Bioinformatics, 33(15), 2017, 2258–2265
doi: 10.1093/bioinformatics/btx150
Advance Access Publication Date: 22 March 2017
Original Paper

Genome analysis

GenoGAM: genome-wide generalized additive models for ChIP-Seq analysis

Georg Stricker[1,2,†], Alexander Engelhardt[1,†], Daniel Schulz[1], Matthias Schmid[3], Achim Tresch[4] and Julien Gagneur[1,2,*]

[1]Gene Center and Department of Biochemistry, Ludwig-Maximilians-Universität München, 80333 Munich, Germany, [2]Department of Informatics, Technische Universität München, 85748 Garching, Germany, [3]Institut für Medizinische Biometrie, Informatik und Epidemiologie, University Hospital Bonn, 53105 Bonn, Germany and [4]Institute for Genetics, University of Cologne, 50647 Cologne, Germany

*To whom correspondence should be addressed.

†The authors wish it to be known that these authors contributed equally.

Associate Editor: Janet Kelso

Received on April 22, 2016; revised on February 1, 2017; editorial decision on March 6, 2017; accepted on March 20, 2017

Abstract

Motivation: Chromatin immunoprecipitation followed by deep sequencing (ChIP-Seq) is a widely used approach to study protein–DNA interactions. Often, the quantities of interest are the differential occupancies relative to controls, between genetic backgrounds, treatments, or combinations thereof. Current methods for differential occupancy of ChIP-Seq data rely however on binning or sliding window techniques, for which the choice of the window and bin sizes are subjective.

Results: Here, we present GenoGAM (Genome-wide Generalized Additive Model), which brings the well-established and flexible generalized additive models framework to genomic applications using a data parallelism strategy. We model ChIP-Seq read count frequencies as products of smooth functions along chromosomes. Smoothing parameters are objectively estimated from the data by cross-validation, eliminating *ad hoc* binning and windowing needed by current approaches. GenoGAM provides base-level and region-level significance testing for full factorial designs. Application to a ChIP-Seq dataset in yeast showed increased sensitivity over existing differential occupancy methods while controlling for type I error rate. By analyzing a set of DNA methylation data and illustrating an extension to a peak caller, we further demonstrate the potential of GenoGAM as a generic statistical modeling tool for genome-wide assays.

Availability and Implementation: Software is available from Bioconductor: https://www.bioconductor.org/packages/release/bioc/html/GenoGAM.html.

Contact: gagneur@in.tum.de

Supplementary information: Supplementary information is available at *Bioinformatics* online.

1 Introduction

Chromatin immunoprecipitation followed by deep sequencing (ChIP-Seq) is the reference method used for genome-wide quantification of protein–DNA interactions (Robertson *et al.*, 2007). It is used to study a wide range of fundamental processes covering transcription, replication, and genome maintenance.

ChIP-Seq consists of cross-linking DNA with chromatin, followed by DNA fragmentation and immunoprecipitation of the protein of interest along with its bound DNA fragments. The DNA fragments are then released, amplified, and sequenced. ChIP-Seq has been applied to study DNA-bound proteins of various functions and therefore with various patterns of distribution along the genome.

These include transcription factors that are bound at discrete binding sites (Barski *et al.*, 2007; Johnson *et al.*, 2007), histone modifications (Albert *et al.*, 2007; Barski *et al.*, 2007) which are found at nucleosomes, or the transcription (Barski *et al.*, 2007) and replication machinery which are even more broadly distributed.

Often, the quantities of interest are the occupancies relative to technical controls such as the input (a sample that was not subject to the immunoprecipitation step), between genetic backgrounds, treatments, or combinations thereof. Testing for local occupancy differences between multiple groups of ChIP-Seq samples is therefore an important goal in ChIP-Seq analysis.

In principle, testing could be performed at every single base of the genome. However, the per-base coverage is in practice too low and too noisy for such an approach to have enough statistical power. Testing for differential occupancy has been therefore done by integration of data over larger regions. Generally, benchmarks have shown that differential occupancy perform better if they handle replicate samples (Steinhauser *et al.*, 2016). DESeq tests for differential overall occupancies at pre-defined regions of interest by testing for differences in number of reads overlapping the region (Anders and Huber, 2010). Complementary to testing for overall occupancies, MMDiff (Schweikert *et al.*, 2013) allows testing for differences in shapes in given regions. Other approaches are scanning the genomes or large areas for local differential occupancies. These include diffReps (Shen *et al.*, 2013), where a sliding window moves along the genome in a fixed step size and a robust test based on negative binomial distribution is performed on the number of reads falling into the window. PePr (Zhang *et al.*, 2014) follows a similar scanning approach and estimates local variance. Lun and Smyth (2014) demonstrate how to test differential occupancies across windows in given regions while properly controlling for false discovery rate. Their approach has been published in an R package called csaw. THOR (Allhoff *et al.*, 2016) uses a Hidden Markov Model approach to segment the genome into regions that are enriched, depleted or not differentially occupied. Of these methods, only DESeq and csaw, which are based on generalized linear models (Nelder and Wedderburn, 1972), can go beyond comparisons between two groups of samples by supporting any full factorial designs including crossed designs. Moreover, all current methods rely on binning or sliding window techniques, for which the choice of the window and bin sizes are not data-driven but subjective.

Here, we introduce GenoGAM, which brings generalized additive models to genomic applications (Fig. 1). Generalized additive models are extensions of generalized linear models for which covariates can be modeled as smooth functions (Hastie *et al.*, 1986). We use them to model ChIP-Seq count rates along the genome. GenoGAM normalizes for sequencing depth and can handle factorial experimental designs, including biological replicates and multiple controls. The amount of smoothing is estimated in an automatic, data-driven manner and thus avoids introducing subjectivity from the analyst. When analyzing differential binding in a factorial design, we obtain well-calibrated per-base-pair *P*-values and region-wise *P*-values. Application to a dataset of yeast shows that GenoGAM is more sensitive than state-of-the art differential occupancy methods.

GenoGAM brings further modeling flexibility for which we provide proof-of-principle applications. As a generalized additive model, GenoGAM offers flexible choice of the response distribution. Using proportions rather than absolute counts as response variables, we show that GenoGAM is applicable to estimate DNA methylation rate from bisulfite sequencing data. Moreover, the smooth functions fitted by GenoGAM can be analyzed analytically. Our results indicate that this can be used to identify summits of narrow ChIP-Seq peaks with accuracy close to state-of-the art peak callers.

2 Materials and methods

2.1 A generalized additive model for ChIP-Seq data

We consider an experiment consisting of a set of ChIP-Seq samples. A data point is defined by a pair of a ChIP-Seq sample and a genomic position. We denote by x_i the genomic position of the *i*-th data point, by j_i its ChIP-Seq sample and by $y_i \geq 0$ the number of fragments in sample j_i centered at position x_i. For single-end libraries, the fragment center is estimated by shifting the read end position by a constant. In case of single end data, the fragment length *d* was estimated using the Bioconductor package chipseq and its *coverage* method. It is defined as the optimal shift for which the number of bases covered by any read is minimized. Thus, the center was taken as the start of the read shifted by $\frac{d}{2}$ downstream. When reducing ChIP-Seq data to fragment centers rather than full base coverage, each fragment is counted only once. This reduces artificial correlation between adjacent nucleotides.

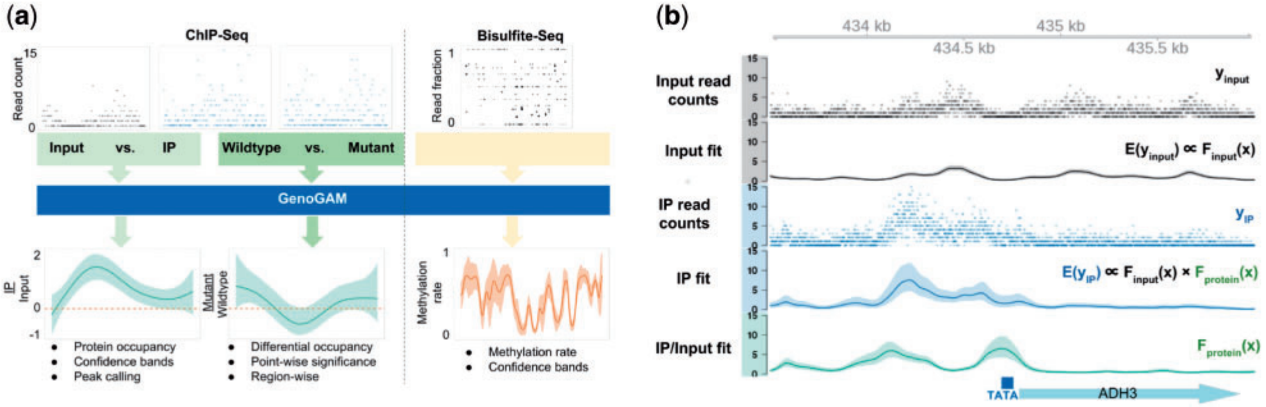

Fig. 1. GenoGAM applications and concept. (**a**) GenoGAM provides a general framework to analyze ChIP-Seq data for both absolute (left arrow) and differential protein (center arrow) occupancy. It can also be applied to infer DNA methylation rate from bisulfite sequencing data (right arrow). (**b**) ChIP-Seq analysis with GenoGAM yields base-pair resolution occupancy profiles with confidence bands. Input (top) and IP (middle) centered read counts (dots) and fitted smooth (solid line) with 95% confidence intervals (ribbons) for the transcription factor TFIIB for a section of the chromosome XIII of *S. cerevisiae*. Additionally, the extracted fold change of IP over Input (bottom) and gene annotation at the very bottom. Simplified equations depict model components

We model the counts y_i using the following generalized additive model:

$$y_i \sim \mathrm{NB}(\mu_i, \theta) \tag{1}$$

$$\log(\mu_i) = o_i + \sum_{k=1}^{K} f_k(x_i)\, z_{j_i,k} \tag{2}$$

The counts y_i are assumed to follow a negative binomial distribution with means μ_i (Equation 1) and a dispersion parameter θ that relates the variance to the mean such that $\mathrm{Var}(y_i) = \mu_i + \mu_i^2/\theta$. Consequently, the model accounts for dispersion beyond Poisson noise (Anders and Huber, 2010).

The logarithm of the mean μ_i is the sum of an offset o_i and one or more smooth functions f_k (Equation 2). The offsets o_i are predefined data-point specific constants that account for sequencing depth variations (see subsection 'Sequencing depth variations'). The indicator variable $z_{j_i,k}$ is 1 if the smooth function f_k contributes to the mean counts of sample j_i and 0 otherwise. As shown below, this formulation allows modeling IP versus input experiments as well as factorial experimental designs.

We model IP versus input experiments using GenoGAM with two smooth functions: f_{input} that contributes to both input and IP samples, and f_{protein} that only contributes to IP samples. More specifically, f_{input} models local ChIP-Seq biases common to input and IP, whereas f_{protein} models the protein log-occupancy up to one genome-wide scaling factor. Figure 1b shows the application of this model to one ChIP-Seq library for the *S. cerevisiae* general transcription factor TFIIB and its input control (Supplementary Methods).

In GenoGAM, the smooth functions are represented by cubic spline curves, which are linear combinations of a set of regularly spaced basis functions b_r, i.e. $f_k(x) = \sum_r \beta_r b_r(x)$. We chose second order B-splines as basis functions, which are bell-shaped cubic polynomials over a finite support (De Boor, 1978). To avoid overfitting, regularization of the functions f_k is carried out by penalization of the second order differences of the spline coefficients, which approximately penalizes second order derivatives of f_k—an approach called P-splines or penalized B-splines (Eilers and Marx, 1996). The optimization criterion for P-splines is the sum of the negative binomial log-likelihood (depending on the response vector y and the vector $\boldsymbol{\beta}$ containing the coefficients of all smooth functions) plus a penalty function that is weighted by the smoothing parameter λ:

$$\hat{\boldsymbol{\beta}} = \operatorname{argmax} l_{\mathrm{NB}}(\boldsymbol{\beta}; \mathbf{y}, \theta) - \lambda \boldsymbol{\beta}^{\top} \mathbf{S} \boldsymbol{\beta}, \tag{3}$$

where S is a symmetric positive matrix that encodes the squared second order differences of the coefficients $\boldsymbol{\beta}$ (Eilers and Marx, 1996). This regularization allows dense placements of the basis functions (between 20 and 50 bp), while relying on the smoothing parameter λ to protect against overfitting. Large values of λ yield smoother functions. A single smoothing parameter common to all smooth functions proved to be sufficient for our applications. For given λ and θ, model fitting was performed using penalized iteratively re-weighted least squares (see subsection 'Model fitting').

Adapting a Bayesian view, the penalized likelihood can be interpreted as a posterior probability, and the penalization term arises from a Gaussian prior on the coefficients $\boldsymbol{\beta}$. Large-sample approximations then yield a multivariate Gaussian posterior distribution for $\boldsymbol{\beta}$, and, by the linearity of $f_k(x) = \sum_r \beta_r b_r(x)$, Gaussian posteriors for the point estimates $f_k(x)$. This allows for the construction of pointwise confidence bands (Wood, 2006). An example of the fitted smooth functions and their confidence bands for the yeast transcription factor TFIIB is shown in Figure 1b.

2.2 Fitting of a GAM on a genome-wide scale

Since the computation time of a GAM grows polynomially with the number of basis functions, fitting one model to a whole chromosome is unfeasible. Instead, we propose to fit separate GAMs on sequential overlapping intervals (or tiles, Supplementary Fig. S1a). Each chromosome was partitioned into equally sized intervals called chunks. Tiles were defined as chunks extended on either side by equally sized overhangs. The generalized additive model was fitted on each tile separately using the *gam* function of the R package `mgcv`. Point estimates at each base pair of the smooth functions and their standard errors were extracted with the *predict* function on the fitted object setting 'type' parameter to 'iterms'. The tile fits were then restricted to their chunk to define the chromosome-wide fit. As overlap length increases, agreement of the fit at the midpoint of the overlapping region increases. A genome-wide fit is obtained by joining together tile fits at overlap midpoints (Supplementary Fig. S1a). This approximation yields computation times that are linear in the number of basis functions at no practical precision cost (Fig. S1b). Furthermore, it allows for parallelization, with speed-ups being linear in the number of cores (Supplementary Fig. S1c). This approximation parallelizes the computation over the data, which will allow future implementation of GenoGAM in map-reduce frameworks such as Spark (Zaharia *et al.*, 2010).

2.3 Data-driven determination of the smoothing and dispersion parameters λ and θ

To determine the optimal value for λ and θ, we tried generalized cross-validation, an analytical leave-one-out large-sample approximation (Wood, 2006). However, this yielded very wiggly fits indicative of overfitting. We thus turned to empirical cross-validation.

For efficiency, cross validation was performed using only a subset of the data. We selected a sufficiently large set of distinct regions that are long enough to not suffer from border effects common to spline fitting. Using 20 or more distinct regions containing at least 100 basis functions gave satisfactory empirical results (Supplementary Table S1). For peak calling purposes, regions were selected that had the most significant fold change of IP versus input read counts.

In each region, 10-fold cross-validation was performed, where a tenth of the data points were removed, the model was fitted on the remaining data points, and the log-likelihood of the left-out data points was computed. To avoid overfitting due to short range correlations, each cross-validation fold did not consist of randomly selected single genomic positions, which would recapitulate the leave-one-out scheme, but of short intervals. The length of these intervals was set to 20 bp (approximately one-tenth of the fragment length.) in absence of replicates and twice the average fragment sizes when replicates were available.

For a given pair of values for λ and θ, the score function was defined as the sum of out-of-sample log-likelihood over all cross-validation folds and all tiles, restricted to the data points within chunks to not depend on poor fitting in overhangs. Investigation on grid values of θ and λ showed that the out-of-sample log-likelihood was typically unimodal. We therefore used a numerical optimizer to jointly fit the two parameters (R function *optim*, BFGS method with default finite-difference approximation of the gradient).

2.4 Sequencing depth variations

We used an approach originally suggested by Meyer and Liu (2014) that was robust to variations in signal-to-noise ratio. Variations for sequencing depth was controlled by using size factors computed by DESeq2 (Love *et al.* (2014) version 2_1.10.0). This method robustly estimates fold-changes in overall sequencing depth by comparing read counts of predefined regions. The selection criteria for these regions was application-specific. For differential binding application, all tiles were considered. For IP versus input comparisons (TFIIB and peak calling applications see Supplementary information), the selected regions were the 1000 tiles with smallest *P*-value according to DESeq2 test for enrichment of IP over input performed on total read counts per tile. This allowed to select tiles that were most likely containing peaks.

2.5 TFIIB dataset

This dataset consisted of two samples: one input and one IP without replicates (Processing in Supplementary information). Hence, there was no need for an offset. We used the following GenoGAM model:

$$y_i \sim \mathrm{NB}(\mu_i, \theta)$$

$$\log(\mu_i) = f_{\mathrm{input}}(x_i) + f_{\mathrm{protein}}(x_i) z_{j_i,\mathrm{protein}},$$

where $z_{j_i,\mathrm{protein}} = 1$ whenever j_i is the index of an IP sample and $z_{j_i,\mathrm{protein}} = 0$ whenever j_i is the index of an input sample. Further parameter details are given in Supplementary Table S1.

2.6 Differential binding

2.6.1 Data

This dataset consisted of four samples: two biological replicate IPs for the wild-type strain and two biological replicate IPs for the mutant strain. Data processing and gene boundaries definition are described in Supplementary information.

2.6.2 GenoGAM model

We used a GenoGAM model that compares the mutant with the wild-type ChIP data as follows:

$$y_i \sim \mathrm{NB}(\mu_i, \theta)$$

$$\log(\mu_i) = \log(s_{j_i}) + f_{\mathrm{WT}}(x_i) + f_{\mathrm{mutant/WT}}(x_i) z_{j_i,\mathrm{mutant}}$$

where $z_{j_i,\mathrm{mutant}} = 1$ for j index of mutant samples and 0 for wild-type samples. The offsets $\log(s_{j_i})$ are log-size factors computed to control for sequencing depth variation and overall H3K4me3 across all four samples (see subsection ‘Sequencing depth variations’).

2.6.3 Position-level significance testing

Null hypotheses of the form $H_0 : f_k(x) = 0$ for a smooth function f_k at a given position x of interest were tested assuming approximate normal distribution of the corresponding z-score, i.e.:

$$T_k(x) = \frac{\widehat{f}_k(x)}{\widehat{\sigma}^2_{f_k(x)}} \sim N(0, 1)$$

where $\widehat{f}_k(x)$ and $\widehat{\sigma}^2_{f_k(x)}$ denote point estimate and standard error of the smoothed value using the *predict* function of the R package `mgcv` (Wood, 2006).

2.6.4 False discovery rate for predefined regions

Let $R_1, \ldots, R_p$ be p regions of interest, where a region is defined as a set of genomic positions. Regions are typically, but not necessarily, intervals (e.g. genes or promoters). For instance, all exons of a gene could make up a single region. Regions can be a priori defined or defined on the data using independent filtering (Bourgon *et al.*, 2010). For instance, when testing for significant differences between two conditions, regions can be selected for having a large total number of reads over the two conditions (Lun and Smyth, 2014).

For j in $1, .., p$, let H_0^j be the composite null hypothesis that the smooth function f_k values 0 at every position of the region R_j. The False Discovery Rate was controlled as in Lun and Smyth (2014):

1. Position-level *P*-values at all region positions were computed using position-level significance testing as described earlier.
2. Within each region R_j, position-level *P*-values were corrected for multiple testing using Hochberg family-wise error rate correction (Hochberg, 1988). The *P*-value for the null hypothesis H_0^j was then computed as the minimal family-wise error rate corrected position-level *P*-value. This step gives one *P*-value per region.
3. FDRs were controlled using the Benjamini–Hochberg procedure (Benjamini and Hochberg, 1995) applied to the region-level *P*-values.

2.6.5 Benchmarking

THOR allows restricting the search to pre-defined regions. However, running THOR in this mode yielded worse results. We therefore called differential binding sites with a genome-wide run of THOR and then overlapped them with the gene coordinates. The same was done for PePr and diffReps, because these methods do not allow for a fixed set of regions as input. Supplementary information gives further details on how the competitor methods were run.

Gene expression levels were computed as the median normalized probe levels for the three replicate YPD conditions of all tiling array probes provided by Xu *et al.* (2009) overlapping gene coordinates defined by Xu *et al.* (2009). Genes from Xu *et al.* (2009) and from Thornton *et al.* (2014) were matched by symbol.

To compute ROC curves, binary labels (expressed = 1 if above a given expression level quantile cutoff, or not = 0 otherwise) were assigned to each gene, and for each method, genes were ranked according to their respective significance value. For THOR, PePr and diffReps, genes that did not overlap any differentially bound site, *P*-values were set to 1. Then, ROC curves and AUC for all expression level quantile cutoffs in steps of 0.01 were computed.

3 Methylation data

To model y_i, the number of reads of methylated state, out of n_i, the total number of reads, we used the quasi-binomial model defined by:

$$\mathrm{E}(y_i/n_i) = \mu_i$$

$$\log\left(\frac{\mu_i}{1-\mu_i}\right) = f_{\mathrm{methylation}}(x_i)$$

$$\mathrm{Var}(y_i/n_i) = \theta \cdot \frac{\mu_i(1-\mu_i)}{n_i},$$

where the scale parameter $\theta > 0$ models dispersion. The model was applied on only one tile with a width of 120 kb. Further parameter details are given in Supplementary Table S1.

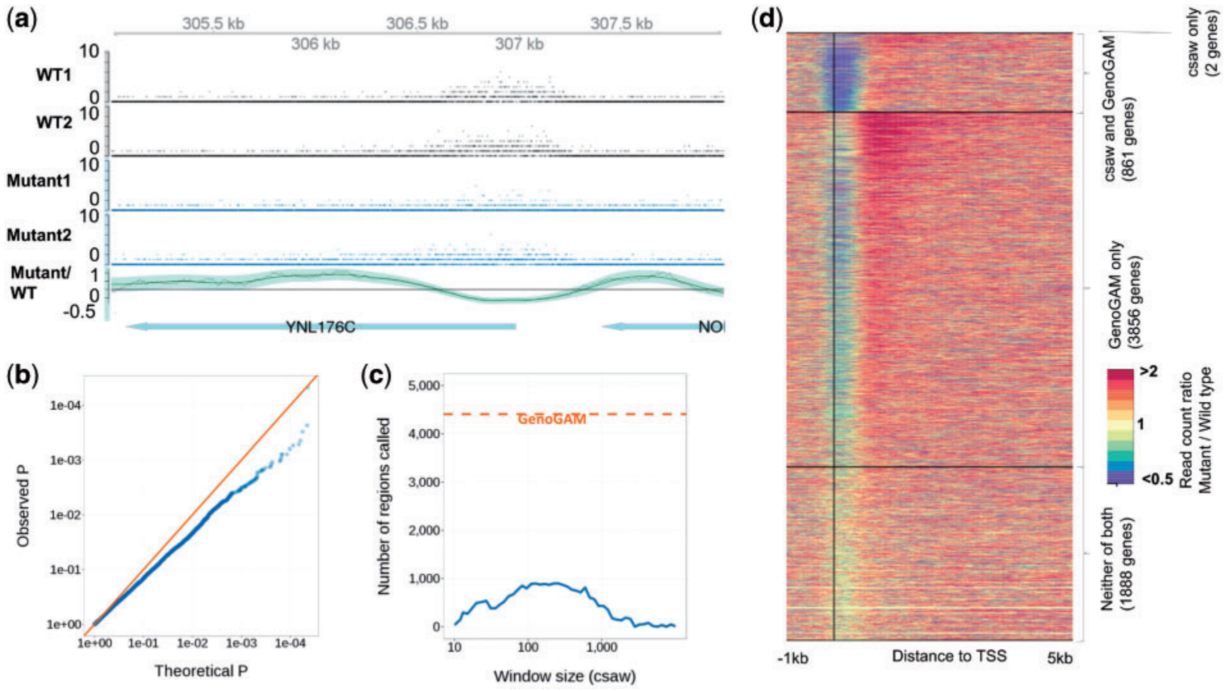

Fig. 2. Statistical testing for factorial designs. (**a**) Read counts (dots) and fitted rates with 95% confidence bands for wild-type (black) and mutant (blue) and the log-ratio of mutant over wild-type with confidence band (bottom row, green) around *YNL176C*. For comparison, log-ratios computed in sliding windows of size 184 bp (bottom row, gray, optimized window size, see section 'Comparison of GenoGAM fit with sliding window smoothing'). (**b**) Empirical (*y*-axis) versus theoretical (*x*-axis) *P*-values in base-level permuted count data (Supplementary Methods). *P*-values at every 200 bp positions are shown. (**c**) Number of genes with significant differential occupancies in mutant over wild-type (FDR < 0.1) reported by GenoGAM (orange) and by csaw (blue) as function of window size (*x*-axis). (**d**) Fold-change of counts in mutant over wild-type in 150 bp windows for all 6607 yeast genes in the -1 to 5-kb region centered on TSS (vertical black line). The genes are sorted into four groups (separated by the black horizontal lines) according to which method reports them significant. From top to bottom: csaw only (2 genes), csaw and GenoGAM (861 genes), GenoGAM only (3856 genes) and none (1888 genes). Within each group genes are ordered by *P*-values (lowest to highest from top to bottom). The 'csaw and GenoGAM' group is sorted by GenoGAM *P*-values. Comparisons to all other methods can be found in Supplementary Figure S3–S5

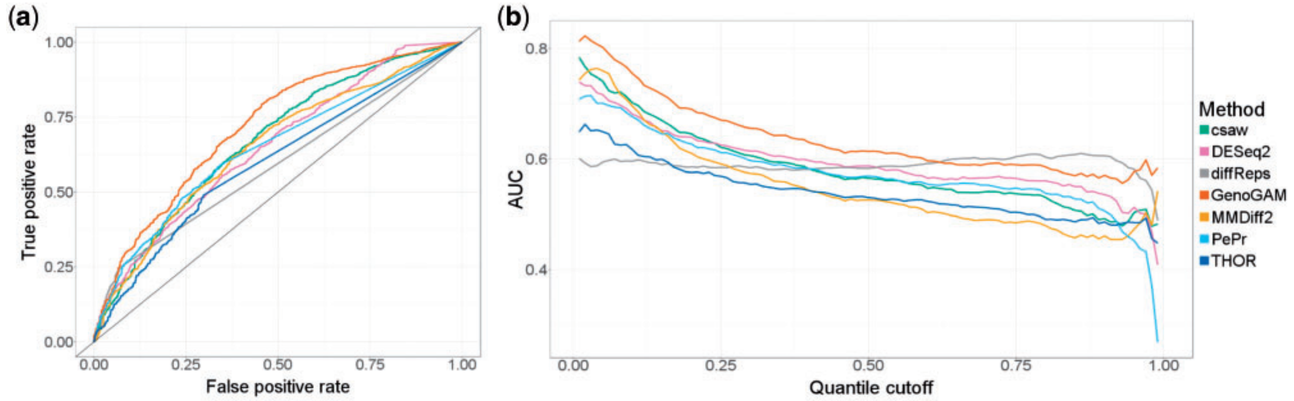

Fig. 3. GenoGAM identifies differential regions with greater recall. (**a**) Area under the curve (AUC) for all possible quantile cutoffs from 0 to 1 in steps of 0.01. Up to a cutoff of 0.6, GenoGAM (red) performs consistently better than all competitor methods by ~0.03–0.04 points above the second best method (csaw and DESeq2, green and pink, respectively). The entire range of quantile cutoffs is shown out of completeness, reasonable values are between 0.15 and 0.25. (**b**) ROC curve based on a quantile cutoff of 0.15 (see Figure S7 for other cutoffs). GenoGAM has a constantly higher recall with a lower false positive rate. The partially straight lines for THOR, PePr and diffReps are stemming from tied genes with no significance value. (**c**) Boxplots showing gene expression levels for significant and not significant genes according to the respective threshold of the method (0.1 for FDR based methods and 1e-5 for PePr). Gene expression levels are clearly distinct between the group of significant and not significant genes across all methods

4 Results

4.1 Testing for differential occupancy with GenoGAM with controlled type I error

To assess the performance of GenoGAM for calling differential occupancy, we re-analyzed histone H3 Lysine 4 trimethylation (H3K4me3) ChIP-Seq data of a study (Thornton *et al.*, 2014) comparing wild-type yeast versus a mutant with a truncated form of Set1, the H3 Lysine 4 methylase. H3K4me3 is a hallmark of promoters of actively transcribed genes. Thornton *et al.* (2014) have reported genome-wide redistribution in the truncated Set1 mutant of

H3K4me3, which is depleted at the promoter and enriched in the gene body. This dataset is interesting for differential occupancy analysis because it is not about the overall number of counts, but about the redistribution of H3K4me3 within the gene. Hence, methods must be sensitive to differences at any location within the gene. We expect such redistribution of the mark at all genes that are transcriptionally active for yeast cells grown in rich media.

We modeled the two replicate IPs for mutant and for the wild-type with GenoGAM using one smooth function f_{WT} for the wild-type reference occupancy, and one further smooth function $f_{\mathrm{mutant/WT}}$ for the differential effect (see 'Methods'). The offsets were computed to control for variations in sequencing depth between replicates and overall genome-wide H3K4me3 level (see 'Methods'). This yielded base-level log-ratio estimates and their 95% confidence bands genome-wide (see 'Methods', Fig. 3a for data and fit at the gene *YNL176C* consistent with the report of reduced binding at promoter regions).

Confidence bands of GAMs are formally Bayesian credible intervals (see 'Methods'). However, previous studies based on simulated data showed that these confidence bands have close to nominal coverage probabilities and can, in practice, be used in place of frequentist confidence intervals (Marra and Wood, 2012). We estimated base-level *P*-values using the point-wise estimates and standard deviations (see 'Methods'). To empirically verify that the *P*-values were at least conservative, we created a negative control dataset by per-base-pair independent permutation of the counts between the four samples. The offsets were set to 0 and the smoothing and dispersion parameters were estimated again. This non-parametric permutation scheme makes less assumptions than previous simulation studies (Marra and Wood, 2012). Nonetheless, per-base-pair *P*-values in this negative control experiment were slightly overestimated (Fig. 3b). These results show that GenoGAM can be used to identify individual positions of significant differential occupancies with controlled type I error. Here, correction for multiple testing can either be done using the Benjamini and Hochberg procedure (Benjamini and Hochberg, 1995) or procedures that exploit dependencies between adjacent positions (Wei *et al.*, 2009).

Complementary to *de novo* identification, predefined regions, such as genes, can be tested for differential occupancies. To test for differences at any position in a region using GenoGAM, we propose to apply Hochberg's procedure to correct the pointwise *P*-values for multiple testing, and to report the smallest of these corrected *P*-values (see 'Methods'). We confirmed by permutation analysis that this approached conservatively controlled for type I error rate (Supplementary Fig. S2).

4.2 Higher sensitivity in testing for differential occupancy

We first compared GenoGAM to csaw, which is its most directly comparable method because only GenoGAM and csaw can model flexible factorial designs and assess differences in overall read counts and in shape. One fundamental difference is that csaw is based on a sliding window approach requiring an a priori defined window size. In contrast, the smoothing parameter of GenoGAM is learnt from the data by maximizing the out-of-sample likelihood in cross-validation (see 'Methods'). Across all investigated window sizes, the csaw algorithm reported a maximum of 863 significant genes at FDR < 0.1 (Fig. 3c). Moreover, the number of identified genes depended strongly on the choice of the window size (Fig. 3c). In contrast, GenoGAM reported 4717 significant genes at the same FDR cutoff, which is much closer to the number of transcriptionally active genes (Xu *et al.*, 2009). The genes reported by GenoGAM included all the genes reported by csaw except two, indicating that GenoGAM captured the same signal but with a higher sensitivity (Fig. 3d). The genes reported only by GenoGAM showed a differential occupancy pattern similar yet weaker to the genes common to csaw and GenoGAM, with depletion in the promoter and enrichment in the gene body (Fig. 3d), indicating that GenoGAM captured true biological signal.

We next compared GenoGAM against a comprehensive set of differential occupancy methods that proved to be competitive in a recent benchmark (Steinhauser *et al.*, 2016). These methods were DESeq2 (Anders and Huber, 2010), MMDiff2, the current version of MMDiff (Schweikert *et al.*, 2013), csaw (Lun and Smyth, 2014), diffReps (Shen *et al.*, 2013), and PePr (Zhang *et al.*, 2014). Two more methods highlighted by Steinhauser *et al.* (2016) were excluded: ChIPComp, as the R package is hardcoded to be used on mouse and human datasets only, and DiffBind, which is redundant, since it is essentially a test for differences in overall counts based on either DESeq2 (already present) or edgeR (used by csaw). We furthermore included the HMM-based method THOR (Allhoff *et al.*, 2016) (see 'Methods').

The least number of significant genes (FDR < 0.1 or the respective default threshold set by the method) were identified by DESeq2 (735) csaw (863) and diffReps (1193). The most were reported by THOR (2687), PePr (3248) and MMDiff2 (3482), closer to GenoGAM. To make sure that i) the reported genes indeed corresponded to active genes (also see Supplementary Fig. S6) and ii) that these results did not depend on FDR cutoffs we performed receiver operating characteristic (ROC) using expressed genes as a proxy for true positives (see 'Methods'). GenoGAM had the largest Area Under the ROC curve, when considering that the 15% of the genes with lowest expression levels in Xu *et al.* (2009) are not expressed (Fig. 4a, see 'Methods'). Moreover, GenoGAM consistently had the largest AUC for any gene expression cutoff up to 60% genes to be not expressed (Fig. 4b). These results indicate that GenoGAM is more sensitive than current methods for testing differential occcupancy, while still controlling for type I error rate.

4.3 Comparison of GenoGAM fit with sliding window smoothing

In the uncommon situation where a benchmark is available as for the Thornton *et al.* dataset, one can objectively define an optimal window size for sliding window approaches. The log-ratios estimated by GenoGAM fit well to log-ratios computed in sliding windows of size 184bp, the window size maximizing the area under curve for csaw for a gene expression quantile cutoff of 0.15 (Fig. 3a). In addition, the GenoGAM 95% confidence ribbon captures very well the short-range fluctuations of the sliding window estimates. Hence there is a general agreement between the two approaches. However, the benefits of GenoGAM are clear: First, the GenoGAM fit is smooth and differentiable. Second, unlike in the window-based approach, the amount of smoothing is solely estimated from the ChIP-Seq data, without prior knowledge from the benchmark.

4.4 Application to DNA methylation data

Generalized additive models are based on the generalized linear modeling framework and thus allow any distribution of the exponential family for the response. Therefore, GenoGAM can be also used to model continuous responses, for instance using the Gaussian distribution, and proportions using the Binomial distribution. For ChIP-Seq data, a log-linear predictor–response relationship of the form (Equation 2) is justified by the fact that effects on the mean are

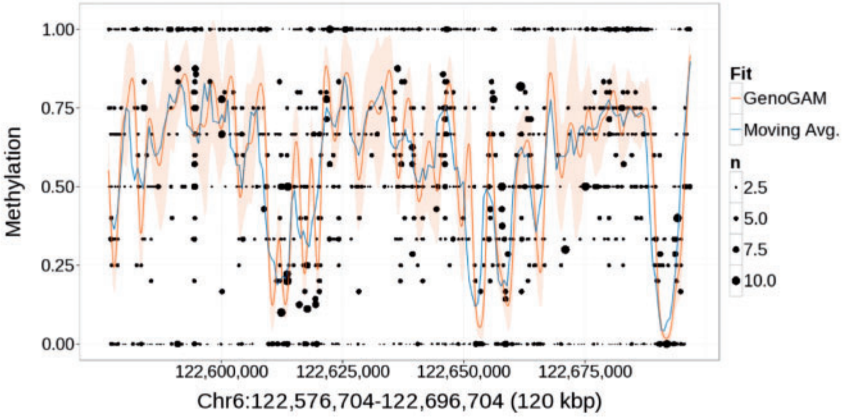

Fig. 4. Application to DNA methylation data. Estimated DNA methylation rates in a 120-kb region of chromosome 6 of the mouse (cf. Smallwood *et al.* 2014). Shown are the data for bulk embryonic mouse stem cells grown in serum; ratios of methylated counts for each CpG position (black dots), with point size proportional to the number of reads. The estimated rates are shown for the moving average approach Smallwood *et al.* (2014) of 3000 bp bins in 600 bp steps (blue line) and for the GenoGAM (orange line) with 95% confidence band (ribbon)

typically multiplicative. However, other monotonic link functions could also be used. Moreover, quasi-likelihood approaches are supported, allowing for the specification of flexible mean–variance relationships (Wedderburn, 1974).

To test the flexibility of GenoGAM, we conducted a proof-of-principle study on modeling bisulfite sequencing of bulk embryonic mouse stem cells grown in serum (Smallwood *et al.*, 2014). Bisulfite sequencing quantifies methylation rate by converting cytosine residues to uracil, leaving 5-methylcytosine residues unaffected. At each cytosine, the data consisted of the number n_i of fragments overlapping the cytosine and the number y_i of these fragments for which the cytosine was not converted to uracil. The quantity of interest was the methylation rate, i.e. the expectation of the ratio y_i/n_i. In the original publication, single nucleotide position methylation rates were estimated using a sliding window approach with an *ad hoc* choice of window size of 3 kb computed in steps of 600 bp. Figure 2 reproduces an original figure showing the fit in a 120-kb section of chromosome 6. We modeled this 120-kb section with GenoGAM using a quasi-binomial model, where the response was the number of successes y_i out of n_i trials, the log-odd ratio was modeled as a smooth function of the genomic position, and the variance was equal to a dispersion parameter times the variance of the binomial distribution. Smoothing and dispersion parameters were determined by cross-validation (see 'Methods'). The GenoGAM fit was consistent with the original publication (Smallwood *et al.*, 2014), but did not rely on manually set window sizes and provided confidence bands (Fig. 2). As expected, wider confidence bands were obtained in regions of sparse data and tighter bands in regions with a lot of data (Fig. 2).

4.5 Calling ChIP-Seq peak summits

The smooth function estimates and their representation as P-splines provided by GAM offer new opportunities for subsequent analyses: First and second order derivatives can be computed immediately. Those can be used to infer summits of ChIP-Seq peaks (as positions x where $f'(x) = 0$ and $f''(x) < 0$). Supplementary information shows how a peak caller can be therefore built, including a test for statistical significance of the peaks (Supplementary Fig. S8). Comparison of this approach to a few widely used peak callers [MACS (Zhang *et al.*, 2008), JAMM (Ibrahim *et al.*, 2015) and ZINBA (Rashid *et al.*, 2011)] on small size datasets (Human chromosome 22 and yeast) indicates reasonable performance (Supplementary Figs S9–S12).

4.6 Implementation and current computational limitations

We implemented the method in the freely available Bioconductor R package `GenoGAM`. Given a configuration file of the BAM files, experiment design matrix and model formula, it automatically estimates all parameters. Alternatively, users can provide size factors or smoothing and overdispersion parameters. `GenoGAM` provides downstream analysis functions for differential binding and peak calling. `GenoGAM` supports a number of parallel backends through the Bioconductor parallel framework `BiocParallel`.

The genome-wide analysis on yeast, which is 12 Mb long, presented in this paper took ~20 h on 60 cores including parameter estimation by cross-validation. Once the smoothing and overdispersion parameters are estimated, the runtime reduces to around an hour. Nonetheless, runtime for whole human genome, which is ~3 Gb is long, is at this stage not practical. Our main focus so far has been to establish the framework and to evaluate its statistical properties, which we present in this paper. Solutions to these computational limitations are currently being addressed with promising results, in order to extend GenoGAM to genome-wide application for larger genomes.

5 Discussion

We have introduced a generic framework based on generalized additive models to model ChIP-Seq data. Unlike most other methods for ChIP-Seq analysis, GenoGAM is a data generative model, which gives an explicit likelihood of the data. This in turn yields an objective criterion to set the amount of smoothing. Smoothing and dispersion parameters were obtained by cross-validation, i.e. they were fitted for the accuracy in predicting unseen data. This criterion turned out to provide useful values of smoothing and dispersion for inference. Moreover it led to reasonable uncertainty estimates since confidence bands of the fits were found to be only slightly conservative. To our best knowledge, GenoGAM is the first method so far that has addressed the setting of the amount of smoothing for ChIP-Seq data. The possibility exists to estimate the smoothing and dispersion parameters separately for each sample, which would result in more robust estimates at the cost of some flexibility. However, in our analyses the samples within an experiment were all similar enough to estimate the parameters globally.

The utilization of genome-wide GAMs comes with a number of advantages: First, we flexibly model factorial designs, as well as replicates with different sequencing depths using size factors as offsets. More elaborate usage could include position- and sample-specific copy number variations, or GC-biases. Second, applying GAMs yields confidence bands as a measure of local uncertainty for the estimated rates. We showed how these can be the basis to compute point-wise and region-wise *P*-values. Third, GAMs output analytically differentiable smooth functions, allowing flexible downstream analysis. We discussed how peak calling can be elegantly handled by making use of the first and second derivatives. Fourth, various link functions and distributions can be used, providing the possibility to model a wide range of genomic data beyond ChIP-Seq, as we illustrated with a first application on DNA methylation. Hence, we foresee GenoGAM as a generic method for the analysis of genome-wide assays.

Scalability to fit very long longitudinal data such as whole chromosomes at base-pair resolution was made possible by parallelization over the data and allowing approximations rather than exact computation of the fit (Heinis, 2014). Nonetheless, practical usage of our current implementation remains limited to organisms with small genomes such as yeast or bacteria, or to selected subsets of larger genomes, such as promoters. Despite the present limitations in runtime and genome size, we are confident that GenoGAM is of importance to the bioinformatics community. Future research directions include improving the computing time, for instance leveraging on recent progresses for fitting large GAMs (Wood *et al.*, 2016).

Author's Contributions

Conceived the project and supervised the work: J.G. and A.T. Developed the software and carried out the analysis: G.S., A.E. and J.G. Carried out the ChIP-Seq experiments for TFIIB on yeast: D.S. Gave advice on statistics: M.S. Wrote the manuscript: J.G., A.E., G.S., M.S. and A.T.

Acknowledgements

We thank Ulrike Gaul, Ulrich Unnerstall and Michael Lidschreiber for fruitful discussions on data analysis, Martin Morgan and Hervé Pagès for support during the implementation of the GenoGAM package, Stefan Krebs for sequencing and raw data preprocessing, and Ulrich Mansmann and Patrick Cramer for institutional support.

Funding

J.G. was supported by the Bavarian Research Center for Molecular Biosystems and by the Bundesministerium für Bildung und Forschung through the Juniorverbund in der Systemmedizin 'mitOmics' (FKZ 01ZX1405A). This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 633974 (J.G. and G.S.) as well as from the Graduate School for Quantitative Biosciences Munich (QBM) and Federal Ministry of Education and Research (BMBF) (e:Bio framework, SysCore 0316176) for A.E. and A.T.

Conflict of Interest: none declared.

References

Albert,I. *et al.* (2007) Translational and rotational settings of H2A.Z nucleosomes across the *Saccharomyces cerevisiae* genome. *Nature*, **446**, 572–576.

Allhoff,M. *et al.* (2016) Differential peak calling of Chip-Seq signals with replicates with thor. *Nucleic Acids Res*, **44**, e153.

Anders,S. and Huber,W. (2010) Differential expression analysis for sequence count data. *Genome Biol.*, **11**, R106–R106.

Barski,A. *et al.* (2007) High-resolution profiling of histone methylations in the human genome. *Cell*, **129**, 823–837.

Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. B*, **57**, 289–300.

Bourgon,R. *et al.* (2010) Independent filtering increases detection power for high-throughput experiments. *Proc. Natl. Acad. Sci.*, **107**, 9546–9551.

De Boor,C. (1978). *A Practical Guide to Splines*. Vol. **27**. Springer-Verlag, New York.

Eilers,P.H. and Marx,B.D. (1996) Flexible smoothing with B-splines and penalties. *Stat. Sci.*, 89–102.

Hastie,T. *et al.* (1986) Generalized additive models. *Stat. Sci.*, **1**, 297–310.

Heinis,T. (2014) Data analysis: approximation aids handling of big data. *Nature*, **515**, 198.

Hochberg,Y. (1988) A sharper Bonferroni procedure for multiple tests of significance. *Biometrika*, **75**, 800–802.

Ibrahim,M.M. *et al.* (2015) JAMM: a peak finder for joint analysis of NGS replicates. *Bioinformatics*, **31**, 48–55.

Johnson,D.S. *et al.* (2007) Genome-wide mapping of in vivo protein-DNA interactions. *Science*, **316**, 1497–1502.

Love,M.I. *et al.* (2014) Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2. *Genome Biol.*, **15**, 550.

Lun,A.T., and Smyth,G.K. (2014) De novo detection of differentially bound regions for ChIP-Seq data using peaks and windows: controlling error rates correctly. *Nucleic Acids Res.*, **42**, e95–e95.

Marra,G., and Wood,S.N. (2012) Coverage properties of confidence intervals for generalized additive model components. *Scand. J. Stat.*, **39**, 53–74.

Meyer,C.A., and Liu,X.S. (2014) Identifying and mitigating bias in next-generation sequencing methods for chromatin biology. *Nat. Rev. Genet.*, **15**, 709–721.

Nelder,J.A., and Wedderburn,R.W.M. (1972) Generalized linear models. *J. R. Stat. Soc. A*, **135**, 370–384.

Rashid,N.U. *et al.* (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.

Robertson,G. *et al.* (2007) Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat. Methods*, **4**, 651–657.

Schweikert,G. *et al.* (2013) MMDiff: quantitative testing for shape changes in ChIP-Seq data sets. *BMC Genomics*, **14**, 826.

Shen,L. *et al.* (2013) diffReps: detecting differential chromatin modification sites from ChIP-Seq data with biological replicates. *PLoS One*, **8**, e65598.

Smallwood,S.A. *et al.* (2014) Single-cell genome-wide bisulfite sequencing for assessing epigenetic heterogeneity. *Nat. Methods*, **11**, 817–820.

Steinhauser,S. *et al.* (2016) A comprehensive comparison of tools for differential ChIP-Seq analysis. *Brief. Bioinformatics*, **17**, 953–966.

Thornton,J. *et al.* (2014) Context dependency of Set1/COMPASS-mediated histone H3 Lys4 trimethylation. *Genes Dev.*, **28**, 115–120.

Wedderburn,R.W. (1974) Quasi-likelihood functions, generalized linear models, and the Gauss Newton method. *Biometrika*, **61**, 439–447.

Wei,Z. *et al.* (2009) Multiple testing in genome-wide association studies via hidden Markov models. *Bioinformatics (Oxford, England)*, **25**, 2802–2808.

Wood,S. (2006). *Generalized Additive Models: An Introduction with R*. CRC Press, Boca Raton, FL.

Wood,S.N. *et al.* (2016) Generalized additive models for gigadata: modelling the UK black smoke network daily data. *J. Am. Stat. Assoc.*, 1–40.

Xu,Z. *et al.* (2009) Bidirectional promoters generate pervasive transcription in yeast. *Nature*, **457**, 1033–1037.

Zaharia,M. *et al.* (2010). Spark: Cluster Computing with Working Sets. In *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, USENIX Association Berkeley, CA, Vol. 10, p. 10.

Zhang,Y. *et al.* (2008) Model-based analysis of ChIP-Seq (MACS). *Genome Biol.*, **9**, R137.

Zhang,Y. *et al.* (2014) Pepr: A peak-calling prioritization pipeline to identify consistent or differential peaks from replicated ChIP-Seq data. *Bioinformatics*, **30**, 2568–2575.