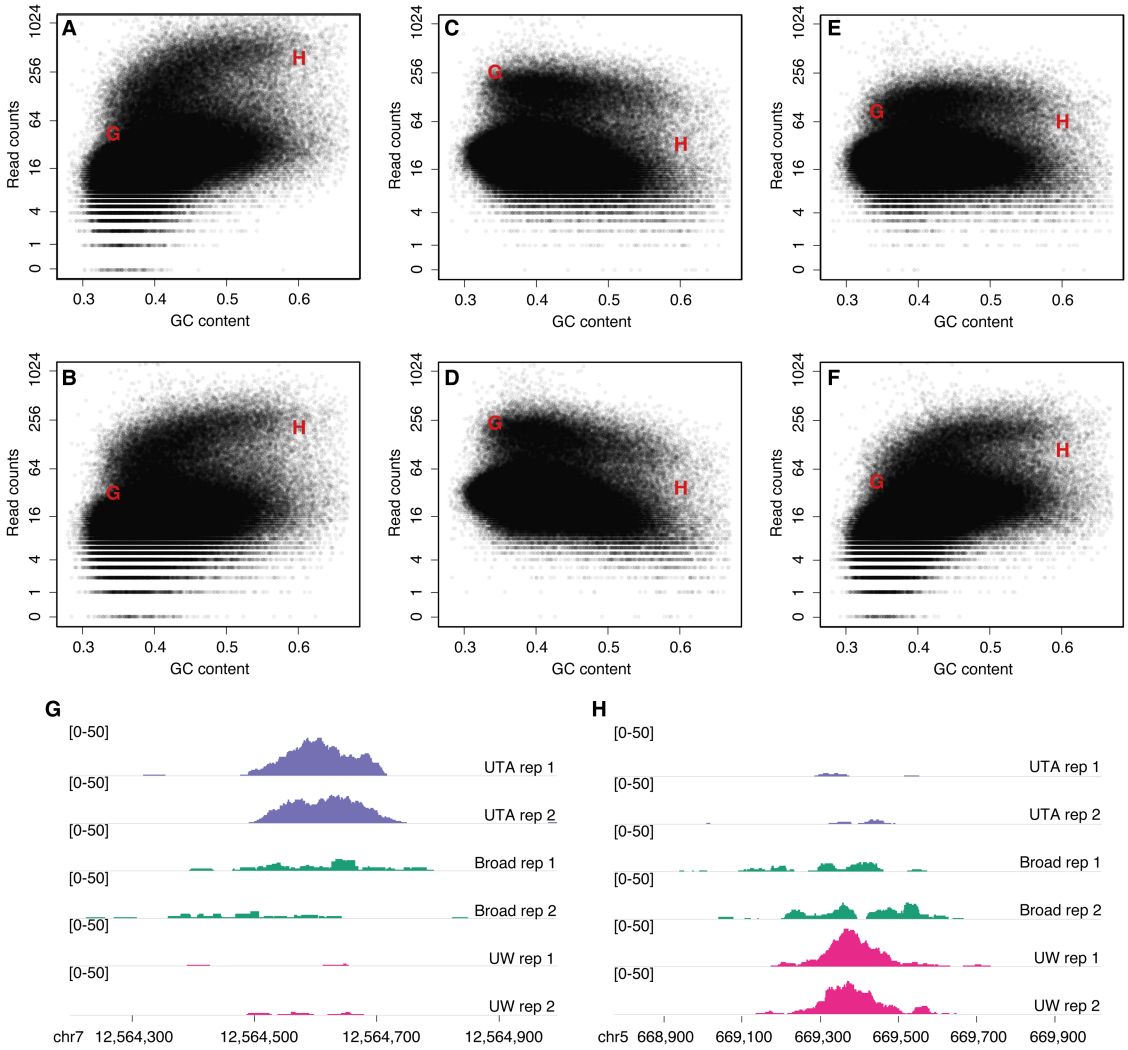

Figure 1. Evidence of GC-content effects at the bin level and its downstream result on peaks demonstrated on the *CTCF* HUVEC cell line. (*A*) The genome is divided into 10-kb bins and counts are computed in the first replicate of laboratory UW as well as the GC content of each bin. Counts are plotted against GC content. (*B*) As in *A* but for the second replicate. (*C*) As in *A* but for the first replicate of laboratory UTA. (*D*) As in *C* but for the second replicate. (*E*) As in *A* but for the first replicate of laboratory Broad. (*F*) As in *E* but for the second replicate. (*G*,*H*) Examples of two peaks that change substantially from laboratory to laboratory. The peaks shown in *G* and *H* were selected to illustrate how coverage plots can change from laboratory to laboratory and that the change appears to be driven by GC content. These regions are associated with the bins annotated with the letters 'G' and 'H' in *A*–*F*.

and Speed 2012; Jiang et al. 2015) and RNA-seq (Love et al. 2016).

Note that here we use the large bin size for exploratory and illustrative purposes only since it results in enough of a reduction in sampling variance to clearly highlight the GC-content bias. Similarly, we used the HUVEC data to illustrate the challenge because it exhibited the strongest GC-content bias. The resulting data visualization (Fig. 1; Supplemental Fig. S1) motivated the approach that we now describe and is applied to smaller bin sizes that are more appropriate for peak calling.

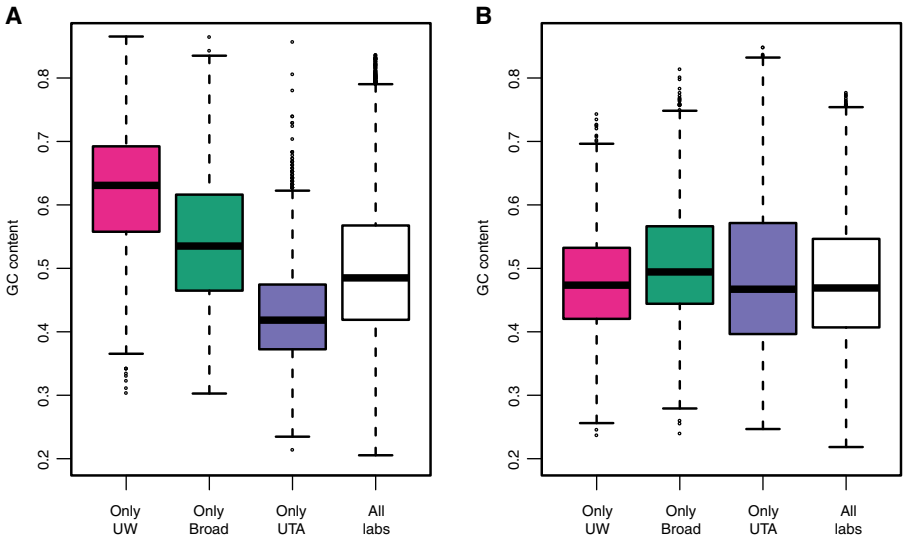

Figure 2. GC content of peaks called by only one laboratory. (*A*) For the *CTCF* HUVEC cell line, we formed four groups of peaks reported by the ENCODE portal. We split them into those called only in laboratory UW, those called only in laboratory Broad, those called only in laboratory UTA, and those called in all three. We computed the GC content for each of these peak regions, shown in four box plots. (*B*) As in *A* but after finding peaks with our algorithm.

GC bias leads to variability of ChIP-seq peak calling

The effects described above are strong enough to affect downstream analysis, such as peak detection. For example, coverage can change drastically across laboratories depending on the GC content of the region (Fig. 1G,H; Supplemental Fig. S2). Note that, in the high GC-content region, laboratory UW shows a peak, but laboratory UTA does not, while in the low GC-content region it is the other way around. These regions are not isolated examples. In fact, the agreement in peak calls across laboratories (Supplemental Fig. S3A) is rather low. For example, the peaks reported for the HUVEC cell line on the ENCODE (Landt et al. 2012) portal report in 37,920, 44,033, and 37,412 peaks called for the three laboratories, respectively, with 24.3% of regions reported by only one laboratory. Note that ENCODE uses the IDR algorithm (Li et al. 2011) to select peaks that consistently appear on two replicates for each laboratory; thus, the number of peaks reported for each laboratory may be significantly affected by the quality of the replicates rather than by suboptimal performance from peak callers. To see if GC content was a major driver of these differences, we compared the GC content of the peak regions detected just by laboratory UW, to those detected just by laboratory Broad, to those detected just by laboratory UTA, and found a strong difference (Fig. 2A). Note that these differences cannot be due to biology but rather must be a result of differences in experimental conditions. These results demonstrate that, if left unaccounted for, GC-content bias will lead current peak callers to report a substantial number of false positives.

Mixture model estimates GC-content effect for background and signal

Published work on GC-content bias correction has found that modeling GC-content effects at the fragment level is, currently, the optimal approach (Benjamini and Speed 2012; Love et al. 2016). However, this approach is not directly applicable to ChIP-seq data. One reason is that most peak calling algorithms operate on bin level information. Specifically, these algorithms define bins, compute coverage in these bins, and then peaks are inferred from these coverage measurements (Ji et al. 2008; Kharchenko et al. 2008; Zhang et al. 2008; John et al. 2011; Rashid et al. 2011). Here we develop a method that makes use of an approximation that permits the adaptation of published peak calling algorithms so that they adjust for GC-content bias. Although we focus on the SPP algorithm (Kharchenko et al. 2008) because it was used by the ENCODE project, our approach applies to any peak algorithm based on coverage computed in bins. The approach can also be used to adjust enrichment scores for predefined regions —for example, regions considered interesting by an investigator or regions reported by a peak caller.

The first step in our approach is to estimate a sample-specific GC-content effect from the data. This effect is defined by estimating the GC-bias for both background level and binding signal for any given potential binding region (Fig. 3A). Suppose our targeted protein binds to a region centered at genomic location i_0 and has length l. Computing the GC content of the genomic region starting at $i_0 - (l/2)$ and ending at $i_0 + (l/2)$ is straightforward. However, due to the fact that DNA is randomly cut into fragments of sizes ranging from 200 to 500 bp (depending on the experimental protocol), the sequenced reads associated with this binding site map to a larger region of the genome (Fig. 3A). Specifically, if fragments are, on average, size h, then the peak region will span from $i_0 + (l/2) - h$ to $i_0 - (l/2) + h$. Note also that once outside of the $[i_0 - (l/2),\ i_0 + (l/2)]$ range, the probability that a specific fragment appears decreases as its center is further from i_0. Specifically, these different probabilities imply that we should use the following weights:

$$w_i = \begin{cases} i - i_0 - \frac{l}{2} + h & \text{if } i \in \left[i_0 + \frac{l}{2} - h, i_0 - \frac{l}{2}\right] \\ h - l + 1 & \text{if } i \in \left(i_0 - \frac{l}{2}, i_0 + \frac{l}{2}\right) \\ i_0 - \frac{l}{2} + h - i & \text{if } i \in \left[i_0 + \frac{l}{2}, i_0 - \frac{l}{2} + h\right] \end{cases}.$$

The shape of w_i can be seen in Fig. 3B. This implies that the total GC-content bias affecting fragments associated with a protein binding at $[i_0 - (l/2),\ i_0 + (l/2)]$ is a weighted average of all the GC-content effects of all potential fragments in the bin $[i_0 +$

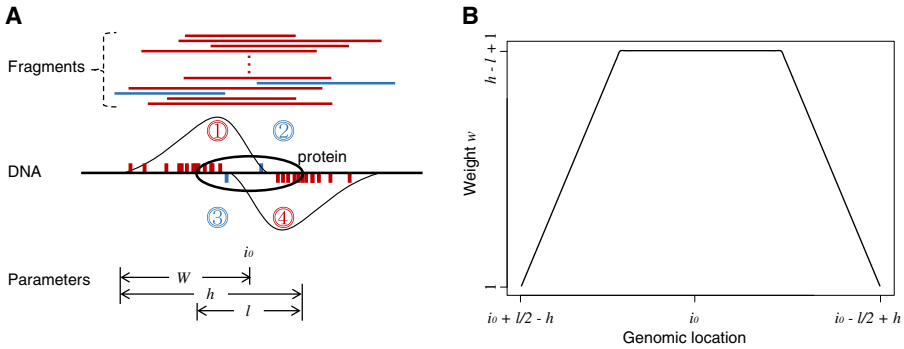

Figure 3. Illustration of regions related to the enrichment score and effective GC-content calculation. (*A*) The regions associated with the counts denoted by $Y_{i,+}$, $Y_{i,-}$, $B_{i,+}$, $B_{i,-}$ in the paper are denoted with the regions 1, 4, 2, and 3, respectively. The red and blue lines represent fragments that are true signals and background noise, respectively, with their start positions labeled as bars on the DNA plot with corresponding colors. The start positions for forward strand fragments are labeled *above* the line while reverse strand fragments are labeled *under* the line. A cartoon indicating the signal coverages formed by forward and reverse strands is also added on the DNA plot. (*B*) Illustration of nucleotide weights when calculating effective GC content for a bin centered at location i_0.

$(l/2) - h$, $i_0 - (l/2) + h$]. We therefore define the *effective GC content* (EGCC) associated with a bin centered at i_0 as

$$\frac{1}{h(h-l+1)} \sum_{i=i_0+\frac{l}{2}-h}^{i_0-\frac{l}{2}+h} w_i x_i,$$

where x_i is 1 if genomic location i is G or C, and 0 otherwise. Note that a positive side effect of this approach is that it results in a GC-content covariate that is less sensitive to the bin size (Supplemental Fig. S4). The parameters l and W (Fig. 3A) are estimated separately for each experiment, as described in detail in the Methods section. Note that $h = W + l/2$.

With an EGCC in place for any given genomic location, we can then estimate GC-content effects for both the background level and signal using a mixture model. Specifically, we pose a mixed generalized linear model with two components corresponding to coverage due to specific binding and background regions, respectively. We assume that each component follows a negative binomial distribution with the log of the rate a smooth function of EGCC. When fitting this model to the HUVEC data, the fitted GC-content-dependent effects demonstrate that each laboratory introduces a different type of bias for both the signal and background (Fig. 4). See the Methods section for details.

Adjusting binding quantification for GC-bias reduces batch effects

We computed counts for each of the regions reported as *CTCF* peaks in at least one cell line by ENCODE (Kundaje et al. 2015) for each of the GM12878, HeLa-S3, HepG2, HUVEC, K562, and NHEK samples. We constructed a matrix with these binding quantifications and performed principal component analysis (PCA) on the log-transformed values in this matrix. The first two principal components (PCs) do not separate by cell line, as expected (Fig. 5A). Furthermore, the large variation seen within each cell line is largely explained by the different laboratories (Fig. 5A). We then adjusted the values in this matrix for GC content using our model-based approach and recomputed the PCs. The results were markedly improved (Fig. 5B), with the samples now clearly clustering by cell line and much of the batch effects removed. The improvement in specificity and batch effect removal was evident from plotting mean squared residuals summarizing across laboratory variability, computed within cell line, before and after GC-content correction and noting a substantial reduction (Fig. 5C). Because the variability is driven by the difference in GC-content biases rather than laboratory, we expect our method to outperform batch-effect adjustment tools such as ComBat (Johnson et al. 2007). As the GC-content effects and laboratories are not perfectly correlated, we expect some of the GC-content bias to remain even after applying ComBat. We confirm this by rerunning the PCA after applying ComBat and noting that the unwanted variability is not removed (Supplemental Fig. S5). In addition, our approach has the further advantage that it can be applied to a single experiment, while ComBat requires multiple samples.

To show that our approach improves downstream results for other transcription factors, we applied the same analysis to *POLR2A*, another binding protein run on several cell lines and across three laboratories (see Methods and Supplemental Table S1). Although the GC-content effect was not as strong in this experiment as in the *CTCF* experiment, our method still showed improved specificity (Supplemental Fig. S6).

Integrating GC-content adjustment into peak calling algorithms

Our model-based approach provides an adjustment value for any genomic bin. This implies that it can be integrated with peak algorithms that use bins as raw data (Ji et al. 2008; Kharchenko et al. 2008; Zhang et al. 2008; Rashid et al. 2011). Here, we demonstrate the advantages of our approach by adapting the peak detection algorithm used by ENCODE, namely the ChIP-seq processing pipeline (SPP) (Kharchenko et al. 2008). SPP starts by estimating the average half-width of the binding protein, referred to here as W (see Methods section for details). With this estimate in place, the SPP algorithm then computes read counts for positive and negative strands separately for each genomic location i, denoted here with $Y_{i,+}$ and $Y_{i,-}$, respectively (red fragments in Fig. 3A). The $Y_{i,+}$ represents positive strand counts in a region starting at $i - W$ and ending at i, and the $Y_{i,-}$ represents negative strand counts in a region starting at i and ending at $i + W$. As described by Pepke et al. (2009), these counts should be large when a protein binds a region centered at i. To account for local background, SPP also computes counts in regions that should be associated with nonspecific binding, denoted here with $B_{i,+}$ and $B_{i,-}$, respectively (blue fragments in Fig. 3A). The background level $B_{i,+}$ represents positive strand counts in a region starting at i and ending at $i + W$, and the background level $B_{i,-}$ represents negative strand counts in a region starting at $i - W$ and ending at i. As described by Pepke et al. (2009), there should be no counts in these regions when a protein binds a region centered at i. Then, for each i, SPP defines the *enrichment score* as a geometric mean of the signal counts minus the average background signal $S_i = 2\sqrt{Y_{i,+} \times Y_{i,-}} - (B_{i,+} + B_{i,-})$. Note that this is the geometric average of the signal minus the arithmetic average of background multiplied by two. To find candidate peaks, SPP then estimates binding significance and uses the local maxima of enrichment scores to call peaks. To correct for

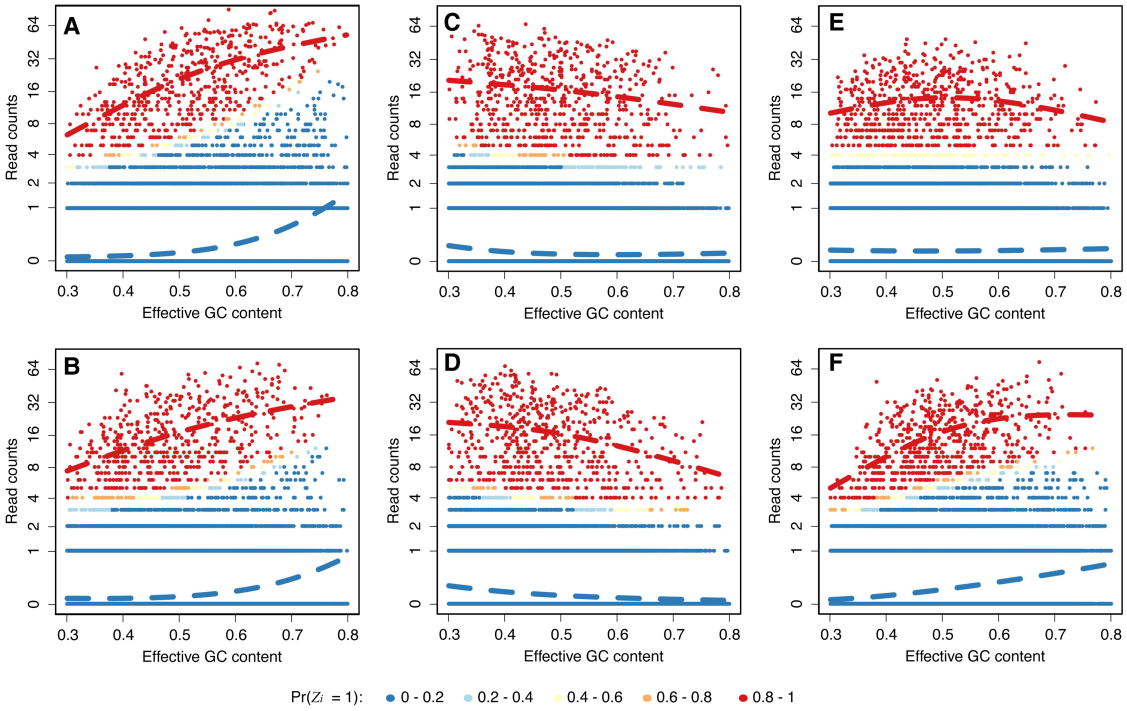

Figure 4. Visualization of the fitted generalized linear mixture model. (*A*) We defined bins using estimated binding size (*W* in Fig. 3A) and randomly selected 5% of all genome-wide bins. We computed counts for these bins in the first replicate of the *CTCF* HUVEC cell line for laboratory UW. We fitted our model to these bins. The colors represent the probability of being background (blue) or signal (red). The GC-content bias smooth functions are plotted with dashed curves. (*B*) As in *A* but for the second replicate for laboratory UW. (*C*) As in *A* but for laboratory UTA. (*D*) As in *C* but for the second replicate for laboratory UTA. (*E*) As in *A* but for laboratory Broad. (*F*) As in *E* but for the second replicate for laboratory Broad.

GC, we simply compute the effective GC content of each bin and adjust $Y_{i,+}$, $Y_{i,-}$, $B_{i,+}$, $B_{i,-}$ accordingly (see Methods). We then used an approach similar to SPP to quantify uncertainty for each candidate peak reported by our GC-corrected SPP algorithm (Methods and Supplemental Fig. S7). We compared these peaks to those obtained by the original SPP and found that our method (gcapc) resulted in substantial improvement in consistency (Fig. 6A). If, as done by ENCODE, we filter peaks using an IDR (Li et al. 2011) of 0.02, our algorithm reports improved results of 31,051, 33,769, and 30,250 peaks (Supplemental Fig. S3B) called for the three laboratories and now only 16.9% of regions reported by only one laboratory. More importantly, the differences are no longer due to differences in GC content (Fig. 2B). Note that the ENCODE pipeline is more complicated than running a peak caller and IDR (https://www.encodeproject.org/chip-seq/transcription_factor/). If we simply run SPP followed by IDR, the improvements of our algorithm are even larger, since this approach produced 29.5% of regions reported by only one laboratory (Supplemental Fig. S3C).

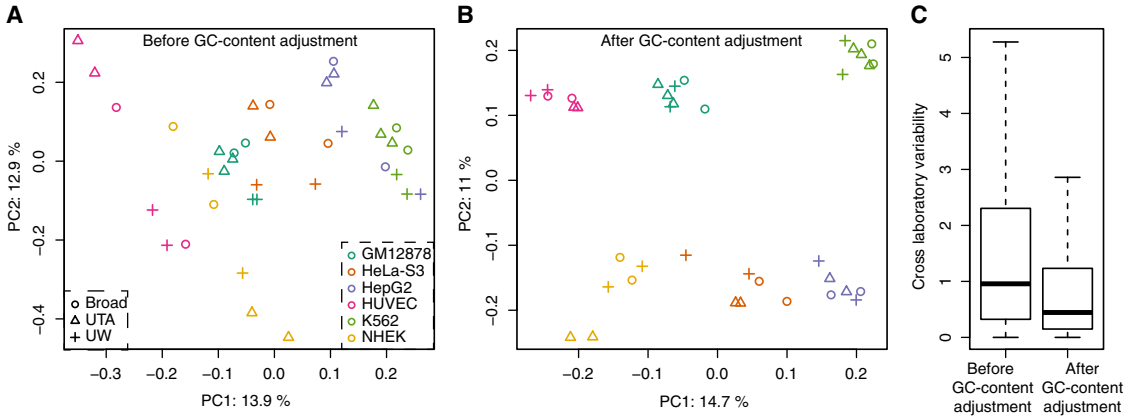

Figure 5. GC-bias correction reduces the impact of batch effects. (*A*) For the regions reported as binding sites by ENCODE, we computed counts for the GM12878, HeLa-S3, HepG2, HUVEC, K562, and NHEK for *CTCF*. The first two principal components of this matrix are shown, with color representing cell line and different symbols used to represent laboratory. (*B*) As in *A* but after performing the batch correction. (*C*) Box plots showing the within cell line across laboratory variability before and after correction for the HUVEC cell line.

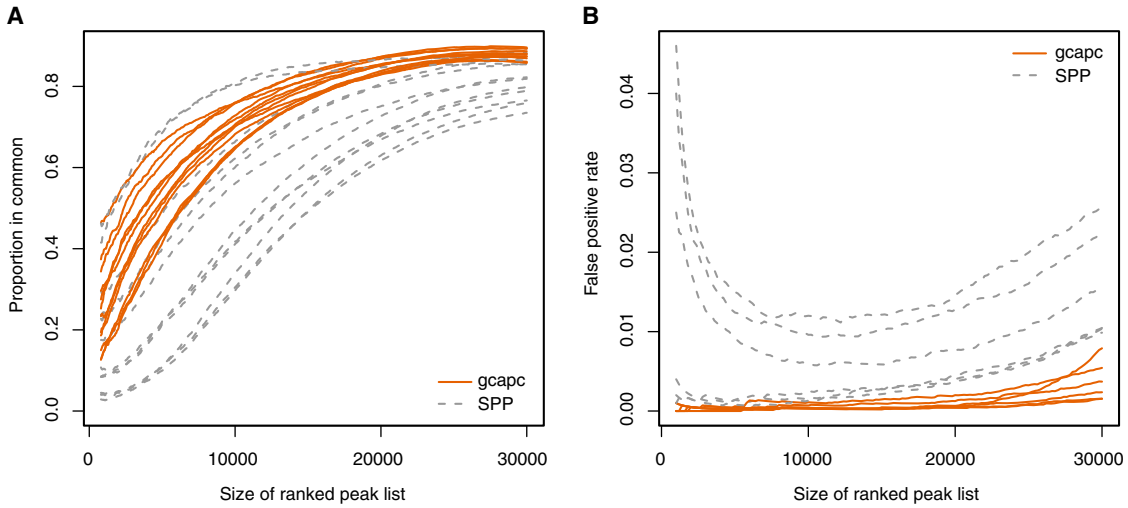

Figure 6. Improvements of peak calling consistency and transcription factor binding site enrichment with GC-content adjustment. For the *CTCF* HUVEC cell line, we create lists of candidate peaks and rank them based on enrichment score. (*A*) Correspondence at the top (CAT) plots. For each list size, we compute the proportion of peaks in common reported by two different laboratories. We do this for each pairwise comparison and plot this percentage against the list size. Peak width is scaled to the same median size between gcapc and SPP for each sample. (*B*) For each list size, we compute PWM scores for each peak and define with lower than 72% as false positives. We do this for each replicate and plot the false-positive rate against the list size and plot the number of false positives for each list size ranging from 1 to 30,000.

Because IDR analysis is sensitive to the quality of replicates (Li et al. 2011), to provide a more systematic comparison we generated a CAT plot (Irizarry et al. 2005), including the comparison of each sample (Fig. 6A).

Finally, to further assess the improvement provided by our approach, we performed *CTCF* binding site enrichment analysis. Specifically, we used the human *CTCF* motif from the JASPAR 2016 database (Mathelier et al. 2016) to define a position weight matrix (PWM) score sequence of the same size as the motif (Wasserman and Sandelin 2004). A background probability of 30% for A and T and 20% for C and G were used for the PWM score calculation. Then, we assigned a PWM score to each reported peak by selecting the maximum PWM within the region associated with the peak. Following Sandelin et al. (2004) and Wasserman and Sandelin (2004), we defined peaks with a maximum PWM score lower than 72% of all possible PWM scores for this *CTCF* motif as a false positive. gcapc had substantially less false positives than SPP (Fig. 6B). For example, if we examine the top 100 peaks across all six replicates, SPP results in a total of 54 false positives, while gcapc has none. The improvements were consistent across several cut-off choices ranging from 67% to 92% (Supplemental Fig. S8).

To demonstrate that the improvements offered by gcapc generalize to other transcription factors, we applied our approach to the ChIP-seq data of three other transcription factors, *YY1*, *EP300*, and *GATA2* (see Methods and Supplemental Table S1). Our method reduced the inconsistencies between laboratories compared to SPP (Supplemental Fig. S9). We note that the improvements for the *GATA2* data set are less substantial. This appears to be the case because the GC-content bias is less pronounced in this data set. Note that using our R package, one can visualize the data (Fig. 4) to determine the severity of the GC-content effect. If the fitted curves look flat for all your data sets, a GC-bias method might not be necessary.

Finally, to demonstrate that our approach can be incorporated into other peak callers, we adapted the MACS2 (Zhang et al. 2008) and hotspot (John et al. 2011) algorithms (see Methods). Note that the MACS2 method estimates a local background level from the data that could, in principle, correct for sequence bias. However, because this background level estimation procedure does not appear to be local enough to capture the GC effects, as a result we see similar inconsistencies in MACS2 as in SPP (Supplemental Fig. S10). The GC-bias correction added by our approach showed marked improvements for these two peak calling algorithms as well (Supplemental Fig. S11).

Discussion

We have demonstrated how GC-content bias induces substantial variability into ChIP-seq data and that this variability is large enough to result in different peaks being reported by different laboratories when studying the same cell lines. We described how published GC-content adjustment methods are not directly applicable to ChIP-seq data due to confounding between the GC content of regions and their biological relevance. We described gcapc (http://bioconductor.org/packages/gcapc/), a method that adjusts for GC-content bias in ChIP-seq data using a mixed model, which permits independent adjustments of the signal and background signals and thus circumvents the confounding challenge and can be incorporated into most current peak callers. Our method permits the GC-content bias correction for any predefined bin. We demonstrated the practical advantage of our approach by removing batch effects from binding quantifications in ENCODE data and by adapting the widely used SPP algorithm and showing substantial improvements in peak calling consistency across laboratories.

Note that, although we tested our approach on several transcription factors, all are examples of what are referred to as punctate data sets: characterized by peaks that are short and marked. We do not yet recommend our method for broad peak data sets such as histone modifications. Extending our GC-bias adjusting method to these data sets is the subject of future work.

Methods

Data acquisition and preprocessing

We chose data for the transcription factors *CTCF*, *POLR2A*, *YY1*, *EP300*, and *GATA2* provided by ENCODE as example data sets because they include a wide range of cell types or experiments performed by two to three different laboratories using the same protocol. The five production centers (laboratories) which conducted experiments on selected transcription factors were located at the Broad Institute (Broad), HudsonAlpha Institute for Biotechnology (HAIB), Stanford University or University of Southern California (SYDH), University of Texas at Austin (UTA), and University of Washington (UW). For *CTCF*, we focused on the GM12878, HeLa-S3, HepG2, HUVEC, K562, and NHEK cell lines because each of these was processed in replicates by each of three laboratories: Broad, UTA, and UW. Similarly, we focused on GM12878, HeLa-S3, HepG2, and HUVEC cell line samples for *POLR2A* by HAIB, SYDH, and UTA. For the other three factors, we only selected one cell line in two laboratories for analysis (Supplemental Table S1). Raw sequencing reads were downloaded from the ENCODE data portal (https://www.encodeproject.org/) or UCSC ENCODE portal (https://genome.ucsc.edu/ENCODE/) using accession IDs or links documented in Supplemental Table S1.

The raw reads were aligned to human genome build hg19 with aligner BWA (Li and Durbin 2009). We note that since our analysis is based on bin counts, with bins of size 100–250 bp, given the characteristics of the differences between hg19 and GRCh38, realigning the reads to GRCh38 will not affect our conclusions. Reads from Chromosome Y were ignored to avoid sex effects. Mapped reads with a mapping score less than 30 were removed. Secondary alignments were also removed. Duplicate reads were thinned down to one read. For the purposes of quantifying binding in predefined regions, we only considered the start position at the 5′ end.

Estimating GC-content bias with the mixed generalized linear model

Figures 1 and 4 clearly demonstrate the presence of two clusters. We assume the cluster characterized by low counts is related to nonspecific binding and refer to it as the *background*. We assume that the cluster characterized by higher counts is related to the specific binding signals that constitute the peaks. The counts in both clusters show a strong nonlinear dependence on GC content and motivate the following mixed model. We assume that for any given position i, $Z_i = 1$ if binding occurs at that position, and 0 otherwise. We denote with π the probability that any given $Z_i = 1$. We then assume that, conditioned on the state Z_i, the counts Y_i follow a negative binomial distribution with $\log(\mathrm{E}[Y_i|Z_i = a, X_i = x_i]) = \mu_a + f_a(x_i)$, with μ_a the mean count level for the positions, x_i the effective GC content for position i, and f_a is a smooth function that we represent with a cubic spline. Note that a is indexing the two possible states, background or specific signal, which implies that the GC-content effect is modeled differently for each state.

Because we start with millions of bins, to improve computational efficiency we selected a random subset of bins representing 5% of the genome and estimated the parameters π, μ_0, μ_1, and the parameters used to represent the splines f_0 and f_1 using an expectation–maximization (EM) algorithm on this subset. We repeated this procedure with five independent subsets and updated the estimates to be the average across the five resulting sets of parameters. Using 15% and 25% resulted in practically identical estimates (Supplemental Fig. S12). The GC-content effect for binding quantification is simply

$$e^{(1-\hat{Z}_i)\hat{f}_0(x_i)+\hat{Z}_i\hat{f}_1(x_i)},$$

with $\hat{Z}_i = \Pr(Z_i = 1)$ the estimate obtained with the EM algorithm. To correct for the GC-content bias, we simply divide the counts by this quantity: $Y_i/e^{(1-\hat{Z}_i)\hat{f}_0(x_i)+\hat{Z}_i\hat{f}_1(x_i)}$.

To extend SPP, we use the correction $e^{\hat{f}_0(x_{i,1,+})}$ and $e^{\hat{f}_0(x_{i,1,-})}$ for the $Y_{i,+}$ and $Y_{i,-}$, respectively, where $x_{i,1,+}$ and $x_{i,1,-}$ are the effective GC content in the positive and negative strand bins, described above, respectively. Similarly, we used the correction $e^{\hat{f}_0(x_{i,0,+})}$ and $e^{\hat{f}_0(x_{i,0,-})}$ for the $B_{i,+}$ and $B_{i,-}$, respectively. Note that here, we use f_0 for both signal and background components, because this summary is intended as a test statistic for which we define a null distribution assuming there is no signal. The term f_1 is therefore only used when fitting the model and to correct region binding quantification already determined to be potential peaks.

Expectation–maximization (EM) algorithm for fitting the mixture model

The log-likelihood for the expectation step is as follows:

$$\mathrm{LL}(\mu, \theta, f, p) = \log \prod_i \left[\mathrm{nb}_1(Y_i|f_1, \mu_1, \theta_1, x_i) \times p\right]^{Z_i} \times \left[\mathrm{nb}_0(Y_i|f_0, \mu_0, \theta_0, x_i) \times (1-p)\right]^{1-Z_i},$$

where nb_a represents the probability density function of a negative binomial distribution with log mean $\mu_a + f_a(x_i)$ and shape parameter θ_a. p is the probability of a bin belonging to a signal mixture component. The other parameters are defined above.

In the maximization step, the parameters are estimated as follows:

$$\hat{Z}_i = \frac{p \times \mathrm{nb}_1(Y_i|f_1, \mu_1, \theta_1, x_i)}{p \times \mathrm{nb}_1(Y_i|f_1, \mu_1, \theta_1, x_i) + (1-p) \times \mathrm{nb}_0(Y_i|f_0, \mu_0, \theta_0, x_i)},$$

$$\hat{p} = \frac{c + \sum_{i=1}^{n} \hat{Z}_i}{2 \times c + n},$$

c is a constant 2, n is total bin number, μ, θ, f are estimated using the *glm.nb* function in the R package MASS, with μ_1, θ_1, f_1 based on bins with $\hat{Z}_i \geq 0.5$, and μ_0, θ_0, f_0 based on bins with $\hat{Z}_i < 0.5$.

Analysis of regions reported by ENCODE

For the analysis involving the binding regions reported by ENCODE (https://www.encodeproject.org/data/annotations/v2/), we did not need to run a peak calling algorithm since regions were already provided. We used data from all the regions reported by ENOCDE as potential peaks. To perform a GC-content bias correction of binding quantification, we assumed a binding width of 150 bp and used flanking regions of width 250 bp. We fit the model described above and we can correct as described. PCA analysis was based on binding regions reported for GM12878, HeLa-S3, HepG2, HUVEC, K562, and NHEK cell lines for *CTCF* and GM12878, HeLa-S3, HepG2, and HUVEC for *POLR2A*. As an example, the across-laboratory variability was computed in the HUVEC cell line for *CTCF*.

Quantifying uncertainty

We implement a method similar to SPP. Specifically, we compute the enrichment score S_i for each region i. Then, for each of these regions, we permuted the start sites of all the reads falling within the region and recomputed the enrichment scores, denoted here with S_i^*. We used the S_i^* to form a null distribution and assign a P-value to each candidate peak (Supplemental Fig. S7). The user should treat the P-values obtained from this procedure, as well as SPP, with caution as they are based on several assumptions that are hard to test empirically. Furthermore, these uncertainty

estimates do not account for the selection process. Permutation approaches such as those implemented by the bumphunter approach (Aryee et al. 2014) are the subject of future research. Regardless, we find this quantification useful for prioritizing peaks.

Other improvements on the SPP algorithm

Apart from the GC-content correction, we adapted the SPP algorithm in three other ways, which we describe in detail here.

The SPP algorithm computes the enrichment score S_i for every location i on the genome. To do this, SPP defines a window size W that is used to compute the read counts as described in the Results section (Fig. 3A). This window size is supposed to define the region that includes fragments resulting with a protein binding at location i. To define W, SPP uses the cross-correlation function between the fragment start sights from positive and negative strands. SPP uses an ad hoc procedure picking W to be a number between the lag that maximizes the cross-correlation function and a defined maximum window size that defaults to 500 bp. Instead, we make the assumption that the window size W should maximize the correlation of read counts between positive-strand windows and corresponding negative-strand windows. Using this criteria, the estimated window sizes in *CTCF* data sets are always much smaller than those estimated by SPP. In addition, we define another parameter l (Fig. 3A) to represent the real protein-binding width for effective GC-content estimation. This parameter is estimated using the lag that maximizes the cross-correlation function.

The second difference is related to computational efficiency. While SPP computes S_i and the uncertainty estimate for every location on the genome, in our software we perform a filter that removes regions with small counts in all four relevant bins ($Y_{i,+}$, $Y_{i,-}$, $B_{i,+}$, $B_{i,-}$). This reduces the number of regions for which the uncertainty quantification is computed.

Finally, to report the center of the binding site, SPP searches for local maxima of enrichment scores S_i. However, distributions of enrichment scores are not always symmetric around these local maxima, and we find that neighboring peak regions sometimes represent the same binding site. To avoid reporting two maxima associated with the same binding site as two separate peaks, we merged any two neighboring peaks that are within the estimated binding width from each other. Uncertainty is quantified only for merged peaks.

Adapting other peak callers

First, optimal bin size and the parameter estimates needed to describe the GC-content effects are obtained using the same strategy described above. Then, the peak caller—MACS2 or hotspot, for example—is used to identify peaks. Then, each candidate region is extended by half the bin size at both ends to ensure every position in the reported peaks receives an effective GC content. For these regions, we then move a sliding window of the size of a bin, in steps of 1 bp and compute read counts and effective GC content in each. Now, for each bin we have a count and an effective GC content, so using the estimated GC-content bias model, we can perform a correction for each of these bins using the same approach as described above for SPP. The maximum corrected binding score with each region is reported as a summary binding quantification for that peak. Permutation is performed as described before. It is noted that a flexible set of peak regions is required to use this functionality to ensure the accuracy of permutation analysis.

Software availability

The method described in this manuscript is available as an R/Bioconductor package (http://bioconductor.org/packages/gcapc/). The source code for the main results is documented here (https://github.com/tengmx/gcapc_manuscript). Both the R package and source code are also available in the Supplemental Material.

Acknowledgments

We thank Anshul Kundaje for the suggestion on transcription factor binding sites analysis. We acknowledge support from the National Human Genome Research Institute (grant numbers U24HG009446 and R01HG005220).

References

Aird D, Ross MG, Chen WS, Danielsson M, Fennell T, Russ C, Jaffe DB, Nusbaum C, Gnirke A. 2011. Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. *Genome Biol* **12:** R18.

Alkan C, Kidd JM, Marques-Bonet T, Aksay G, Antonacci F, Hormozdiari F, Kitzman JO, Baker C, Malig M, Mutlu O, et al. 2009. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nat Genet* **41:** 1061–1067.

Aryee MJ, Jaffe AE, Corrada-Bravo H, Ladd-Acosta C, Feinberg AP, Hansen KD, Irizarry RA. 2014. Minfi: a flexible and comprehensive Bioconductor package for the analysis of Infinium DNA methylation microarrays. *Bioinformatics* **30:** 1363–1369.

Benjamini Y, Speed TP. 2012. Summarizing and correcting the GC content bias in high-throughput sequencing. *Nucleic Acids Res* **40:** e72.

Celniker SE, Dillon LA, Gerstein MB, Gunsalus KC, Henikoff S, Karpen GH, Kellis M, Lai EC, Lieb JD, MacAlpine DM, et al. 2009. Unlocking the secrets of the genome. *Nature* **459:** 927–930.

Cheung MS, Down TA, Latorre I, Ahringer J. 2011. Systematic bias in high-throughput sequencing data and its correction by BEADS. *Nucleic Acids Res* **39:** e103.

Dohm JC, Lottaz C, Borodina T, Himmelbauer H. 2008. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Res* **36:** e105.

The ENCODE Project Consortium. 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature* **489:** 57–74.

Hansen KD, Irizarry RA, Wu Z. 2012. Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics* **13:** 204–216.

Irizarry RA, Warren D, Spencer F, Kim IF, Biswal S, Frank BC, Gabrielson E, Garcia JG, Geoghegan J, Germino G, et al. 2005. Multiple-laboratory comparison of microarray platforms. *Nat Methods* **2:** 345–350.

Ji H, Jiang H, Ma W, Johnson DS, Myers RM, Wong WH. 2008. An integrated software system for analyzing ChIP-chip and ChIP-seq data. *Nat Biotechnol* **26:** 1293–1300.

Jiang Y, Oldridge DA, Diskin SJ, Zhang NR. 2015. CODEX: a normalization and copy number variation detection method for whole exome sequencing. *Nucleic Acids Res* **43:** e39.

John S, Sabo PJ, Thurman RE, Sung MH, Biddie SC, Johnson TA, Hager GL, Stamatoyannopoulos JA. 2011. Chromatin accessibility pre-determines glucocorticoid receptor binding patterns. *Nat Genet* **43:** 264–268.

Johnson WE, Li C, Rabinovic A. 2007. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8:** 118–127.

Jothi R, Cuddapah S, Barski A, Cui K, Zhao K. 2008. Genome-wide identification of in vivo protein-DNA binding sites from ChIP-seq data. *Nucleic Acids Res* **36:** 5221–5231.

Kharchenko PV, Tolstorukov MY, Park PJ. 2008. Design and analysis of ChIP-seq experiments for DNA-binding proteins. *Nat Biotechnol* **26:** 1351–1359.

Kundaje A, Meuleman W, Ernst J, Bilenky M, Yen A, Heravi-Moussavi A, Kheradpour P, Zhang Z, Wang J, Ziller MJ, et al. 2015. Integrative analysis of 111 reference human epigenomes. *Nature* **518:** 317–330.

Landt SG, Marinov GK, Kundaje A, Kheradpour P, Pauli F, Batzoglou S, Bernstein BE, Bickel P, Brown JB, Cayting P, et al. 2012. ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Res* **22:** 1813–1831.

Li H, Durbin R. 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25:** 1754–1760.

Li Q, Brown JB, Huang H, Bickel PJ. 2011. Measuring reproducibility of high-throughput experiments. *Ann Appl Stat* **5:** 1752–1779.

Love MI, Hogenesch JB, Irizarry RA. 2016. Modeling of RNA-seq fragment sequence bias reduces systematic errors in transcript abundance estimation. *Nat Biotechnol* **34:** 1287–1291.

Mathelier A, Fornes O, Arenillas DJ, Chen CY, Denay G, Lee J, Shi W, Shyr C, Tan G, Worsley-Hunt R, et al. 2016. JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles. *Nucleic Acids Res* **44:** D110–D115.

Pepke S, Wold B, Mortazavi A. 2009. Computation for ChIP-seq and RNA-seq studies. *Nat Methods* **6:** S22–S32.

Rashid NU, Giresi PG, Ibrahim JG, Sun W, Lieb JD. 2011. ZINBA integrates local covariates with DNA-seq data to identify broad and narrow regions of enrichment, even within amplified genomic regions. *Genome Biol* **12:** R67.

Ross MG, Russ C, Costello M, Hollinger A, Lennon NJ, Hegarty R, Nusbaum C, Jaffe DB. 2013. Characterizing and measuring bias in sequence data. *Genome Biol* **14:** R51.

Rozowsky J, Euskirchen G, Auerbach RK, Zhang ZD, Gibson T, Bjornson R, Carriero N, Snyder M, Gerstein MB. 2009. PeakSeq enables systematic scoring of ChIP-seq experiments relative to controls. *Nat Biotechnol* **27:** 66–75.

Sandelin A, Wasserman WW, Lenhard B. 2004. ConSite: web-based prediction of regulatory elements using cross-species comparison. *Nucleic Acids Res* **32:** W249–W252.

Valouev A, Johnson DS, Sundquist A, Medina C, Anton E, Batzoglou S, Myers RM, Sidow A. 2008. Genome-wide analysis of transcription factor binding sites based on ChIP-seq data. *Nat Methods* **5:** 829–834.

Wasserman WW, Sandelin A. 2004. Applied bioinformatics for the identification of regulatory elements. *Nat Rev Genet* **5:** 276–287.

Zhang Y, Liu T, Meyer CA, Eeckhoute J, Johnson DS, Bernstein BE, Nusbaum C, Myers RM, Brown M, Li W, et al. 2008. Model-based analysis of ChIP-seq (MACS). *Genome Biol* **9:** R137.

Received January 15, 2017; accepted in revised form August 14, 2017.